EVALUASI BERBASIS *NORMALIZE MUTUAL INFORMATION* UNTUK PENGELOMPOKAN PENELITIAN DENGAN *K-MEANS CLUSTERING*

Rahmat, Mustikasari*, Nur Afif
Jurusan Teknik Informatika
Fakultas Sains dan Teknologi Universitas Islam Negeri Alauddin Makassar
Jl. Sultan Alauddin No. 63, Gowa, Sulawesi Selatan, Indonesia. 92113
*E-mail: mustikasari@uin-alauddin.ac.id

Abstrak: Secara manual, aglomerasi dokumen dapat diklasifikasikan berdasarkan judul dokumen. Untuk jumlah dokumen yang tidak terlalu banyak, cara ini masih mungkin dilakukan, namun jika jumlah dokumen yang akan dikelompokkan bertambah, maka akan memakan waktu yang cukup lama. Metode yang digunakan dalam penelitian ini bekerja dengan mengklasifikasikan dokumen penelitian untuk memberikan informasi tentang hubungan antara isi penelitian yang satu dengan penelitian yang lain sehingga dapat dijadikan acuan bagi peneliti dengan fokus area yang berbeda, dalam memetakan skema kerjasama penelitian yang akan datang. Proses pengelompokan judul penelitian dilakukan dengan menggunakan metode *k-Means Clustering* pada sekumpulan judul penelitian dengan mengambil judul penelitian dan abstrak sebagai informasi yang dapat mewakili isi dokumen. Dokumen akan melalui proses *preprocessing* menggunakan metode *text mining*. Selanjutnya, judul dapat dikelompokkan menggunakan metode yang diterapkan. Hasilnya adalah membuat aplikasi yang secara otomatis dapat mengklasifikasikan judul penelitian untuk nilai k. Untuk mencapai keseimbangan dalam pemilihan jumlah *cluster* dan kualitas hasil pengelompokan, maka teknik pengujian kualitas *cluster* yang digunakan dalam penelitian ini adalah *Normalized Mutual Information* (NMI). Percobaan dilakukan dengan memasukkan jumlah cluster yang bervariasi untuk mendapatkan hasil pengelompokan terbaik dengan nilai NMI.

Kata Kunci: *clustering*; dokumen; *mutual information*; *text mining*

Abstract: Manually, document agglomerations can be classified based on the document title. For the number of documents that are not excessively enough, this method is still possible to do, however, if the number of documents to be grouped increases, then it will take considerable time. The method used in this research works to classify research documents to provide information about the relationship between the contents of one research and another so that it can be used as a reference for researchers with different focus areas, in mapping future research collaboration schemes. The process of grouping research titles is carried out using the k-Means Clustering method on a set of research titles by taking research titles and abstracts as information that can represent the contents of the document. The document will go through a preprocessing process using the text mining method. Furthermore, the titles can be grouped using the applied method. The result is to create an application that can automatically classify research titles for k values. In order to achieve a balance in the selection of the number of clusters and the quality of the grouping

results, the cluster quality testing technique used in this study is Normalized Mutual Information (NMI). The experiment was carried out by entering a varying number of clusters to get the best grouping results with the NMI value.

Keywords: clustering; documents; mutual information; text mining

PENDAHULUAN

Setiap tahun pemerintah menggenjot peningkatan kualitas pendidikan serta kegiatan penelitian dan publikasi. Berbagai dukungan diberikan, baik berupa insentif maupun penghargaan kepada para peneliti. Hal ini berhasil mendorong bertambahnya aktivitas penelitian setiap tahunnya. Bagaimanapun juga, kegiatan penelitian ini dipacu, selain untuk menumbuhkan semangat menulis dan berkarya, diharapkan ke depannya, bahwa karya hasil penelitian tersebut dapat bermanfaat di berbagai sektor kehidupan serta berdampak pada masyarakat luas. Untuk melakukan kegiatan penelitian lintas sektor, tentu dibutuhkan analisis dan wawasan yang luas dan sistematis. Mengkaji hanya dengan mencari secara manual riset-riset terkait yang relevan akan seperti upaya pencarian jarum dalam jerami. Karenanya upaya pengelompokan judul penelitian untuk mencari sebuah fenomena baru atau kerja penelitian yang belum tuntas dalam karya penelitian terdahulu, bukan merupakan pekerjaan yang sepele. Di lain pihak, hal tersebut merupakan salah satu solusi yang dibutuhkan untuk memperoleh kemudahan pagi para peneliti, dalam mengakses kajian pada berbagai bidang terutama untuk aspek kajian yang berbeda.

