BERKALA SAINSTEK 2022 10(4): 210-226
doi: 10.19184/bst.v10i4.29524
https://jurnal.unej.ac.id/index.php/BST/article/view/29524

RESEARCH ARTICLE

Implementation of K-Means Clustering Method for Trend Analysis of Thesis Topics (Case Study: Faculty of Computer Science, University of Jember)

(Implementasi Metode K-Means Clustering Untuk Analisis Trend Topik Skripsi (Studi Kasus: Fakultas Ilmu Komputer Universitas Jember)

Maulana Rafael Irianto, Achmad Maududie*, Fajrin Nurman Arifin
Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Jember,
Jln. Kalimantan 37, Jember 68121, Indonesia

ABSTRACT

The development of information technology causes a large number of digital documents, especially thesis documents, so that it can create opportunities for students to take the same and not varied topics. Thesis documents can be grouped by topic by identifying the abstract section. The results of the grouping can be seen with the trend with data visualization so that it can be analyzed to find out the trend of each topic. Retrieval of data in the repository of the University of Jember through a web scraping process as many as 490 thesis documents for students of the Faculty of Computer Science, University of Jember. The preprocessing stage is carried out by text mining methods which include cleaning, filtering, stemming, and tokenizing. Then calculate the weight of each word with the Term Frequency - Inverse Document Frequency algorithm, followed by the dimension reduction process using the Principal Component Analysis algorithm, which is normalized by Z-Score first. The outliers removal process is carried out before classifying documents. Furthermore, document grouping uses the K-Means Clustering method with Cosine Similarity as the distance calculation and the Silhouette Coefficient algorithm as a test. The test results were carried out with various k values and the optimal value was obtained at k = 2 with a Silhouette value of 0.80. Then the topic detection uses the Latent Dirichlet Allocation algorithm for each cluster that has been formed. Each cluster is visualized with a line chart and Trend Linear algorithm and analyzed to find out the trend. From the results of the analysis, it can be concluded that the topic of Decision Support System Development is trending down, and the topic of IT Performance Measurement and Forecasting is trending up. It can be concluded that the topic of Decision Support System Development needs to be reduced so that other topics can emerge.

Perkembangan teknologi informasi menyebabkan banyaknya jumlah dokumen *digital* khususnya dokumen skripsi sehingga dapat memunculkan peluang mahasiswa mengambil topik yang sama dan tidak variatif. Dokumen skripsi dapat dikelompokkan berdasarkan topiknya dengan mengidentifikasi bagian abstrak. Hasil pengelompokkan dapat diketahui *trend*-nya dengan visualisasi data sehingga dapat di analisis untuk mengetahui *trend* setiap topiknya. Pengambilan data pada *repository* Universitas Jember melalui proses *web scraping* sebanyak 490 dokumen skripsi mahasiswa Fakultas Ilmu Komputer Universitas Jember. Tahap *preprocessing* dilakukan dengan metode *text mining* yang meliputi *cleaning, filtering, stemming*, dan *tokenizing*. Lalu menghitung bobot setiap kata dengan algoritma *Term Frequency - Inverse Document Frequency*, dilanjutkan proses reduksi dimensi menggunakan algoritma *Principal Component Analysis* yang dilakukan normalisasi *Z-Score* terlebih dahulu. Proses outliers removal dilakukan sebelum mengelompokkan dokumen. Selanjutnya pengelompokkan dokumen menggunakan metode *K-Means Clustering* dengan *Cosine Similarity* sebagai perhitungan jarak dan algoritma *Silhouette Coefficient* sebagai pengujiannya. Hasil pengujian dilakukan dengan nilai k yang bervariasi dan didapatkan nilai optimal pada k = 2 dengan nilai *Silhouette* 0,80. Lalu pendeteksian topik menggunakan algoritma *Latent Dirichlet Allocation* pada setiap cluster yang telah terbentuk. Setiap *cluster* dilakukan visualisasi dengan *line chart* dan algoritma *Least Square* serta di analisis untuk mengetahui *trend* yang terjadi. Dari hasil analisis dapat disimpulkan bahwa topik Pengembangan Sistem Pendukung Keputusan terjadi *trend* turun, dan topik Pengukuran Kinerja TI dan Peramalan terjadi *trend* naik. Maka dapat disimpulkan topik Pengembangan Sistem Penunjang Keputusan perlu dikurangi sehingga topik-topik lain dapat muncul.

Keywords: Trend analysis, Thesis, K-Means Clustering, Trend Linear.

*) Corresponding author:
Achmad Maududie
E-mail: maududie@unej.ac.id

PENDAHULUAN

Perkembangan teknologi informasi semakin pesat dengan adanya internet menyebabkan banyaknya jumlah dokumen yang terdapat pada suatu *repository* instansi khususnya universitas. Berbagai dokumen karya tulis ilmiah dari civitas akademik seperti artikel penelitian mahasiswa, disertasi, *thesis*/skripsi dan lain-lain dapat diakses dalam bentuk *digital*. Jumlah pengetahuan yang dapat disarikan tidak sebanding dengan banyaknya dokumen tersebut [1].

Tugas akhir atau skripsi mahasiswa merupakan salah satu syarat kelulusan gelar sarjana. Penyusunan skripsi diperlukan mahasiswa dengan topik yang sesuai dengan jurusannya sehingga semakin lama, topik yang diambil semakin bervariasi dan mengikuti perkembangan zaman. Kumpulan dokumen skripsi dapat dikelompokkan untuk mengetahui topik-topik yang telah ada. Pengelompokkan dapat dilakukan dengan mengidentifikasi abstrak suatu dokumen. Bagian pada karya tulis yang dapat mengidentifikasi isinya tanpa perlu membaca keseluruhan bagiannya adalah abstrak [2]. Dengan membaca abstrak, maka pembaca dapat mengetahui isi dari karya tulis atau skripsi.

Penelitian ini melakukan pengelompokkan dokumen skripsi dari Fakultas Ilmu Komputer Universitas Jember karena jumlah dokumen skripsi semakin bertambah setiap tahunnya dengan topik yang beragam. Keberagaman topik ini perlu diketahui mahasiswa agar dapat mempermudah mahasiswa menentukan topik skripsinya. Semakin banyaknya dokumen skripsi memunculkan peluang mahasiswa yang akan mengambil skripsi dengan topik yang sama. Selama ini penelitian skripsi yang diseleksi oleh pembimbing hanya berdasarkan dari pengalaman mahasiswa yang telah dibimbing sebelumnya, sedangkan kesamaan dengan topik skripsi pada pembimbing lainnya tidak diketahui sehingga diperlukan pengelompokkan dokumen skripsi agar dapat di analisis untuk mengetahui topik-topik skripsi yang sering muncul. *Trend* dapat memperlihatkan topik skripsi yang banyak diminati pada waktu tertentu sehingga dapat mempermudah mahasiswa dalam menentukan topik yang berbeda dari penelitian sebelumnya. Dengan mengacu pada *trend* topik skripsi, juga dapat membantu pembimbing agar lebih variatif dalam menyetujui topik skripsi yang akan dilakukan mahasiswa.

Visualisasi data dibutuhkan untuk menyajikan data atau informasi berupa trend. Cara efektif untuk menyajikan data agar lebih mudah dipahami adalah dengan mengubahnya menjadi data visual [3]. Visualisasi data merupakan suatu proses mengubah data agar lebih mudah dipahami dengan cara mengubah bentuk menjadi grafik. Trend topik skripsi dapat dipahami dengan menerapkan visualisasi data.

Penelitian ini menggunakan data bertipe *text*, maka metode yang dapat digunakan dalam melakukan *preprocessing* yaitu *text mining*. *Text mining* merupakan pengembangan dari *data mining*. *Text mining* dapat digunakan untuk mengenali data yang bersifat semi terstruktur hingga tidak terstruktur misalnya abstrak suatu dokumen [4]. *Clustering* merupakan salah satu metode dari *text mining* yang digunakan untuk mengelompokkan objek atau data menjadi beberapa kelompok (*cluster*) sehingga data yang mirip akan berada pada satu cluster dan membuat jarak antar *cluster* sejauh mungkin.

Clustering terbagi menjadi dua jenis yaitu *cluster* hirarki dan *cluster* non-hirarki. Perbedaan keduanya terdapat pada jumlah cluster yang terbentuk. Pada *cluster non*-hirarki, diawali dengan penentuan jumlah cluster. Jenis cluster ini memiliki beberapa algoritma yang cukup terkenal yaitu *K-Means Clustering* dan *Fuzzy C-Means*, keduanya akan mengelompokkan data menjadi beberapa *cluster* sehingga data yang mirip akan dikelompokkan ke dalam *cluster* yang sama dan data yang tidak mirip akan dikelompokkan ke dalam *cluster* lain. Perbedaan keduanya terdapat pada cara pengelompokkan, algoritma *Fuzzy C-Means* mengelompokkan data dengan memperbolehkan satu data masuk ke dalam beberapa *cluster*. Sedangkan dalam algoritma *K-Means Clustering,* satu data hanya dapat masuk ke dalam satu *cluster*. *K-Means Clustering* merupakan algoritma yang biasa digunakan dalam permasalahan clustering karena kesederhanaannya dan memiliki tingkat ketelitian yang tinggi terhadap ukuran data sehingga *K-Means Clustering* lebih efisien dan terukur dalam pengolahan data berjumlah besar, selain itu urutan objek tidak mempengaruhi algoritma ini [5]. Data dokumen skripsi dapat dikelompokkan ke dalam kelompok-kelompok tertentu dengan menggunakan algoritma *K-Means Clustering* karena satu dokumen skripsi hanya dapat menjadi anggota satu *cluster* atau satu topik skripsi.

