Journal of Applied Informatics and Computing (JAIC)
Vol.7, No.2, December 2023, pp. 258~262
e-ISSN: 2548-6861

Clustering Balinese Language Documents using the Balinese Stemmer Method and Mini Batch K-Means with K-Means++

Made Agus Putra Subali [1]*, I Gusti Rai Agung Sugiartha [2]*, Komang Budiarta [3]*, I Made Budi Adnyana [4]*
* Sistem Informasi, Institut Teknologi dan Bisnis STIKOM Bali
madeagusputrasubali@gmail.com[1], sugiarta@stikom-bali.ac.id[2], komang_budiarta@stikom-bali.ac.id[3], budi.adnyana@stikom-bali.ac.id[4]

Article Info

Article history:

Received 2023-09-03
Revised 2023-09-18
Accepted 2023-09-26

Keyword:

Clustering,
Balinese Language Documents,
Balinese Stemmer,
Mini Batch k-Means,
k-Means++.

ABSTRACT

Clustering aims to categorize data into n groups, where data within each group exhibits maximum similarity, while the similarity between groups is minimized. Among various clustering methods, k-means is widely employed due to its simplicity and ability to yield optimal clustering results. However, the k-means method is susceptible to slow processing in high-dimensional datasets and the clustering outcomes are sensitive to the initial selection of cluster center values. In addressing these limitations, this study employs the k-means mini-batch method to enhance processing speed for high-dimensional data and utilizes the k-means++ method to optimize the selection of initial cluster center values. The dataset for this research comprises 300 news articles in Balinese sourced from the https://balitv.tv/ website. Prior to the clustering process, a stemming procedure is applied using the Balinese stemmer method to enhance recall. The obtained results reveal that a majority of the 300 data instances exhibit a high degree of similarity, as indicated by the clustering results. If the number of clusters (n) exceeds two, the data fails to be distinctly separated due to the high structural similarity among the data instances. This can be attributed to the relatively small number of words or attributes produced. In future research, feature reduction will be implemented, and a clustering method capable of addressing data overlap will be explored.

I. PENDAHULUAN

Meningkatnya volume dokumen teks berbahasa Bali merupakan upaya digital untuk mendukung pelestarian bahasa Bali sebagai sumber referensi yang dapat diakses secara fleksibel [1]. Mengingat jumlah dokumen yang banyak akan menimbulkan kesulitan jika penentuan kelompok setiap dokumen dilakukan secara manual [2]. Metode *clustering* adalah analisis statistik yang bertujuan untuk mengelompokkan data menjadi beberapa kelompok agar data dalam satu kelompok memiliki kemiripan yang maksimal dan data antar kelompok memiliki kemiripan yang minimal [3]. Data yang digunakan dalam penelitian ini adalah 300 artikel berita berbahasa Bali berbentuk deskripsi singkat dari artikel video yang diperoleh dari situs https://balitv.tv/. Apabila diperhatikan pada data tersebut belum memiliki *label* berita sehingga metode *clustering* akan digunakan untuk mengelompokkan seluruh artikel tersebut.

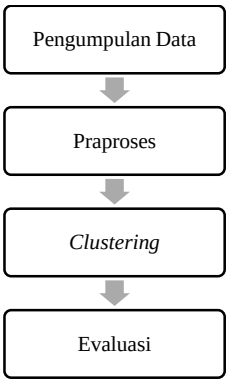
K-means adalah salah satu metode *clustering* yang paling banyak digunakan [4]. *k*-means memiliki kelebihan yaitu mudah diimplementasikan, diadaptasi, umum digunakan, dan memberikan hasil yang optimal pada data dua dimensi [5], namun *k*-means memiliki kelemahan pada lambatnya proses pengelompokan data untuk dataset yang besar [6]. Metode *mini batch k*-means mampu mengatasi kekurangan *k*-means dengan menggunakan *mini batch* untuk mengurangi waktu komputasi pada dataset yang besar [7], berdasarkan hal tersebut, metode *mini batch k*-means sangat ideal digunakan untuk karakteristik data dalam penelitian ini.

Metode *k*-means dan *mini batch k*-means memiliki kekurangan yang sama yaitu sensitif terhadap pusat *cluster* awal yang dipilih [8]. Metode *k*-means++ digunakan untuk mengatasi kekurangan pada *k*-means dan *mini batch k*-means dengan cara memilih pusat klaster pertama secara acak dan kemudian dipilih pusat klaster berdasarkan perhitungan jarak terdekat antara titik data dan pusat *cluster* yang dipilih [9].