Adanya berbagai macam teknologi saat ini telah membantu manusia untuk menemukan solusi yang tepat pada suatu permasalahan. Perkembangan teknologi informasi yang maju begitu pesat mulai dimanfaatkan di segala bidang kehidupan. Perkembangan teknologi ini juga telah mendorong individu maupun kelompok peneliti untuk lebih kreatif dan inovatif dalam menyelesaikan masalah yang dihadapi. Salah satu cara atau metode yang dapat digunakan untuk melakukan penelusuran dan pengelompokan dokumen adalah dengan metode *clustering* atau teknik *cluster*. Sebuah cluster adalah sekumpulan objek yang digabung bersama karena persamaan atau kedekatannya. Metode ini dapat digunakan untuk menganalisis dokumen penelitian dengan mengelompokkan secara otomatis penelitian yang memiliki kesamaan atau kemiripan. Walau demikian, kekhasan tipe data dari setiap masalah dapat menjadi tantangan tersendiri dalam menemukan solusi yang efektif dalam pencarian sebuah solusi. Salah satunya adalah pemisahan dan pengelompokan objek berdasarkan karakteristik yang dimiliki masing-masing objek secara otomatis, yang menggunakan data *text* dalam pemrosesannya. Meskipun kerja pengelompokan data sudah sejak lama diperkenalkan, namun pemrosesan data *text* memiliki tantangan tersendiri. Struktur kalimat, tanda baca dan konteks kalimat yang ambigu merupakan beberapa aspek yang tidak dapat disepelekan dalam pemrosesan data jenis ini. Beberapa penelitian terdahulu yang dilakukan Indraloka & Santosa (2017) dan Deolika et al. (2019) tentang pengelompokan data, mengadopsi sebuah teknik pemrosesan *text* yang bernama *text mining*.

Metode *Clustering* bersama dengan *text mining* dalam penelitian ini, diharapkan dapat membantu dalam mengelompokkan dokumen penelitian yang nantinya dapat berguna untuk memberikan informasi tentang penelitian satu dengan yang lainnya yang memiliki keterkaitan satu sama lain serta dapat menjadi acuan bagi peneliti dalam

menganalisis atau menyeleksi penelitian, untuk dijadikan acuan dalam memetakan skema kolaborasi penelitian prioritas yang dapat dilakukan di masa depan.

METODE PENELITIAN

Beberapa kerja terkait pengelompokan penelitian telah dilakukan. Pertama, Penelitian dilakukan oleh Herny Februarianty yang berjudul "*Hirarchical Agglomerative Clustering* untuk Pengelompokan Penelitian Mahasiswa". Penelitian Februariyanti & Santoso (2017) ini juga bertujuan untuk mengelompokkan data penelitian menggunakan algoritma agglomeratif. Tidak diperolehnya teknik validasi pada metode tersebut adalah suatu yang perlu digali lebih jauh pada penelitian ini. Berbeda metode dengan penelitian tersebut, penelitian yang diusulkan ini menggunakan *K-means clustering* sebagai metode pengelompokan. Hal ini dipilih berhubung data penelitian ini akan terus bertambah dari waktu ke waktu, dibandingkan dengan *Hirarchical agglomerative clustering*, algoritma *K-means* lebih mampu dalam menghandel data besar dan kompetitif dalam komputasi.

Penelitian juga telah dilakukan oleh Widodo dan Dina Wahyudi yang berjudul "Implementasi Algoritma *K-Means Clustering* Untuk Mengetahui Bidang Penelitian Mahasiswa Multimedia Pendidikan Teknik Informatika dan komputer Universitas Negeri Jakarta". Penelitian ini bertujuan untuk mengetahui bidang penelitian mahasiswa peminatan multimedia PTIK UNJ angkatan tertentu yang digunakan sebagai saran bagi mahasiswa yang belum mengajukan penelitian. Sedangkan bagi mahasiswa yang telah mengajukan penelitian dapat dijadikan sebagai perbandingan antara hasil program bantu dengan bidang penelitian yang diteliti (Widodo & Wahyuni, 2017). Meskipun sama-sama menggunakan metode *K-means Clustering*, namun pada penelitian Widodo tesebut tidak menggunakan *text mining* melainkan menggunakan nilai mata kuliah sebagai data yang nantinya akan diuji atau sebagai objek penelitian, sedangkan pada penelitian ini penulis menggunakan dokumen teks yaitu judul dan abstrak penelitian sebagai data input.