Penelitian terdahulu telah membahas mengenai permasalahan pengelompokkan dokumen dengan

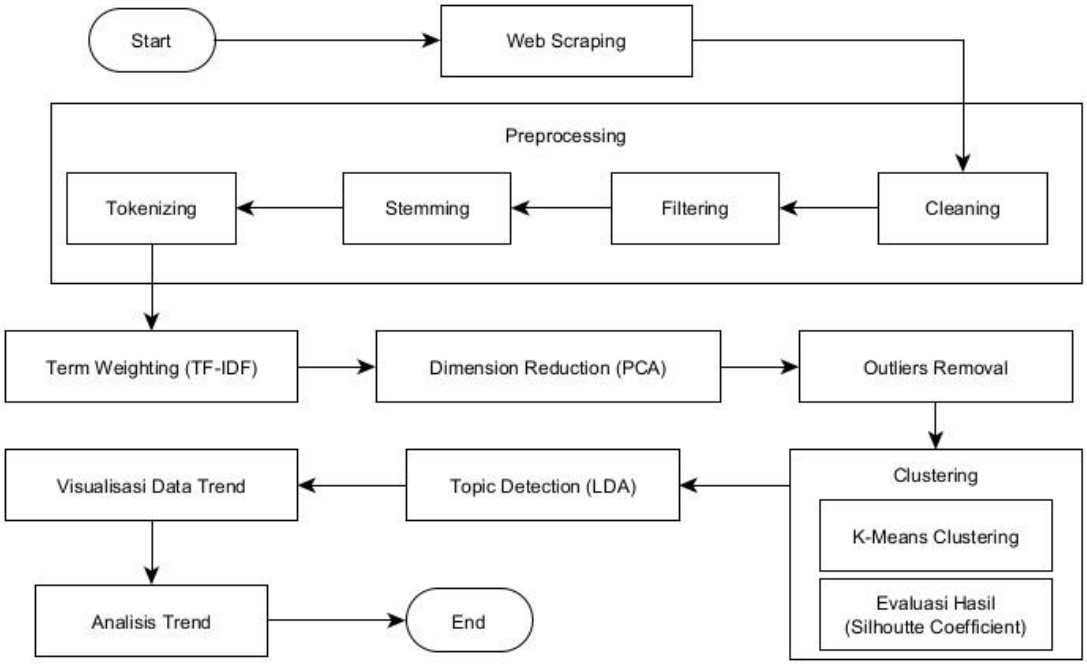
clustering. Misalnya penelitian pertama dengan menggunakan objek dokumen tugas akhir, serta metode *text mining* dan *K-Means* dalam mengelompokkan objek menghasilkan pengelompokkan dokumen tugas akhir dengan topik yang sesuai, seperti ditulis oleh Astuti, dkk [6]. Kontribusi penelitian pertama tersebut yaitu metode K-Means dapat digunakan untuk mengelompokkan dokumen tugas akhir berdasarkan topiknya dengan hasil yang baik, sehingga metode tersebut akan digunakan oleh peneliti. Penelitian kedua dengan menerapkan *text mining* dalam *preprocessing* dan *K-Means Clustering* dalam pengelompokkan dokumen skripsi serta algoritma *silhouette coefficient* dalam pengujiannya sehingga menghasilkan empat *cluster* data dengan akurasi 48%, seperti ditulis oleh Hudin, dkk [7]. Kontribusi penelitian kedua tersebut yaitu Algoritma Sillhouette Coefficient cukup baik untuk menguji pengelompokkan dokumen skripsi menggunakan K-Means Clustering dengan Cosine Similarity, sehingga algoritma tersebut akan digunakan oleh peneliti. Penelitian ketiga dengan menggunakan metode *Latent Dirichlet Allocation* dalam pemodelan topik pada dokumen skripsi menghasilkan topik telah sesuai dengan *trend* dan minat mahasiswa, seperti ditulis oleh Alfanzar, dkk [8]. Kontribusi penelitian ketiga tersebut yaitu algoritma Latent Dirichlet Allocation dapat memodelkan topik dengan baik, sehingga algoritma tersebut akan digunakan oleh peneliti. Dengan mengacu dari beberapa penelitian tersebut telah menjadi pertimbangan bahwa bagian dokumen skripsi yang digunakan berupa abstrak dan menggunakan *text mining* untuk *preprocessing*, serta metode *K-Means Clustering* dalam pengelompokkan dokumen skripsi.

Solusi untuk mengatasi permasalahan keberagaman topik yang tidak diketahui oleh mahasiswa maupun pembimbing yang memunculkan peluang topik skripsi yang diambil menjadi tidak beragam yaitu dengan dilakukan pengelompokkan dokumen skripsi berdasarkan topiknya sehingga akan membentuk kelompok-kelompok topik skripsi. Setelah dokumen skripsi dikelompokkan akan dilakukan visualisasi data berdasarkan periode waktu tahun untuk menganalisis *trend* topik skripsi yang berkembang sehingga dapat menjadi acuan mahasiswa dalam memilih topik skripsi yang akan diambil maupun pembimbing dalam menyetujui topik penelitian skripsi yang baru. Hal ini bertujuan agar penelitian skripsi dapat lebih variatif kedepannya sehingga dapat meningkatkan kreatifitas mahasiswa. Penelitian ini menggunakan dokumen *text* sebagai objek atau data, maka dibutuhkan *text mining* untuk melakukan *preprocessing* data. Begitu pula karena *K-Means Clustering* dikenal sebagai salah satu metode *clustering* yang terukur dan efisien, maka *K-Means Clustering* menjadi metode yang digunakan untuk pengelompokkan. Dengan menggunakan *text mining* dan *K-Means Clustering* diharapkan penelitian ini dapat mengelompokkan dokumen skripsi berdasarkan topiknya sehingga dapat dilakukan analisis *trend* untuk mengetahui *trend* topik skripsi. Oleh karena itu peneliti mengambil judul tentang "Implementasi Metode *K-Means Clustering* untuk Analisis *Trend* Topik Skripsi (Studi Kasus: Fakultas Ilmu Komputer Universitas Jember)".

METODE PENELITIAN

Penelitian ini menggunakan jenis penelitian kuantitatif karena akan mengolah data secara statistik seperti *machine learning* dan analisis *trend*. Pengolahan data akan menggunakan bahasa pemrograman *python* karena terdapat *library-library* untuk statistika.

Jenis data yang digunakan pada penelitian ini adalah data sekunder yaitu data dokumen skripsi mahasiswa strata satu Fakultas Ilmu Komputer Universitas Jember tahun 2014-2020. Data berjumlah 500 dokumen yang terdiri dari tiga program studi berbeda yaitu Sistem Informasi, Teknologi Informasi, dan Informatika. Sumber data diperoleh dari Repository Perpustakaan Universitas Jember dengan alamat https://repository.unej.ac.id/handle/123456789/175. Tahapan yang dilakukan pada penelitian ini digambarkan melalui Gambar 1 sebagai berikut.



Gambar 1. Tahapan Penelitian

Web Scraping

Tahap pengumpulan data (*web scraping*) dokumen skripsi dari sumber data yaitu *repository* universitas jember, Data yang diambil berupa judul, abstrak, dan tahun.

Preprocessing

Preprocessing merupakan tahap awal dalam pengolahan data. Karena data yang digunakan berupa dokumen maka tahap *preprocessing* sangat dibutuhkan untuk pengolahan data selanjutnya. *Cleaning* merupakan tahap menghapus data kosong atau data yang tidak dapat digunakan seperti isi data yang tidak sesuai. Tahap ini juga menghilangkan tanda baca, angka, dan seluruh karakter selain alphabet serta mengubah seluruh huruf menjadi *lowercase*.

Filtering atau biasa disebut *stopword* merupakan proses menyaring atau membuang kata-kata yang tidak dapat merepresentasikan isi dari dokumen dengan berdasarkan pada kamus *stopword*. Kamus *stopword* yang digunakan dalam penelitian ini berasal dari kamus Sastrawi. *Stemming* merupakan proses untuk mengambil kata dasar dari hasil *filtering* dengan dilakukan penghilangan kata imbuhan awalan *(prefix)* dan akhiran (*surfix*) sehingga menghasilkan kata dasar dari setiap kata. *Tokenizing* bertujuan untuk memecah dokumen berdasarkan setiap kata sehingga hasilnya berupa kumpulan kata pada setiap dokumen.