Penelitian lain juga telah dilakukan oleh Nyoman Gede Yudiarta yang berjudul "Penerapan Metode *Clustering Text Mining* untuk Pengelompokan Berita Pada *Unstructured Text*". Penelitian ini bertujuan untuk membangun suatu aplikasi yang dapat mengelompokkan jenis berita menggunakan metode *text mining* dan algoritma *K-Means Clustering* (Yudiarta et al., 2018). Persaman antara penelitian tersebut dengan yang diusulkan dalam tulisan ini adalah sama-sama bertujuan membangun suatu aplikasi yang dapat mengelompokkan data dengan menggunakan *K-means clustering*. Letak perbedaannya adalah penelitian sebelumnya adalah sebuah penerapan teknik validasi untuk pengukuran keakuratan pengelompokkan data. Teknik validasi yang digunakan adalah *recall*, *precision* dan *purity*, sedangkan ketepatan pengelompokkan pada metode yang diusulkan ini, adalah menggunakan *Normalize Mutual Information* (NMI). Sebuah standar pengukuran kualitas informasi dalam mengestimasi kualitas *cluster*/ pengelompokan data judul penelitian.

1. *Text Preprocessing*

Text preprocessing dilakukan untuk mengubah data tekstual yang tidak terstruktur ke dalam data yang terstruktur dan disimpan dalam basis data. Tujuan dari *preprocessing* yakni menghasilkan sebuah set *term index* yang bisa mewakili dokumen. *Preprocessing* yang berbeda dapat memengaruhi kesimpulan (Denny & Spirling, 2018).

Komponen dari *text prepocessing* dibagi menjadi beberapa bagian, yaitu:

a. *Case Folding*

Case folding merupakan proses mengkonversi keseluruhan teks dalam dokumen menjadi satu bentuk standar (biasanya huruf kecil atau *lowercase*). Sebagai contoh, *user*

ingin mendapatkan informasi “KOMPUTER” dan mengetik “KOMPOTER”, “KomPUter”, atau “komputer”, tetap diberikan hasil *retrieval* yang sama yakni “komputer”. *Case folding* adalah mengubah semua huruf dalam dokumen menjadi huruf kecil. Hanya huruf ‘a’ sampai dengan ‘z’ yang diterima. Karakter selain huruf dihilangkan dan dianggap delimiter.

b. *Tokenizing*

Tokenizing adalah proses pemotongan *string* input berdasarkan tiap kata penyusunannya. Pada prinsipnya proses ini adalah memisahkan setiap kata yang menyusun suatu dokumen. Pada proses ini juga dilakukan penghilangan angka, tanda baca dan karakter lain huruf alfabet. Hal ini dikarenakan karakter-karakter tersebut dianggap sebagai pemisah kata (delimiter) dan tidak memiliki pengaruh terhadap pemrosesan teks.

c. *Filtering*

Filtering adalah tahap pemilihan kata-kata penting dari hasil token, yaitu kata-kata yang bisa digunakan untuk mewakili isi dari sebuah dokumen. Proses *filtering* juga biasa disebut sebagai *stopword removal*. Pada proses ini, terdapat dua teknik yang bisa dilakukan, yaitu *stop list* dan *word list*. *Stop list* yaitu membuang kata yang tidak deskriptif atau tidak penting. Sedangkan *word list* yaitu menyimpan kata yang dianggap penting.

d. *Stemming*

Steamming adalah proses pengubahan bentuk kata menjadi kata dasar atau tahap mencari *root* kata dari setiap kata hasil *filtering*. Proses *stemming* secara luas sudah digunakan di dalam *information retival* (pencarian informasi) untuk meningkatkan kualitas informasi yang akan didapatkan. Dengan dilakukannya proses *stimming* ini, setiap kata yang berimbuhan akan berubah menjadi kata dasar. Dengan demikian dapat lebih megoptimalkan proses *text mining*. *Stemming* sering digunakan sebagai teknik pengurangan kosakata karena menggabungkan berbagai bentuk kata. Namun, *stemming* terkadang dapat menggabungkan kata-kata dengan arti yang berbeda secara kontekstual ("taman merdeka", dan "merdeka belajar"), yang mungkin menyesatkan.

2. Pembobotan kata atau *Term Weighting* (TF-IDF)

a. TF (*Term Frequency*)

TF (*Term Frequency*) adalah frekuensi dari kemunculan sebuah *term* dalam dokumen yang bersangkutan. Semakin besar jumlah kemunculan suatu term (TF tinggi) dalam dokumen, semakin besar pula bobotnya atau akan diberikan nilai kesesuaian yang semakin besar.

Ada beberapa formula yang dapat digunakan:

i) TF biner (binary TF), hanya memperhatikan apakan suatu kata atau *term* ada atau tidak dalam dokumen, jika ada diberi nilai satu (1), jika tidak diberi nilai nol (0).

ii) TF murni (raw TF) nilai TF diberikan berdasarkan jumlah kemunculan suatu *term* di dokumnen. Contohnya, jika muncul lima (5) kali maka kata tersebut akan bernilai lima (5).

iii) TF logaritmatik, hal ini untuk menghindari dominasi dokumen yang mengandug sedikit *term* dalam query, namun mempunyai frequensi yang tinggi.

$$TF = \begin{cases} 1 + LOG_{10}\left(F_{t,d}\right), & f_{t,d} > 0 \\ 0, & f_{t,d} = 0 \end{cases} \quad (1)$$

Dimana nilai ft,d adalah *frequensi term* (t) pada dokumen (d). jadi jika suatu kata atau *term* terdapat dalam suatu dokumen sebanyak 5 kali maka diperoleh bobot = 1 + log(5) = 1.699 jika *term* tidak terdapat dalam dokumen tersebut, bobotnya adalah nol. Berikut ini contoh langkah-langkap penyiapan data hingga konversi TF-IDF nya.

Tabel 1. Contoh daftar judul penelitian

No	Judul Penelitian
1	Aplikasi Media Pembelajaran Matematika Tingkat SMU berbasis Android
2	Media Aplikasi *E-Marketplace* Jasa Desain Grafis berbasis Android
3	Perancangan Alat Pendeteksi Hewan Pengganggu Tanaman Kebun Menggunakan Sensor Gerak PIR (*Passive Infra Red*) berbasis Mikrokontroler
4	Sistem Hafalan Al-Quran Santri *Online* Pondok Pesantren Tahfizh Buton berbasis Web
5	Media Pembelajaran Pengembangan Nilai Agama dan Moral Pada PAUD berbasis Desktop

Kemudian menentukan kata kunci yang akan dihitung frekuensinya

Tabel 2. Contoh daftar kata kunci

No	Kata Kunci/Dasar
1	Aplikasi
2	Android
3	Web
4	Sistem
5	Rancang
6	Mikrokontroler
7	Desktop

Dan yang ketiga adalah tahap perhitungan *Term Frequency* (TF) yang dapat dituliskan vektor dari TF nya sebagai berikut:

- Judul 1 (1, 1, 0, 0, 0, 0, 0)
- Judul 2 (1, 1, 0, 0, 0, 0, 0)
- Judul 3 (0, 0, 0, 0, 1, 1, 0)
- Judul 4 (0, 0, 1, 1, 0, 0, 0)
- Judul 5 (0, 0, 0, 0, 0, 0, 1)

b. *Inverse Dokument Frequency* (IDF)

Inverse Dokument Frequency adalah sebuah perhitungan dari *term* didistribusikan secara luas pada koleksi dokumen yang bersangkutan. IDF menunjukkan hubungan ketersediaan sebuah *term* dalam seluruh dokumen. Semakin sedikit jumlah dokumen yang mengandung *term* yang dimaksud, maka nilai IDF semakin besar.

Rumus *inverse document frequency* sebagaimana ditulis pada rumus (2) berikut ini:

$$idf_t = log \frac{N}{df_t} \qquad (2)$$

N merupakan banyaknya dokumen pada sebuah corpus, sedangkan *fdt* merupakan total dokumen yang memuat kata *t*.

$$tfidf_{t,d} = tf_{t,d} \text{ x } idf_t \qquad (3)$$

Setelah mengenal idf, kita dapat mengkombinasikan idf dengan *term frequency* (TF). Kombinasi pembobotan ini kita notasikan dengan $tf\text{-}idf_{t,d}$ atau *term-frequency inverse document frequency* pada kata *t* di dokumen *d*. Rumus pembobotannya dapat dilihat pada rumus (3). Berdasarkan Tabel 1 maka cara memperoleh hasil *Invers Document Frequency* (IDF) adalah sebagai berikut:

IDFaplikasi = $log10$ (5/2)= 0, 39794
IDFandroid = $log10$ (5/2)= 0, 39794
IDFweb = $log10$ (5/2)=0,698970004
IDFsistem = $log10$(5/2)=0,698970004
IDFrancang = $log10$(5/2)= 0,698970004
IDFmikrokontroler = $log10$(5/2)=0,698970004
IDFdeskop = $log10$(5/2)=0,698970004

Fungsi dari text mining ini sendiri untuk memfilter setiap kata yang yang ada pada judul penelitian kemudian semua kata penting/kata kunci yang didapat selanjutnya akan dikonversi menjadi angka dengan metode pembobotan kata atau yang disebut *Term Frequency dan Invers Document Frequency* (TF-IDF). Proses perhitungan hingga mendapatkan nilai TD-IDF dari tabel V.1 dapat dilihat sebagai berikut:

1) Judul 1 TF-IDF = tf-idf (aplikasi) = 1 x 0, 39794 = 0,397940009
   = tf-idf (android) = 1 x 0, 39794 = 0,397940009
   = tf-idf (web) = 0 x 0,698970004= 0
   = tf-idf (sitem) = 0 x 0,698970004= 0
   = tf-idf (rancang) = 0 x 0,698970004= 0
   = tf-idf (mikrokontroler) = 0 x 0,698970004= 0
   = tf-idf (desktop) = 0 x 0,698970004 = 0
2) Judul 2 TF-IDF = tf-idf (aplikasi) = 1 x 0, 39794 = 0,397940009
   = tf-idf (android) = 1 x 0, 39794 = 0,397940009
   = tf-idf (web) = 0 x 0,698970004= 0
   = tf-idf (sitem) = 0 x 0,698970004= 0
   = tf-idf (rancang) = 0 x 0,698970004= 0
   = tf-idf (mikrokontroler) = 0 x 0,698970004= 0
   = tf-idf (desktop) = 0 x 0,698970004 = 0
3) Judul 1 TF-IDF = tf-idf (aplikasi) = 0 x 0, 39794 = 0
   = tf-idf (android) = 0 x 0, 39794 =

   = tf-idf (sitem) = 0 x 0,698970004= 0

   = tf-idf (web) = 0 x 0,698970004= 0

   = tf-idf (sitem) = 0 x 0,698970004= 0
   = tf-idf (rancang) = 1 x 0,698970004= 0,698970004
   = tf-idf (mikrokontroler) = 1 x 0,698970004= 0,698970004
   = tf-idf (desktop) = 0 x 0,698970004 = 0
4) Judul 2 TF-IDF = tf-idf (aplikasi) = 0 x 0, 39794 = 0
   = tf-idf (android) = 0 x 0, 39794 = 0
   = tf-idf (web) = 1 x 0,698970004= 0,698970004
   = tf-idf (sitem) = 1 x 0,698970004= 0,698970004

= tf-idf (rancang) = 0 x 0,698970004= 0
= tf-idf (mikrokontroler) = 0 x 0,698970004= 0
= tf-idf (desktop) = 0 x 0,698970004 = 0

5) Judul 2 TF-IDF = tf-idf (aplikasi) = 0 x 0, 39794 = 0
= tf-idf (android) = 0 x 0, 39794 = 0
= tf-idf (web) = 1 x 0,698970004= 0
= tf-idf (sitem) = 1 x 0,698970004= 0
= tf-idf (rancang) = 0 x 0,698970004= 0
= tf-idf (mikrokontroler) = 0 x 0,698970004= 0
= tf-idf (desktop) = 1 x 0,698970004 = 0,698970004

c. *K-Means Clustering*

Algoritma pengelompokan berbasis *K-Means* pertama untuk menghitung bobot fitur dirancang lebih dari 30 tahun yang lalu. Dengan sejarah lebih dari 50 tahun, *K-Means* bisa dibilang algoritma pengelompokan partisional paling populer yang pernah ada (de Amorim, 2016).

Clustering adalah tugas mensegmentasi kumpulan dokumen menjadi beberapa partisi di mana dokumen dalam kelompok yang sama lebih mirip satu sama lain daripada yang ada di kelompok/*cluster* lain (Allahyari et al., 2017). Dalam statistik dan pembelajaran mesin, *k-means clustering* adalah metode analisis cluster yang bertujuan untuk mempartisi observasi 'n' ke dalam cluster 'k' dimana setiap observasi termasuk dalam cluster dengan mean terdekat.

Tahap-tahap algoritma dasar *K-Means* seperti berikut:

1) Tentukan jumlah k sebagai *cluster* yang ingin dibentuk.
2) Menentukan pusat setiap data terhadap sebanyak k.
3) Menentukan jaraks setiap data terhadap pusat *cluster* (*centroid*).
4) Mengelompokkan setiap data yang bersangkutan bersadarkan kedekatannya dengan *centroid* (jarak terkecil).
5) Menentukan pusat cluster baru. Memperbaharui nilai *centroid* dari rata-rata *cluster* yang bersangkutan dengan menggunakan persamaan (4):

$$y_i(t+1) = \frac{1}{N_{sj}} \sum j \in sj\, x_j \qquad (4)$$

Algoritma biasanya sangat cepat, biasanya dijalankan beberapa kali dengan kondisi awal yang berbeda.