Algoritma Term Frequency-Inverse Document Frequency (TF-IDF)

Term weighting adalah proses pembobotan kata sebelum dilakukan *clustering*. TF-IDF merupakan algoritma pada text mining yang biasa digunakan untuk pembobotan kata [9]. Pembobotan kata dengan TF-IDF terdiri dari beberapa proses yaitu seperti berikut.

1. *Term frequency* yaitu menghitung frekuensi kemunculan sebuah term (kata/frasa) dalam suatu dokumen. Semakin banyak suatu term muncul pada dokumen, maka akan semakin besar bobot katanya.
2. *Term weighting* yaitu menghitung bobot dari setiap term pada term *frequency*. Dalam menghitung bobot term *frequency*, dapat menggunakan Persamaan (1) sebagai berikut.

$$W_{t,f} = \begin{cases} 1 + \log_{10} tf_{t,d}, & if\ tf_{t,d} > 0 \\ 0, & otherwise \end{cases} \quad \text{..... (1)}$$

3. *Inverse Document Frequency* yaitu suatu dokumen yang mengandung *term*. Dalam menghitung IDF, dapat menggunakan Persamaan (2) sebagai berikut.

$$idf_t = \log_{10} N/df_t \quad \text{..... (2)}$$

Keterangan :

N : Jumlah seluruh dokumen

df_t : *Frequency* dokumen yang mengandung *term*

4. TF-IDF yaitu hasil perkalian dari TF dan IDF. Dalam menghitung TF-IDF, dapat menggunakan Persamaan (3) sebagai berikut.

$$W_{t,d} = W_{t,f} \times idf_t \quad \text{..... (3)}$$

Keterangan :
N : Jumlah seluruh dokumen
$W_{t,f}$: Bobot *term* pada dokumen
df_t : Frequency dokumen yang memuat $term$

Algoritma *Principal Component Analysis* (PCA)

PCA adalah algoritma untuk mereduksi dimensi suatu data mutivariat menjadi dimensi data yang lebih sedikit, namun tetap mempertahankan jumlah variannya. Tujuan PCA untuk menemukan suatu kelompok dimensi data dengan variasi yang lebih baik. Hasil dari PCA berupa kombinasi linear dari atribut-atribut yang lebih sedikit dari data yang berdimensi lebih banyak, dalam hal ini dimensi data dikurangi agar mendapat pengelompokkan yang optimal berdasarkan atribut dan relasi yang kuat. PCA dapat menjelaskan seluruh varian data menggunakan dimensi yang lebih sedikit.

Normalisasi merupakan proses mengubah nilai numerik suatu data agar memiliki nilai yang seragam. Normalisasi yaitu tabel yang menunjukkan entitas dan relasi dari pengelompokkan elemen data. Normalisasi *Z-Score* digunakan apabila nilai maksimum dan minimum suatu data tidak diketahui. *Z-Score* mengubah seluruh nilai dari suatu data dengan mempertahankan variasinya. Sebelum melakukan *dimension reduction* perlu dilakukan normalisasi data menggunakan algoritma *Z-Score*. Hal ini bertujuan agar seluruh data memiliki nilai yang seragam. Normalisasi perlu dilakukan agar meningkatkan performa dari *dimension reduction* agar hasilnya lebih maksimal. Rumus dari *Z-Score* menggunakan Persamaan (4) sebagai berikut.

$$v' = (v - \bar{A})/\sigma_A \quad \text{..... (4)}$$

Keterangan :
v' : Nilai baru
v : Nilai lama
$\bar{A}$: Rata-rata atribut A
σ_A : Standar deviasi atribut A

Selanjutnya proses *dimension reduction* dengan algoritma *Principal Component Analysis* (PCA). Tahapan reduksi dimensi menggunakan PCA yaitu sebagai berikut.

1. Menghitung matriks D yang merupakan hasil dari total matriks X_{ij} dikurangi rata-rata matriks X. Rumus menghitung matriks D menggunakan Persamaan (5) sebagai berikut.

$$m' = X_{ij} - \bar{X}_J \quad \text{..... (5)}$$

2. Menghitung matriks C_x yang merupakan hasil perhitungan *covariance* matriks D. Rumus menghitung matrix C_x menggunakan Persamaan (6) sebagai berikut.

$$C_x = \frac{1}{M} X^T X \quad \text{..... (6)}$$

3. Menghitung eigenvalue dan eigenvector matriks C_x
4. Memilih nilai eigenvalue dan eigenvector terbesar
5. Membuat dataset baru berdasarkan penentuan panjang atribut.

Sebelum dilakukan clustering, perlu mencari data *outliers* terlebih dahulu agar hasil clustering lebih optimal. Data *outliers* sangat berpengaruh buruk terhadap suatu *clustering*. *Outliers* adalah data atau objek yang muncul dengan nilai yang ekstrim. Ciri dari sebuah data dikategorikan sebagai *outliers* yaitu data tersebut memiliki nilai yang jauh berbeda dengan sebagian besar nilai pada data lainnya. Data *outliers* dapat mengakibatkan proses klasifikasi atau *clustering* menjadi kurang baik. Cara mengatasi data *outliers* dapat dengan menghapusnya. Umumnya *outliers* dapat diketahui menggunakan metric IQR (*interquartile range*). Berikut langkah-langkah tahap *outliers removal*.

1. Menghitung nilai rata-rata setiap data.
2. Menentukan nilai $Q3$ (kuartil 3) dan nilai $Q1$ (kuartil 1).
3. Menghitung nilai IQR (interquantile range) yaitu selisih $Q3$ dan $Q1$. Rumus dari IQR menggunakan Persamaan (7) sebagai berikut.

$$IQR = Q3 - Q1 \quad \text{..... (7)}$$

Keterangan :
IQR : Nilai *interquartile range*
$Q3$: Nilai kuartil 3
$Q1$: Nilai kuartil 1

4. Menghapus data yang memiliki nilai sesuai persyaratan tersebut.

Metode *K-Means Clustering*

K-Means Clustering merupakan metode analisis data dengan pemodelan *unsupervised* yang melakukan pengelompokkan data ke dalam beberapa kelompok (*cluster*). Kelompok-kelompok tersebut membentuk karakteristik sehingga antar kelompok memiliki karakteristik berbeda. Tujuan *K-Means Clustering* adalah meminimalkan kesamaan karakteristik antar kelompok dan memaksimalkan kesamaan karakteristik pada satu kelompok. Menurut Han & Kamber (2006), *K-Means Clustering* mengelompokkan data menjadi *k* buah kelompok yang telah ditentukan. Metode ini mengelompokkan data dengan karakteristik sama ke dalam satu kelompok dan data dengan karakteristik berbeda dikelompokkan ke dalam kelompok berbeda.

Langkah pertama yaitu menentukan *k* buah *cluster* (kelompok) yang akan dibentuk. Lalu menentukan centroid (pusat *cluster*) secara *random* sebanyak *k*. Selanjutnya menghitung jarak setiap data terhadap *centroid* dengan algoritma *Cosine Similarity*. *Cosine similarity* adalah fungsi untuk menghitung jarak atau derajat kemiripan antara dua vektor dengan melihat sudut yang terbentuk. *Cosine similarity* dapat digunakan untuk menghitung derajat kemiripan antar dokumen dengan bantuan *centroid*. Dalam menghitung *cosine similarity* dengan menghitung langsung menggunakan Persamaan (8) sebagai berikut.

$$CosSim(d_j, q) = \frac{\vec{d_j}.\vec{q}}{\left|\vec{d_j}\right|.\left|\vec{q}\right|} = \frac{\sum_{i=1}^{i}(W_{ij}.W_{iq})}{\sqrt{\sum_{i=1}^{t} W_{ij^2}.\sum_{i=1}^{t} W_{iq^2}}} \quad \text{…. (8)}$$

Keterangan :

d_j, q : Jarak data j dengan data q

W_{ij} : Nilai data ij

W_{iq} : Nilai data iq

Lalu mengelompokkan setiap data yang berjarak minimum terhadap *centroid*. Ulangi langkah perhitungan jarak hingga data tiap *cluster* tidak berubah. Jika *cluster* tidak berubah, maka pengelompokkan selesai. Pengujian dilakukan dengan menerapkan *clustering* pada nilai k yang bervariasi sebanyak jumlah kemungkinan topik skripsi yang terbentuk. Nilai *k* yang digunakan antara lain 2, 3, 4, 5, 6, 7, 8, 9, dan 10. Nilai *k* yang optimal akan dipilih menjadi model yang digunakan pada tahap berikutnya.

Algoritma *Silhouette Coefficient*

Silhoutte coefficient adalah algoritma untuk menguji kekuatan dan kualitas dari *cluster* (kelompok) dengan menerapkan *cohesion* dan *separation*. *Cohesion* digunakan untuk menghitung jarak antar data dalam suatu *cluster*, sedangkan *separation* berfungsi untuk menghitung jarak antar *cluster*. Berikut tahapan pengujian *cluster* menggunakan *silhouette coefficient* [10].