6) Ulangi langkah 3 hingga 5 sampai anggota yang ada pada setiap *cluster* tidak berubah.
7) Jika langkah 6 sudah tepenuhi, maka nilai pusat klaster pada perulangan terakhir akan digunakan sebagai parameter untuk kelompok dokumen penelitian.

d. Sumber Data

Penelitian ini menggunakan data sekunder yang bersumber dari dokumen penelitian yang diperoleh dari repositori Universitas Islam Negeri Alauddin Makassar. Jumlah data penelitian yang digunakan sebanyak 237 judul penelitian. Bagian penelitian yang diolah adalah judul dan abstrak dari penelitian.

e. NMI dan langkah-langkah pengujian

Penelitian ini menggunakan NMI untuk mengukur kualitas hasil klasterisasi judul penelitian. Pengujian NMI akan dilakukan setelah proses klasterisasi. Jumlah klaster atau

kelompok yang diuji adalah 3, 5 dan 10 klaster. Adapun langkah-langkah pengujiannya sebagai berikut:

1) Menghitung *entropy* dari *class*/kelompok. Persamaannya ditunjukkan pada persamaan (5) berikut:

$$\sum_{y \in Y} -P(Y = 1)\ log\ P(Y = 1) \qquad (5)$$

2) Menghitung *Entropy* dari label Klaster. Adapun persamaan *Entropy* dari label klaster ditunjukkan oleh persamaan (6) berikut ini:

$$\sum_{y \in Y} -P(C = 1) \log P(C = 1) \qquad (6)$$

3) Menghitung *Conditional entropy* dari *class label* untuk klasterisasi. Persamaan *Conditional entropy* dari *class label* sebagai berikut:

$$\mathrm{H(Y|C = 1)} = -\mathrm{P(= 1)} \sum_{\mathrm{y} \in \{1,2,3\}} \mathrm{P(Y = y|C = 1)} * \log(\mathrm{P(Y = y|C = 1)} \qquad (7)$$

4) Menghitung *Mutual Information* I(Y;C). Persamaan dari *Mutual informatio*-nya sebagai berikut:

$$\mathrm{I(Y;C) = H(Y) - H(Y \mid C)} \qquad (8)$$

Dan melanjutkan dengan mengitung hasil *Normalize Mutual Information*. Adapun persamaannya sebagai berikut:

$$\mathrm{NMI\ (Y, C) = \ (2 * \ I\ (Y; C))/([H(Y) + H(C)])} \qquad (9)$$

Dimana:

Y= *Class labels*; C= Klaster *labels*; H(.)= Entropy; dan I(Y;C)= *Mutual information*

HASIL DAN PEMBAHASAN

Penelitian ini menggunakan 237 data judul dan abstrak penelitian di jurusan Teknik Informatika UIN Alauddin Makassar, yang kemudian hasil dari klasterisasi tersebut diuji menggunakan metode *Normalize Mutual Information* (NMI). Pada kelompok yang diuji adalah untuk 3, 5 dan 10 klaster, dimana nilai NMI terbaik diraih pengelompokkan dengan 5 klaster. Sebagai langkah awal, *preprocessing* diterapkan sebagai metode pembobotan kata.

Pada pengujian ini ditampilkan *output* perhitungan dengan perolehan NMI tertinggi, yaitu dari pengelompokan 5 klaster, untuk langkah pengujian dan hasil dari eksperimen bisa dilihat sebagai berikut:

a) Menghitung *entropy* dari *class*/kelompok

Tabel 3. Hasil perhitungan *Entropy* dari *label Class*

1. Menghitung H(Y) = *ENTROPY OF CLASS LABEL*		
Nama class	Jumlah class/jumlah data	Hasil bagi (:)
P(Y=1) =	58/237 =	0,244725738
P(Y=2) =	73/237 =	0,308016878
P(Y=3) =	81/237 =	0,341772152
P(Y=4) =	25/237 =	0,105485232

H(Y) *ENTROPY OF CLASS LABEL* =	0,569528181

b. Menghitung *Entropy* dari *Cluster*. Tabel hasil perhitungan *Entropy* dari label *Cluster* ditunjukkan pada Tabel 4.

Tabel 4. Hasil perhitungan *Entropy* dari label *Cluster*

2. Menghitung H(C) = ENTROPY OF CLUSTER LABEL				
Cluster	jum. Data dalam cluster	jum. Data judul	Hasil bagi (:)	H(C)
C1 =	15	237	0,063291139	0,484592994
C2 =	28	237	0,11814346	
C3 =	10	237	0,042194093	
C4 =	152	237	0,641350211	
C5 =	32	237	0,135021097	
Jumlah	237			

Pada beberapa kelompok data yang diuji, pengujian 237 data judul pengelompokan menjadi 5 klaster menghasilkan rata-rata presentase nilai *entropy* klaster sebesar 0,569528181, untuk entropi *class*nya sebesar 0,466520823.