1. Menghitung rata-rata jarak data dengan semua data lain pada suatu *cluster* menggunakan Persamaan (9) sebagai berikut.

   $$a(i) = \frac{1}{[A]-1}\sum j \in_{A, j \neq i} \; d(i,j) \text{……. (9)}$$

   Keterangan :

   $a(i)$: Nilai rata-rata jarak data dengan semua data lain pada suatu *cluster*

   $[A]$: Jumlah data pada suatu *cluster*

   $d(i,j)$: Jarak data dengan semua data lain pada suatu *cluster*

2. Menghitung rata-rata jarak data dengan semua data lain pada *cluster* lain menggunakan Persamaan (10) sebagai berikut.

   $$b(i) = \frac{1}{[A]}\sum j \in C \; d(i,j) \text{ …….... (10)}$$

   Keterangan :

   $b(i)$: Nilai rata-rata jarak data dengan semua data lain pada *cluster* lain

   $[A]$: Jumlah data pada suatu *cluster*

   $d(i,j)$: Jarak data dengan semua data lain pada *cluster* lain

3. Menghitung *silhouette coefficient* menggunakan persamaan (11) sebagai berikut.

   $$s(i) = \frac{b(i)-a(i)}{max(a(i),b(i))} \text{ ……. (11)}$$

   Keterangan :

   $s(i)$: Nilai *silhouette coefficient*

   $a(i)$: Nilai rata-rata jarak data dengan semua data lain suatu *cluster*

   $b(i)$: Nilai rata-rata jarak data dengan semua data lain pada *cluster* lain

   $max(a(i), b(i))$: Nilai terbesar antara $a(i)$ dan $b(i)$

Silhouette coefficient mempunyai nilai bervariasi dari -1 hingga 1. Nilai *silhouette coefficient* dikategorikan baik jika bernilai positif saat nilai a(i) < b(i) dan nilai a(i) mendekati 0. Apabila nilai a(i) mendekati 0 maka akan mendapatkan nilai *silhouette coefficient* maksimum yaitu mendekati 1. Jika nilai s(i) = 1 maka data i berada di *cluster* yang benar, sedangkan jika nilai s(i) = 0 maka data i tidak jelas keberadaannya diantara dua *cluster*.

Apabila nilai s(i) = -1 maka data i berada di *cluster* yang salah karena strukturnya *overlapping*. Nilai *silhouette coefficient* dapat dijadikan ukuran yang menggambarkan seberapa ketat pengelompokkan data pada suatu *cluster*, ukuran tersebut dibagi ke beberapa golongan sebagai berikut [11].

$0{,}7 < s(i) < 1$, *Strong Structure*
$0{,}5 < s(i) < 0{,}7$, *Medium Structure*
$0{,}25 < s(i) < 0{,}5$, *Weak Structure*
$s(i) < 0{,}25$, *No Structure*

Algoritma *Latent Dirichlet Allocation* (LDA)

Latent Dirichlet Allocation (LDA) adalah model probabilistik pemodelan topik pada data teks. Model pada LDA digunakan untuk mengidentifikasi informasi tersembunyi pada beberapa dokumen. Model ini dapat memodelkan teks berdasarkan probabilitas kata. Deteksi topik dilakukan dengan menghitung probabilitas kemunculan setiap kata dari suatu kumpulan dokumen pada setiap *cluster* dan mengambil peringkat tertinggi dari kata-kata tersebut. Hasil deteksi topik tersebut dapat digunakan dalam menentukan topik yang sesuai untuk dijadikan sebagai label dataset. Model ini dapat memodelkan teks berdasarkan probabilitas kata. LDA merupakan salah satu algoritma pemodelan topik yang paling popular [12].

Tahapan proses pemodelan topik LDA dimulai dengan mengubah bentuk dokumen menjadi bentuk *dictionary*, lalu mengubahnya ke bentuk *corpus* dan membuat model topik LDA. Variabel input yang dibutukan untuk membuat model LDA yaitu jumlah iterasi, jumlah topik, dan jumlah kata pada setiap topik [13].

Algoritma *Trend Linear*

Trend linear merupakan *trend* data dengan penurunan atau kenaikan secara *linear* dengan variabel bebas berupa waktu tahun, bulan, minggu, atau hari. Jenis waktu dapat disesuaikan kebutuhan suatu model. Rumus garis *trend linear* menggunakan Persamaan (12) sebagai berikut.

$$Y = a + bX \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (12)$$

Keterangan
Y : data berkala
X : waktu
a, b : bilangan konstan

Nilai bilangan konstan a dan b dari persamaan *trend linear* dihitung dengan Persamaan (16) dan Persamaan (17) sebagai berikut.

$$a = \frac{\sum_{1=i}^{n} Y_i}{n} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (13)$$

$$b = \frac{\sum_{1=i}^{n} X_i Y_i}{\sum_{1=i}^{n} X_i^2} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (14)$$

Keterangan :
Y : data berkala
n : jumlah periode waktu
X : waktu

Variabel waktu (X) dibutuhkan dalam melakukan perhitungan. Nilai variabel waktu dibagi menjadi data ganjil dan genap. Berikut nilai untuk masing-masing variabel waktu.

a. Jumlah periode waktu ganjil, nilai X: ...., -3, -2, -1, 0 + 1, 2, 3,...
b. Jumlah periode waktu genap, nilai X: ...., -5, -3, -1,+ 1, 3, 5,...

Dalam visualisasi data *trend,* digunakan *line chart* karena dapat menyajikan kumpulan data selama periode waktu tertentu. Visualisasi dilakukan pada setiap topik skripsi dengan periode tahun sebagai sumbu *Y* dan jumlah dokumen skripsi sebagai sumbu *X*. Perhitungan metode *trend* menggunakan *trend linear* atau biasa disebut juga metode *least square* karena untuk menentukan *trend* naik atau turun. Berikut langkah-langkah penggunaan metode *least square*.

1. Menyiapkan data yang akan digunakan.
2. Menentukan nilai X yang diperlukan sebagai nilai variabel waktu. Jumlah dari nilai variabel waktu adalah nol. Terdapat dua jenis waktu yang dapat digunakan yaitu sebagai berikut.
   a. Periode waktu ganjil. Terdapat beberapa kriteria sebagai berikut.
      - Jarak antara dua waktu diberi nilai satu satuan.
      - Diatas nilai 0 diberi tanda negatif $(-)$.
      - Dibawah nilai 0 diberi tanda positif $(+)$.
   b. Periode waktu genap. Terdapat beberapa kriteria sebagai berikut.
      - Jarak antara dua waktu diberi nilai dua satuan.
      - Diatas nilai 0 diberi tanda negatif $(-)$.
      - Dibawah nilai 0 diberi tanda positif $(+)$.

3. Menentukan nilai XY dengan cara mengalikan nilai variabel Y yaitu jumlah dokumen skripsi dengan nilai variabel X.
4. Menentukan nilai X^2 dengan cara mengkuadratkan nilai X.
5. Menentukan nilai Y^2 dengan cara mengkuadratkan nilai Y.
6. Menentukan model trend

Selanjutnya melakukan analisis *trend* berdasarkan visualisasi data yang telah terbentuk. Analisis *trend* bertujuan untuk menentukan *trend* yang terjadi. Analisis *trend* dilakukan dengan melihat persamaan garis *trend*. Hasil dari analisis akan menunjukkan *trend* topik skripsi tertentu naik atau turun serta jumlah dokumen setiap topiknya

HASIL DAN PEMBAHASAN

Dataset yang telah disusun memiliki beberapa fitur atau kolom seperti judul, tahun, abstract. Contoh hasil web scraping dapat dilihat pada Gambar 2. Tahap awal proses preprocessing dilakukan pembersihan dataset dengan menghapus data kosong, data duplicate, angka, whitespase, dan karakter selain alphabet serta mengubahnya seluruh huruf menjadi lowercase. Contoh hasil cleaning dapat dilihat pada Tabel 1.

	Judul	Tahun	Abstract
0	SISTEM PAKAR IDENTIFIKASI HAMA DAN PENYAKIT TE...	2014	Hasil wawancara yang dilakukan di Pusat Peneli...
1	SISTEM PENDUKUNG KEPUTUSAN PENENTUAN PRODUK AI...	2015	Perusahaan air minum dalam kemasan (AMDK) meng...
2	Sistem Pendukung Keputusan Seleksi Pemain Bask...	2018	Penerepan metode Simple Additive Weighting dal...
3	Evaluasi User Experience Sister For Students m...	2019	Sister For Students (SFS) merupakan aplikasi b...
4	Penerapan Metode Simple Multi Attribute Rating...	2020	Sistem pendukung keputusan pemilihan game onli...
5	Evaluasi User Experience pada Aplikasi Interne...	2020	Internet banking Bank Rakyat Indonesia (BRI) m...