1) Menghitung *Conditional entropy* dari *label clasS.* Adapun hasil *Conditional entropy* dari *label class* sebagaimana yang tersaji pada Tabel 5.

Tabel 5. Hasil perhitungan *Conditional entropy of class label*

Menghitung H(Y|C): CONDITIONAL ENTROPY OF CLASS LABEL		
	Consider cluster 1	
Jumlah Data C	85	Hasil = 0,031565239
	Consider cluster 2	
Jumlah Data C	6	Hasil = 0,027800461
	Consider cluster 3	
Jumlah Data C	10	Hasil = 0,02592981
	Consider cluster 4	
Jumlah Data C	152	Hasil = 0,034066175
	Consider cluster 5	
Jumlah Data C	32	Hasil = 0,00642623

2) Menghitung *Mutual Information* I(Y;C). Hasil dari perhitungannya ditunjukkan pada Tabel 6:

Tabel 6. Hasil perhitungan *Mutual Information*

I (Y;C) Mutual Information
0,443740266

3) Menghitung hasil *Normalize Mutual Information*. Hasil perhitungan ditunjukkan pada Tabel 7.

Tabel 7. Hasil perhitungan *Mutual Information*

Normalize Mutual Information NMI
0,8419151

No.	judul	Berada di Cluster	kategori
49	Rancang Bangun Aplikasi Taksasi Tebu Berbasis Website Untuk Memprediksi Hasil Panen Tebu di Pabrik Gula Takalar	3	Teknologi Informasi
50	Rancang Bangun Sistem Pengendali Suhu, Keasaman dan Salinitas pada Tambak Ikan Kerapu Berbasis Mikrokontroler	3	Teknologi Informasi
51	Sistem Monitoring Kualitas Air Empang Berbasis Mikrokontroler	3	Teknologi Informasi
52	Aplikasi Tiket Elektronik Untuk pembayaran Bus Rapid Transit di Kota Makassar	3	Teknologi Informasi
53	Pengajuan Restitusi Barbasis Mobile di PT. PLN (PERSERO) UPB SULSELRABAR.	3	Teknologi Informasi
56	Rancang Bangun Sistem Penyortir Kualitas Telur Ayam Ras Berbasis Mikrokontroler	3	Teknologi Informasi
57	Monitoring Kendaraan Menggunakan Long Range (LoRa) Radio Frekuensi Berbasis Web.	3	Teknologi Informasi
58	Perancangan E-Marketplace Jasa Desain Grafis Berbasis Website	3	Teknologi Informasi
59	Rancang Bangun Kotak Amal Anti Maling menggunakan SMS Gateway Berbasis Mikrokontroler	3	Teknologi Informasi
60	Pengumuman Orang Hilang Penyandang Disabilitas Menggunakan Kode QR Berbasis Android	3	Teknologi Informasi

Gambar 1. Antarmuka *output* hasil klasterisasi sesuai kluster 3

No.	judul	Berada di Cluster	kategori
29	Efektivitas Ekstrak Daun Sidaguri (Sida Rhombifoli L) terhadap Kematian Larva Aedes aegypti	1	Kesehatan
14	Determinan Kesehatan Reproduksi Siswa di Pondok Pesantren Madrasah Aliyah (MA) Sultan Hasanuddin dan SMA Negeri 10 Gowa Kabupaten Gowa Tahun 2019	2	Kesehatan
32	Rancang Bangun Aplikasi Pembelajaran Hukum Tajwid metode ummi Berbasis Android	2	Teknologi Informasi
42	Implementasi Game Petualangan sebagai Penunjang Pembelajaran Interaktif Bahasa Jepang	2	Teknologi Informasi
46	Rancang Bangun Game Tradisional Berbasis Android	2	Teknologi Informasi
47	Rancang Bangun Game Interaktif Pembelajaran IPA Mengenai Rantai Makanan Untuk Anak SD (Sekolah Dasar) Berbasis Game Android	2	Teknologi Informasi
54	Sistem Hafalan Al-Qur'an Santri Online Pondok Pesantren Tahfizh Buton Berbasis WEB	2	Teknologi Informasi
55	Aplikasi Pendeteksi Plagiat Menggunakan Algoritma Cosine Similarity pada Jurnal INSTEK Berbasis Android	2	Teknologi Informasi
61	Isolasi bakteri yang mampu meresistensi merkuri di Kota Makassar	3	Biologi (Sains)
62	Aktivitas Antioksidan Zat Ekstraktif Bekatul Sorgum (Sorghum bicolor L.) Varietas Super 2 Secara In Vitro	3	Biologi (Sains)