Gambar 2. Contoh Hasil *Web Scraping*

Tahap awal proses preprocessing dilakukan pembersihan dataset dengan menghapus data kosong, data duplicate, angka, whitespase, dan karakter selain alphabet serta mengubahnya seluruh huruf menjadi lowercase. Contoh hasil cleaning dapat dilihat pada Tabel 1.

Tabel 1. Contoh Hasil *Cleaning*

Abstract	*Cleaning*
Hasil wawancara yang dilakukan di Pusat Penelitian Tembakau PT. Perkebunan Nusantara X Jember diketahui bahwa kendala terbesar dalam budidaya tanaman tembakau yaitu penanganan hama dan penyakit, untuk mengatasi masalah tersebut dibangunlah Sistem Pakar Identifikasi Hama dan Penyakit Tembakau Berbasis Web GIS…	hasil wawancara yang dilakukan di pusat penelitian tembakau pt perkebunan nusantara x jember diketahui bahwa kendala terbesar dalam budidaya tanaman tembakau yaitu penanganan hama dan penyakit untuk mengatasi masalah tersebut dibangunlah sistem pakar identifikasi hama dan penyakit tembakau berbasis web gis…
Perusahaan air minum dalam kemasan (AMDK) mengalami hal yang sama yaitu permasalahan dalam menentukan desain kemasan produk yang digunakan. Jumlah merek dan variasi produk air minum dalam kemasan yang banyak menimbulkan persaingan yang sangat ketat di pasaran….	perusahaan air minum dalam kemasan amdk mengalami hal yang sama yaitu permasalahan dalam menentukan desain kemasan produk yang digunakan jumlah merek dan variasi produk air minum dalam kemasan yang banyak menimbulkan persaingan yang sangat ketat di pasaran…

Selanjutnya tahap filtering yaitu menghapus kata yang tidak bermakna agar dataset hanya terdapat kata-kata bermakna. Penghapusan kata yang tidak bermakna berdasarkan kamus stopword dari Sastrawi. Contoh hasil filtering dapat dilihat pada Tabel 2.

Tabel 2. Contoh Hasil *Filtering*

Cleaning	*Filtering*
hasil wawancara yang dilakukan di pusat penelitian tembakau pt perkebunan nusantara x jember diketahui bahwa kendala terbesar dalam budidaya tanaman tembakau yaitu penanganan hama dan penyakit untuk mengatasi masalah tersebut dibangunlah sistem pakar identifikasi hama dan penyakit tembakau berbasis web gis…	hasil wawancara pusat penelitian tembakau pt perkebunan nusantara jember kendala terbesar budidaya tanaman tembakau penanganan hama penyakit mengatasi dibangunlah sistem pakar identifikasi hama penyakit tembakau berbasis web gis…
perusahaan air minum dalam kemasan amdk mengalami hal yang sama yaitu permasalahan dalam menentukan desain kemasan produk yang digunakan jumlah merek dan variasi produk air minum dalam kemasan yang banyak menimbulkan persaingan yang sangat ketat di pasaran…	perusahaan air minum kemasan amdk mengalami permasalahan menentukan desain kemasan produk merek variasi produk air minum kemasan menimbulkan persaingan ketat pasaran…

Setelah dilakukan filtering, dilanjutkan tahap stemming. Implementasi tahap stemming dengan mengubah kata imbuhan menjadi kata dasar. Contoh hasil stemming dapat dilihat pada Tabel 3.

Tabel 3. Contoh Hasil *Stemming*

Filtering	*Stemming*
hasil wawancara pusat penelitian tembakau pt perkebunan nusantara jember kendala terbesar budidaya tanaman tembakau penanganan hama penyakit mengatasi dibangunlah sistem pakar identifikasi hama penyakit tembakau berbasis web gis…	hasil wawancara pusat teliti tembakau pt kebun nusantara jember kendala besar budidaya tanam tembakau tangan hama sakit atas bangun sistem pakar identifikasi hama sakit tembakau bas web gis…
perusahaan air minum kemasan amdk mengalami permasalahan menentukan desain kemasan produk merek variasi produk air minum kemasan menimbulkan persaingan ketat pasaran…	usaha air minum kemas amdk alami masalah tentu desain kemas produk merek variasi produk air minum kemas timbul saing ketat pasar…

Tahap akhir proses preprocessing yaitu tokenizing. Implementasi tahap tokenizing dengan cara mengubah kalimat menjadi kumpulan kata. Contoh hasil tokenizing dapat dilihat pada Tabel 4.

Tabel 4. Contoh Hasil *Tokenizing*

Stemming	*Tokenizing*
hasil wawancara pusat teliti tembakau pt kebun nusantara jember kendala besar budidaya tanam tembakau tangan hama sakit atas bangun sistem pakar identifikasi hama sakit tembakau bas web gis…	['hasil', 'wawancara', 'pusat', 'teliti', 'tembakau', 'pt', 'kebun', 'nusantara', 'jember', 'kendala', 'besar', 'budidaya', 'tanam', 'tembakau', 'tangan', 'hama', 'sakit', 'atas', 'bangun', 'sistem', 'pakar', 'identifikasi', 'hama', 'sakit', 'tembakau', 'bas', 'web', 'gis',…]
usaha air minum kemas amdk alami masalah tentu desain kemas produk merek variasi produk air minum kemas timbul saing ketat pasar…	['usaha', 'air', 'minum', 'kemas', 'amdk', 'alami', 'masalah', 'tentu', 'desain', 'kemas', 'produk', 'merek', 'variasi', 'produk', 'air', 'minum', 'kemas', 'timbul', 'saing', 'ketat', 'pasar',…]

Pembobotan kata menggunakan algoritma term frequency–inverse document frequency (TF-IDF) diawali menghitung kemunculan setiap kata pada suatu dokumen atau lebih dikenal term frequency (TF). Contoh hasil term frequency (TF) dapat dilihat pada Tabel 5.

Tabel 5. Contoh Hasil *Term Frequency* (TF)

Tokenizing	*Term Frequency (TF)*	
['hasil', 'wawancara', 'pusat', 'teliti', 'tembakau', 'pt', 'kebun', 'nusantara', 'jember', 'kendala', 'besar', 'budidaya', 'tanam', 'tembakau', 'tangan', 'hama', 'sakit', 'atas', 'bangun', 'sistem', 'pakar', 'identifikasi', 'hama', 'sakit', 'tembakau', 'bas', 'web', 'gis',…]	Kata	Nilai TF
	hasil	6
	wawancara	1
	pusat	1
	teliti	2
	tembakau	5
	pt	2

	kebun	2
	nusantara	2
	jember	1
	kendala	1
	besar	1
	budidaya	1
	tanam	2
	tembakau	5
	tangan	1
	hama	8
	sakit	12
	atas	1
	bangun	1
	sistem	3
	pakar	2
	identifikasi	2
	bas	1
	web	1
	gis	1
['usaha', 'air', 'minum', 'kemas', 'amdk', 'alami', 'masalah', 'tentu', 'desain', 'kemas', 'produk', 'merek', 'variasi', 'produk', 'air', 'minum', 'kemas', 'timbul', 'saing', 'ketat', 'pasar',…]	Kata	Nilai TF
	usaha	1
	air	9
	minum	9
	kemas	15
	amdk	1
	alami	1
	masalah	2
	tentu	4
	desain	8
	kemas	15
	produk	17
	merek	1
	variasi	1
	timbul	1
	saing	2
	ketat	1
	pasar	4

Setelah mendapatkan nilai TF, dilanjutkan menghitung nilai inverse document frequency (IDF). Contoh hasil inverse document frequency (IDF) dapat dilihat pada Tabel 6.

Tabel 6. Contoh Hasil *Inverse Document Frequency* (IDF)

No	Kata	Nilai DF	Nilai IDF
1	hasil	343	0,154902
	wawancara	51	0,982625904
	pusat	19	1,411442
	teliti	288	0,230804
	tembakau	12	1,611014834
	pt	35	1,146128
	kebun	12	1,611015
	nusantara	8	1,787106093
	jember	188	0,416038
	kendala	29	1,2277981
	besar	122	0,603836
	budidaya	10	1,69019608
	tanam	23	1,328468
	tembakau	12	1,611014834
	tangan	43	1,056728
	hama	5	1,991226
	sakit	33	1,171682
	atas	85	0,760777
	bangun	166	0,470088
	sistem	368	0,124348
	pakar	21	1,367977
	identifikasi	31	1,198834386
	bas	121	0,607411
	web	66	0,870652
	gis	1	2,690196
2	usaha	137	0,553476
	air	15	1,514105
	minum	11	1,648803
	kemas	4	2,088136
	amdk	2	2,389166
	alami	80	0,787106
	masalah	172	0,4546676

	tentu	223	0,341891
	desain	49	1
	kemas	4	2,088136
	produk	66	0,870652
	merek	2	2,389166
	variasi	6	1,912045
	timbul	21	1,367977
	saing	25	1,292256
	ketat	15	1,514105
	pasar	44	1,046743

Nilai TF dan IDF telah diketahui, selanjutnya menghitung nilai *term frequency–inverse document frequency* (TF-IDF). Contoh hasil *term frequency-inverse document frequency* (TF-IDF) dapat dilihat pada Tabel 7.

Tabel 7. Contoh Hasil *Term Frequency - Inverse Document Frequency* (TF-IDF)

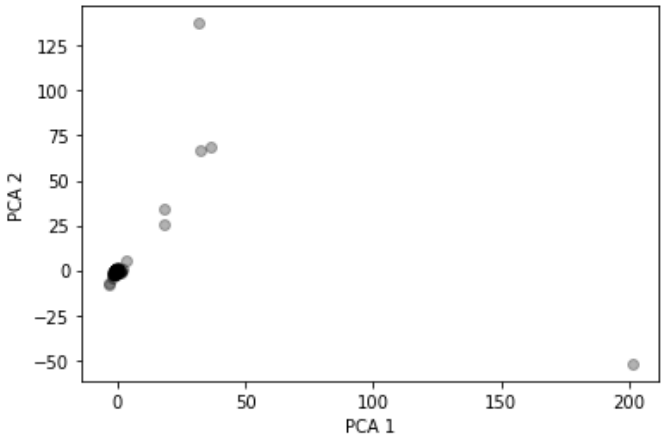
No	Kata	Nilai TF-IDF
1	hasil	0,93
	wawancara	0,983
	pusat	1,411
	teliti	0,462
	tembakau	8,055
	pt	2,292
	kebun	3,222
	nusantara	3,574
	jember	0,416
	kendala	1,228
	besar	0,604
	budidaya	1,69
	tanam	2,656
	tembakau	8,055
	tangan	1,057
	hama	15,928
	sakit	14,064
	atas	0,761
	bangun	0,47
	sistem	0,372
	pakar	2,736
	identifikasi	2,398
	bas	0,607
	web	0,871
	gis	2,69
2	usaha	0,553
	air	13,626
	minum	14,841
	kemas	31,32
	amdk	2,389
	alami	0,787
	masalah	0,91
	tentu	1,368
	desain	8
	kemas	31,32
	produk	14,807
	merek	2,389
	variasi	1,912
	timbul	1,368
	saing	2,584
	ketat	1,514
	pasar	4,188

Dimension reduction berfungsi mengurangi dimensi dari data dengan menjaga variasinya. Hal ini dapat meningkatkan performa dari suatu pengelompokkan atau *clustering*. Berikut contoh hasil *dimension reduction* dapat dilihat pada Tabel 8.

Tabel 8. Contoh Hasil *Dimention Reduction*

No	0	1
1	-0,66985	-0,43262
2	-1,39528	-1,57984

Outliers removal merupakan penanganan data *outliers* dengan cara menghapus data tersebut. Langkah awal dimulai dengan melihat persebaran data *outliers* melalui grafik *scatter*. Berikut grafik persebaran data *outliers* dapat dilihat pada Gambar 2.



Gambar 3. Grafik Persebaran Data *Outliers*

Langkah berikutnya melakukan *outliers removal* agar sebaran data menjadi normal menggunakan metric IQR (*interquartile range*). Berdasarkan hasil *dimensionality reduction* pada *dimension* 0 diketahui nilai Q1 (kuartil 1) yaitu -0,872192 dan nilai Q3 (kuartil 3) yaitu -0,773815, sedangan pada *dimension* 1 diketahui nilai Q1 (kuartil 1) yaitu -0,491609 dan nilai Q3 (kuartil 3) yaitu -0,247648. Berikut perhitungan IQR setiap *dimension* dapat dilihat pada Persamaan (15) dan Persamaan (16) sebagai berikut.

$$IQR_0 = Q3 - Q1$$
$$IQR_0 = -0{,}773815 - (-0{,}872192)$$
$$IQR_0 = 0{,}380583 \quad \text{..... (15)}$$
$$IQR_1 = Q3 - Q1$$
$$IQR_1 = -0{,}247648 - (-0{,}491609)$$
$$IQR_1 = 0{,}526167 \quad \text{..... (16)}$$

Berdasarkan nilai IQR tersebut, dapat menentukan persyaratan data *outlers*. Apabila data memiliki kriteria sesuai persyaratan tersebut, maka data tersebut termasuk data *outliers* dan perlu dihilangkan. Berikut persyaratan data *outliers* pada setiap *dimension* sesuai nilai IQR yang telah diketahui.

Syarat *outliers* 1 pada *dimension* 0

$$Data < Q1 - 1{,}5 * IQR_0$$
$$Data < -0{,}872192 - 1{,}5 * 0{,}380583$$
$$Data < -0{,}872192 - 0{,}5708745$$
$$Data < -1{,}4430665$$

Syarat *outliers* 2 pada *dimension* 0

$$Data > Q3 + 1{,}5 * IQR_0$$
$$Data > -0{,}773815 + 1{,}5 * 0{,}380583$$
$$Data > -0{,}773815 + 0{,}5708745$$
$$Data > -0{,}2029405$$

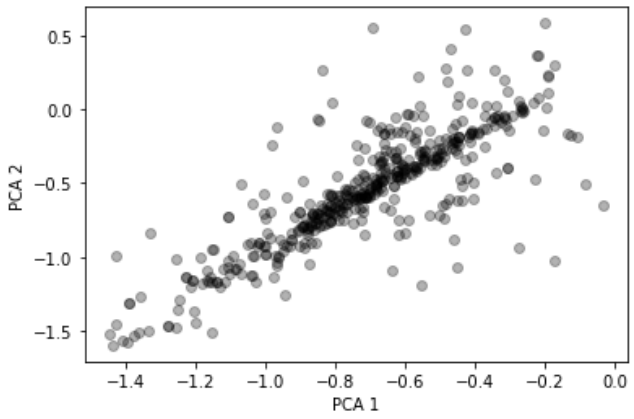
Syarat *outliers* 1 pada *dimension* 1

$$Data < Q1 - 1{,}5 * IQR_1$$
$$Data < -0{,}491609 - 1{,}5 * 0{,}526167$$
$$Data < -0{,}491609 - 0{,}7892505$$
$$Data < -1{,}2808595$$

Syarat *outliers* 2 pada *dimension* 1

$$Data > Q3 + 1{,}5 * IQR_1$$
$$Data > -0{,}247648 + 1{,}5 * 0{,}526167$$
$$Data > -0{,}247648 + 0{,}7892505$$
$$Data > 0{,}5416025$$

Berdasarkan persyaratan nilai *outliers* tersebut, maka syarat nilai *outliers* untuk data pada *dimension* 0 yaitu $-0{,}2029405 < Data < -1{,}4430665$, sedangkan untuk data pada *dimension* 1 yaitu $-0{,}5416025 < Data < -1{,}2808595$. Selanjutnya melihat persebaran data setelah dilakukan *outliers removal*. Berikut grafik persebaran data normal yang dapat dilihat pada Gambar 3.



Gambar 4 Grafik Persebaran Data Normal

Clustering atau pengelompokkan dokumen skripsi dengan algoritma *K-Means Clustering* menggunakan *k* sebagai jumlah *cluster* atau kelompok yang diinginkan. Nilai *k* yang digunakan adalah 2, 3, 4, 5, 6, 7, 8, 9, dan 10. Berikut contoh *centroid* dari *clustering* dengan *k* = 2 dapat dilihat pada Tabel 9 sebagai berikut.

Tabel 9 Contoh *Centroid* dari *Clustering* (*k* = 2)

Centroid	0	1
Cluster 1	-0,76208	-0,61024
Cluster 2	-0,37958	-0,13354

Kemudian menghitung jarak setiap centroid dengan setiap dokumen. Perhitungan jarak menggunakan algoritma Cosine Similarity Berikut contoh perhitungan jarak Cosine Similarity dapat dilihat pada persamaan (17) sebagai berikut.

$$cosSim(d_j, q) = \frac{\sum_{i=1}^{i}(W_{ij} \cdot W_{iq})}{\sqrt{\sum_{i=1}^{t} W_{ij^2} \cdot \sum_{i=1}^{t} W_{iq^2}}}.$$

$$cosSim(d_j, q) = \frac{((-0{,}66985)*(-0{,}76208))+((-0{,}43262)*(-0{,}61024))}{\sqrt{((-0{,}66985)^2+(-0{,}43262)^2)*((-0{,}76208)^2+(-0{,}61024)^2)}}.$$

$$cosSim(d_j, q) = \frac{(0{,}51047)+(0{,}26400)}{\sqrt{((0{,}44869)+(0{,}18716))*((0{,}58076)+(0{,}37239))}}.$$

$$cosSim(d_j, q) = \frac{0{,}77447}{\sqrt{0{,}63585*0{,}95315}}.$$

$$cosSim(d_j, q) = \frac{0{,}77447}{\sqrt{0{,}6060}}.$$

$$cosSim(d_j, q) = \frac{0{,}77447}{0{,}77846}.$$

$$cosSim(d_j, q) = 0{,}9948 \quad \text{……………………………………… (17)}$$

Maka didapatkan jarak setiap *centroid* dengan setiap dokumen. Berdasarkan jarak tersebut, memilih jarak dengan nilai terbesar antara *centroid* 1 dan *centroid* 2 pada setiap dokumen. Berikut contoh hasil perhitungan jarak pada *clustering* (*k* = 2) dapat dilihat pada Tabel 10 sebagai berikut.