Gambar 2. Antarmuka *output* hasil klasterisasi sesuai kluster 1, 2 dan 3

Sedangkan untuk nilai akhir NMI sebesar 0, 8419151, dimana angka tersebut mengindikasikan bahwa pengelompokkan yang dihasilkan cukup baik atau mendekati nilai korelasi informasi penuh yaitu 1, sementara NMI terendah diperoleh pada hasil pengelompokan 3 *cluster* dan 10 *cluster*, yaitu memperoleh nilai NMI 0.5 dan 0.6 berturut-turut. Walau demikan, bila dilihat dari hasil pengelompokkan masih ada sebagian kecil data yang tidak relevan, hal ini sebagaiman diduga sebelumnya, dapat dikarenakan oleh titik awal atau *centroidnya* dipilih secara acak, hal inilah yang juga mempunyai pengaruh terhadap hasil akhir klasterisasi.

Dari hasil analisis selama periode eksperimen juga diperoleh bahwa metode *text mining* dengan menggunakan pembobotan kata atau *term frequence* sebagai fitur akan menghasilkan dimensi vektor yang cukup besar sehingga terkadang membuat pengelompokkan menjadi kurang akurat. Jika *cluster* pilihan yang buruk dibuat di awal, banyak perubahan akan terjadi pada periode pengelompokan, dan dalam kasus ini untuk setiap kali hasil pengelompokan yang berbeda dapat diperoleh dengan jumlah klasterisasi yang sama. Pada saat yang sama, jika dimensi kelompok data berbeda, kepadatan kelompok data akan berbeda atau jika ada perbedaan data, algoritma mungkin tidak

mendapatkan hasil yang baik, dan penentuan *cluster* akhirnya diputuskan berdasarkan pada nilai NMI terbaik. Oleh karena itu, pemilihan *centroid* awal juga perlu menjadi pertimbangan untuk penelitian pada masa mendatang.

KESIMPULAN

Berdasarkan hasil eksperimen yang telah dilakukan, diperoleh kesimpulan bahwa *clustering* dokumen menggunakan *k-means clustering* dapat dilakukan pada judul penelitian. Sistem dapat mengelompokkan judul dengan menggunakan algoritma *k-means clustering* dan *text mining*. Judul penelitian dikelompokkan dengan bantuan *text mining* dan metode pembobotan kata pada awal pemrosesan. Dengan demikian dapat dikatakan bahwa Algoritma *K-means* clustering mampu mengelompokkan dokumen dalam jumlah yang banyak dengan kualitas informasi yang tinggi.

DAFTAR PUSTAKA

Allahyari, M., Pouriyeh, S., Assefi, M., Safaei, S., Trippe, E. D., Gutierrez, J. B., & Kochut, K. (2017). A Brief Survey of Text Mining: Classification, Clustering and Extraction Techniques. https://arxiv.org/abs/1707.02919.

de Amorim, R. C. (2016). A survey on feature weighting based k-means algorithms. *Journal of Classification*, 33(2), 210–242. https://doi.org/10.1007/s00357-016-9208-4.

Denny, M. J., & Spirling, A. (2018). Text preprocessing for unsupervised learning: why it matters, when it misleads, and what to do about it. *Political Analysis*, 26(2), 168–189. https://doi.org/10.1017/pan.2017.44.

Deolika, A., Kusrini, K., & Luthfi, E. T. (2019). Analisis pembobotan kata pada klasifikasi *text mining*. *Jurnal Teknologi Informasi*, 3(2), 179-194. https://doi.org/10.36294/jurti.v3i2.1077.

Februariyanti, H., & Santoso, D. B. (2017). Hierarchical agglomerative clustering untuk pengelompokan skripsi mahasiswa. *Pattern Recognition*, 11(5–6), 365–381.

Indraloka, D. S., & Santosa, B. (2017). Penerapan *text mining* untuk melakukan *clustering data tweet* Shopee Indonesia. *Jurnal Sains dan Seni ITS*, 6(2), A51–A56. https://doi.org/10.12962/j23373520.v6i2.24419.

Widodo, W., & Wahyuni, D. (2017). Implementasi algoritma *k-means clustering* untuk mengetahui bidang skripsi mahasiswa multimedia pendidikan teknik informatika dan komputer Universitas Negeri Jakarta. *PINTER: Jurnal Pendidikan Teknik Informatika dan Komputer*, 1(2), 157–166. https://doi.org/10.21009/pinter.1.2.10.

Yudiarta, N. G., Sudarma, M., & Ariastina, W. G. (2018). Penerapan metode *clustering text mining* untuk pengelompokan berita pada *unstructured textual* data. *Majalah Ilmiah Teknologi Elektro*, 17(3), 339-344. https://doi.org/10.24843/mite.2018.v17i03.p06.