Tabel 10. Contoh Hasil Perhitungan Jarak *Clustering* (*k* = 2)

Iterasi 1		
Dokumen	Cluster 1	Cluster 2
Doc 1	0,99482804	0,97247428
Doc 2	0,98521775	0,87319483
Doc 3	0,89508287	0,99216543
Doc 4	0,96502551	0,99743314
Doc 5	0,97185817	0,99509171

Berikut contoh hasil *clustering* (*k* = 2) dapat dilihat pada Tabel 11 sebagai berikut.

Tabel 11. Contoh Hasil Clustering (k = 2)

Iterasi 1			
Cluster	Dokumen		
Cluster 1	Doc 1	Doc 2	
Cluster 2	Doc 3	Doc 4	Doc 5

Pengujian dilakukan pada setiap nilai k menggunakan algoritma silhouette coefficient untuk menentukan nilai k terbaik dalam mengelompokkan dokumen skripsi. Proses clustering menggunakan metode K-Means Clustering dan algoritma *silhouette* coefficient. Berikut tabel pengujian *silhouette* coefficient dapat dilihat pada Tabel 12.

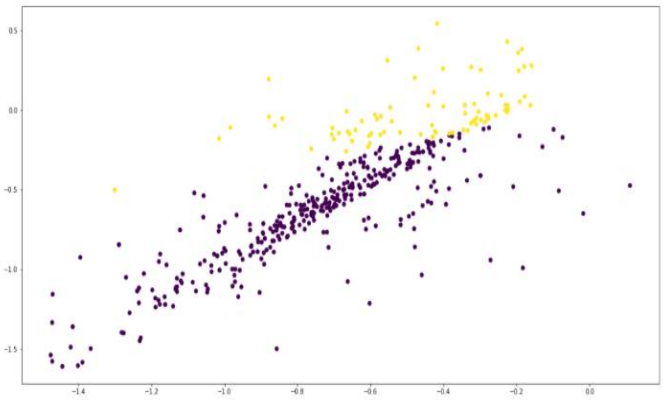
Tabel 12 Tabel Pengujian *Silhouette Coefficient*

Nilai *k*	*Silhouette Coefficient*	Rata-Rata	Nilai *Structure*
k = 2	0.81857570726257 0.7975547372053398 0.7975547372053398 0.81857570726257 0.7975547372053398	0,8059	*Strong Structure*
k = 3	0.7417767705410228 0.7417767705410228 0.7417767705410228 0.7417767705410228 0.7490075721349593	0,74154	*Strong Structure*
k = 4	0.6288436364946974 0.7588345358573692 0.7161725257697091 0.7588345358573692 0.5618699674130047	0,68486	*Medium Structure*
k = 5	0.6855081020380019 0.6807781929701706 0.6477082242527575 0.6855081020380019 0.6477082242527575	0,66942	*Medium Structure*

Nilai *k*	*Silhouette Coefficient*	Rata-Rata	Nilai *Structure*
k = 6	0.6703775561524259 0.6703775561524259 0.6671763544474465 0.668968128740469 0.6703775561524259	0,66938	*Medium Structure*
k = 7	0.6409041967592662 0.6409041967592662 0.6538689785575246 0.6409041967592662 0.639432912153537	0,64318	*Medium Structure*
k = 8	0.6452074454474165 0.6576988866830493 0.648827828214657 0.5903244823012801 0.648827828214657	0,63814	*Medium Structure*
k = 9	0.6851206444127916 0.6530670233198297 0.6524652567855804 0.6540997715695296 0.6464460923655951	0,65822	*Medium Structure*
k = 10	0.6836270567058575 0.6454320223005103 0.6412193218770409 0.6538753871420315 0.6454320223005103	0,65388	*Medium Structure*

Berdasarkan Tabel 12, dapat dilihat bahwa nilai *k* terbaik ketika *k* bernilai 2. Menurut Kaufman dan Rousseeuw, nilai struktur dari *k* bernilai 2 menghasilkan *strong structure* atau struktur yang kuat, maka *term* dari setiap dokumen dapat sepenuhnya menjadi ciri dari dokumen tersebut. Lalu ketika nilai *k* bernilai lebih besar dari 2, maka dokumen yang seharusnya berada satu *cluster* akan terpecah ke cluster lain. Pengelompokkan terbaik terdapat pada nilai *silhouette coefficient* tertinggi yaitu 0.80 dengan nilai *k* = 2. Berikut grafik persebaran *clustering* dapat dilihat pada Gambar 4.

Topic detection berguna untuk menentukan topik setiap *cluster* untuk dijadikan sebagai label dataset. Algoritma LDA digunakan untuk menghitung probabilitas kemunculan setiap kata dan diurutkan berdasarkan nilai probabilitas. Berikut hasil topic detection pada setiap *cluster*



Gambar 5. Grafik Persebaran *Clustering*

a. *Topic detection cluster* 0

```
Topik Cluster 0
[(0, '0.008*"nilai" + 0.007*"jember" + 0.007*"satu" + 0.007*"xd" + 0.006*"salah" + 0.006*"proses" + 0.006*"perusahaan" + 0.006
*"aplikasi" + 0.005*"memiliki" + 0.005*"pengguna" + 0.005*"menjadi" + 0.005*"melakukan" + 0.005*"berdasarkan" + 0.004*"yangxd"
+ 0.004*"lebih" + 0.004*"sesuai" + 0.004*"kabupaten" + 0.004*"jumlah" + 0.004*"keputusan" + 0.004*"kriteria" + 0.004*"sebuah" +
0.004*"perhitungan" + 0.004*"tahap" + 0.004*"produk" + 0.004*"menentukan" + 0.004*"kebutuhan" + 0.004*"suatu" + 0.003*"akan" +
0.003*"masyarakat" + 0.003*"pengembangan"')]
```

Gambar 6 *Topic Detection Cluster* 0

b. *Topic detection cluster* 1

```
Topik Cluster 1
[(0, '0.014*"jember" + 0.014*"data" + 0.011*"nilai" + 0.009*"xd" + 0.009*"produk" + 0.008*"proses" + 0.007*"aplikasi" + 0.007
*"menjadi" + 0.007*"memiliki" + 0.007*"penjualan" + 0.006*"melakukan" + 0.006*"yangxd" + 0.006*"sebuah" + 0.006*"layanan" + 0.0
06*"kriteria" + 0.006*"pengguna" + 0.006*"universitas" + 0.006*"berdasarkan" + 0.006*"danxd" + 0.005*"tembakau" + 0.005*"perusa
haan" + 0.005*"ini" + 0.005*"berbasis" + 0.005*"kabupaten" + 0.005*"suatu" + 0.005*"sesuai" + 0.005*"model" + 0.005*"lebih" +
0.005*"tingkat" + 0.005*"penyakit"')]
```

Gambar 7. *Topic Detection Cluster* 1

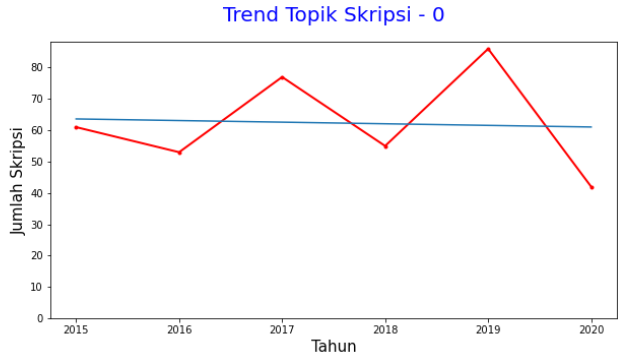
Berdasarkan probabilitas kata yang muncul dari hasil *topic detection cluster* 0 pada Gambar 5. Maka topik *cluster* 0 yaitu Pengembangan Sistem Pendukung Keputusan.

Berdasarkan probabilitas kata yang muncul dari hasil *topic detection cluster* 1 pada Gambar 6. Maka topik *cluster* 1 yaitu Pengukuran Kinerja Layanan TI dan Peramalan.

Visualisasi data dilakukan untuk melihat sebaran data pada setiap topik skripsi. Algoritma *trend linear* digunakan untuk mengetahui *trend* yang terjadi. Berikut hasil visualisasi data *trend* topik skripsi.

a. Visualisasi data topik Pengembangan Sistem Pendukung Keputusan

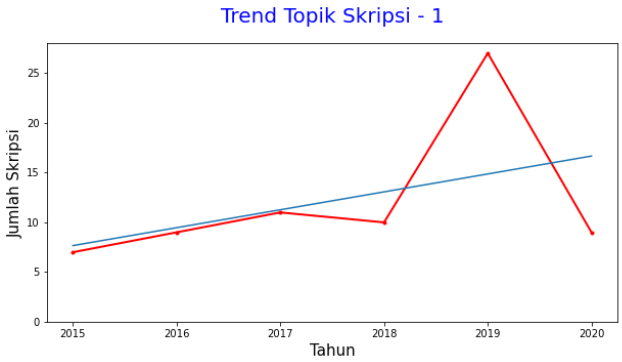
Jumlah dokumen skripsi semakin meningkat setiap tahunnya, namun terjadi penurunan drastis pada tahun terakhir. Total dokumen skripsi dengan topik Pengukuran Kinerja TI sebanyak 374 dokumen dengan rincian tahun 2015 berjumlah 61, tahun 2016 berjumlah 53, tahun 2017 berjumlah 77, tahun 2018 berjumlah 55, tahun 2019 berjumlah 86, dan tahun 2020 berjumlah 42. Berikut grafik *trend* topik Pengembangan Sistem Pendukung Keputusan dapat dilihat pada Gambar 7.



Gambar 8. Grafik *Trend* Topik Pengembangan Sistem Pendukung Keputusan

b. Visualisasi data topik Pengukuran Kinerja TI dan Peramalan

Jumlah dokumen skripsi tidak beraturan setiap tahunnya. Total dokumen skripsi dengan topik Pengembangan Sistem Informasi sebanyak 73 dokumen dengan rincian, tahun 2015 berjumlah 7, tahun 2016 berjumlah 9, tahun 2017 berjumlah 11, tahun 2018 berjumlah 10, tahun 2019 berjumlah 27, dan tahun 2020 berjumlah 9. Berikut grafik *trend* topik Pengukuran Kinerja TI dan Peramalan dapat dilihat pada Gambar 8.



Gambar 9. Grafik *Trend* Topik Pengukuran Kinerja TI dan Peramalan

Analisis data diperlukan untuk menjabarkan hasil dari visualisasi data. Hasil analisis data dapat digunakan dalam menentukan kebijakan pihak instansi agar lebih mengoptimalkan penelitian skripsi di masa depan. Berdasarkan hasil visualisasi data setiap topik skripsi, maka didapatkan hasil analisis sebagai berikut.

a. Topik Pengembangan Sistem Pendukung Keputusan

Topik Pengembangan Sistem Pendukung Keputusan pada hasil visualisasi data menunjukkan bahwa terjadi *trend* turun yang artinya semakin berkurangnya skripsi dengan topik tersebut. Hal ini cukup baik karena topik ini memiliki jumlah yang sangat banyak, namun penurunannya terjadi tidak signifikan sehingga perlu untuk lebih mengurangi topik ini agar penurunannya signifikan.

b. Topik Pengukuran Kinerja TI dan Peramalan

Topik Pengukuran Kinerja TI dan Peramalan pada hasil visualisasi data menunjukkan bahwa terjadi *trend* naik yang artinya semakin meningkatnya skripsi dengan topik tersebut. Hal ini cukup baik karena topik ini memiliki jumlah yang cukup sedikit.

Berdasarkan hasil analisis data pada setiap *cluster*, menunjukkan bahwa topik-topik pada penelitian skripsi yang telah dilakukan mahasiswa terdiri dari Pengembangan Sistem Penunjang Keputusan, Pengukuran Kinerja TI, dan Peramalan. Maka dapat disimpulkan topik Pengembangan Sistem Penunjang Keputusan perlu dikurangi sehingga topik-topik lain dapat muncul seperti topik *Artificial Intelligence*, *E-Business*, Sistem *Monitoring*, *Graph*, Keamanan, Jaringan, *Requirement Analysis*, *Software Testing*, *Digital Forensik*, dan lain-lain.

KESIMPULAN

Dari hasil pengelompokkan dapat disimpulkan bahwa *K-Means Clustering* dengan *Cosine Similarity* cukup baik untuk pengelompokkan dokumen skripsi. Dokumen skripsi dikelompokkan dengan mengambil bagian abstrak sebagai intisari dokumen skripsi. Namun dokumen skripsi melewati beberapa tahapan terlebih dahulu, seperti *preprocessing*, *term weighting*, *dimensionality reduction*, dan *outliers removal*. Tahapan *dimensionality reduction* menggunakan *Principal Component Analysis* cukup penting karena *K-Means Clustering* sangat sensitif terhadap data berdimensi besar sehingga perlu mengurangi jumlah dimensi data tanpa menghilangkan variasi data.

Pengujian menggunakan *Silhouette Coefficient* dengan nilai *k* yang bervariasi yaitu 2, 3, 4, 5, 6, 7, 8, 9, dan 10. Berdasarkan hasil pengujian didapatkan nilai terbaik pada $k = 2$ dengan nilai *Silhouette* 0,80. Berdasarkan hasil pengujian dapat disimpulkan nilai *k* yang semakin besar akan berpengaruh buruk terhadap *cluster* yang terbentuk.

Clustering menghasilkan 2 *cluster* yang artinya terdapat 2 topik dari seluruh dokumen skripsi yang dapat dilakukan *topic detection* menggunakan *Latent Dirichlet Allocation* untuk mendapatkan topik dari setiap *cluster* yang dapat digunakan sebagai topik skripsi. Berdasarkan hasil visualisasi data dapat disimpulkan bahwa *Least Square* dapat membuat garis *trend* dengan baik sehingga dapat mengetahui *trend* yang terjadi pada setiap *cluster*. Dari hasil analisis data seperti yang dijelaskan pada bagian hasil dan pembahasan, menunjukkan bahwa topik Pengembangan Sistem Pendukung Keputusan terjadi *trend* turun, dan topik Pengukuran Kinerja TI dan Peramalan terjadi *trend* naik. Maka dapat disimpulkan topik Pengembangan Sistem Penunjang Keputusan perlu dikurangi sehingga topik-topik lain dapat muncul.

Berdasarkan hasil analisis keseluruhan penelitian terdapat banyak penulisan *typo* (ejaan yang salah) pada dokumen skripsi khususnya abstrak sehingga menyebabkan proses pembobotan menjadi kurang optimal dalam menghitung bobot setiap kata. Maka disarankan melakukan *spell checker* untuk mendeteksi kata *typo* tersebut. Secara umum *spell checker* dapat dilakukan pada tahap *preprocessing*.

DAFTAR PUSTAKA

[1] N. Gupta, Text Mining for Information Retrieval. India: Jaypee Institute of Information Technology University, 2011.

[2] M. K. Nasution, Abstrak - Suatu Karya Ilmiah. 13-16, 2017.

[3] N. A. Syaripul, and A. M. Bachtiar, "Visualisasi data interaktif data terbuka pemerintah Provinsi DKI Jakarta: Topik ekonomi dan keuangan daerah," *Jurnal Sistem Informasi (Journal of Information Systems),* vol. 12, pp. 82-89, 2016.

[4] V. Gupta, and G. S. Lehal, "A survey of text mining techniques and application," *Journal of Emerging Technologies in Web Intelligence,* vol. 1, pp. 60-75, 2019.

[5] B. Simamora, Analisis Multivariat Pemasaran, Jakarta: PT. Gramedia Pustaka Umum, 2005.

[6] R. W. Astuti, Badieah, and B. Satrio, "Rancang bangun sistem klasterisasi dokumen menggunakan metode k-means untuk identifikasi topik dokumen tugas akhir Program Studi Teknik Informatika Universitas Islam Sultan Agung," *Konferensi Ilmiah Mahasiswa UNISSULA (KIMU 2)*, pp. 197-205, 2019

[7] M. S. Hudin, M. A. Fauzi, and S. Adinugroho, "Implementasi metode text mining dan k-means clustering untuk pengelompokkan dokumen skripsi (Studi Kasus: Universitas Brawijaya)," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer,* vol. 2, no. 11, pp. 5518-5524, 2018.

[8] A. I. Alfanzar, Khalid, and I. S. Rozas, "Topic modelling skripsi menggunakan metode latent diriclet allocation," *Jurnal Sistem Informasi,* vol. 7, no. 1, pp. 7-13, 2020.

[9] J. Asian, Effective Techniques for Indonesian Text Retrieval. Royal Melbourne Institute of Technology University*,* 2017.
[10] R. Handoyo, R. M. Rumani, and S. N. Michrandi, "Perbandingan clustering menggunakan metode single linkage dan k-means pada pengelompokkan dokumen," *JSM STMIIK Mikroskil,* vol. 15, no. 2, 2015.
[11] L. Kaufman, and P. J. Rousseuw, Finding Groups in Data. New York: John Wiley & Sons, 1990.
[12] I. M. Putra, and R. P. Kusumawardani, "Analisis topik informasi publik media sosial di surabaya menggunakan pemodelan Latent Dirichlet Allocation (LDA). *Jurnal Teknik ITS,* vol. 6, no. 2, pp. A312, 2017.
[13] A. Syaifuddin, R. A. Harianto, and J. Santoso, "Analisis trending topik untuk percapakan media sosial dengan menggunakan topic modelling berbasis algoritme LDA," *Journal of Intelligent Systems and Computation*, pp. 12-19, 2019.