Karakteristik setiap artikel berita yang digunakan merupakan deskripsi singkat dari berita berbentuk video, hal ini merupakan kendala dalam proses *text mining* karena minimnya dataset yang dimiliki setiap artikel. Untuk menanggulangi hal ini, peneliti menggunakan *stemmer* bahasa Bali pada tahap pra-proses sebelum *clustering* dilakukan. Hal ini bertujuan untuk meningkatkan *recall*, dengan mengubah setiap kata dalam bentuk kata dasarnya [10]. Pada penelitian ini *k*-means++ akan diterapkan untuk pemilihan awal *cluster center* pada metode *mini batch k*-means, sehingga hasil *clustering* data lebih optimal.

II. METODE PENELITIAN

Tahapan penelitian yang dilakukan, dimulai dari pengumpulan data, praproses, pengelompokkan data, dan mengevaluasi hasil.



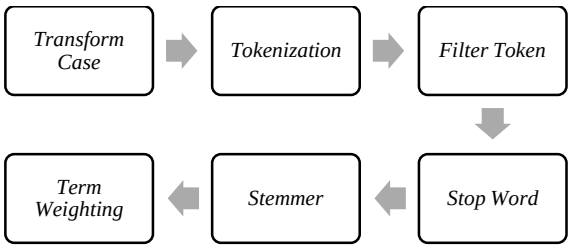
Gambar 1. Tahapan Penelitian

A. Pengumpulan Data

Data yang digunakan berjumlah 300 artikel teks berita berbahasa Bali, yang diperoleh dari situs https://balitv.tv/ dari tahun 2018-2022. Pada Gambar 3 merupakan detail alur proses pengumpulan data.

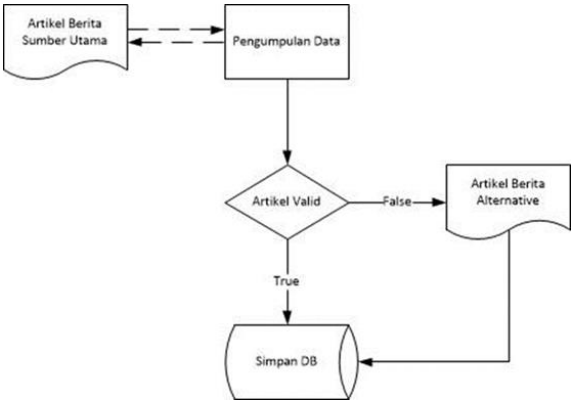
B. Praproses

Setelah data diperoleh, tahap selanjutnya adalah melakukan praproses dengan melakukan beberapa tahapan, seperti *transform case*, *tokenization*, *filter token*, *stop word removal*, *stemmer*, dan *term weighting*. Pada Gambar 2 merupakan alur proses tahapan praproses yang dilakukan.



Gambar 2. Alur Proses Tahapan Praproses

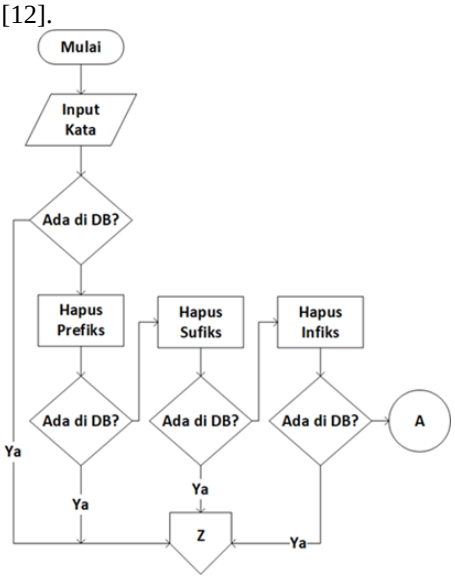
Pada tahap pertama dilakukan *transform case* dimana pada tahap ini semua huruf diubah ke bentuk *lowercase*. Tahap selanjutnya dilakukan proses *tokenization* dengan tujuan untuk memisahkan setiap kata, pada umumnya setiap kata dipisah dengan tanda spasi. Setelah setiap kata terpisah kemudian dilakukan proses *filter token* merupakan tahap untuk mengeliminasi kata yang memiliki jumlah karakter yang kurang dari tiga karakter atau melebihi dua puluh karakter, hal ini dilakukan untuk memastikan setiap token yang digunakan memiliki makna. Proses selanjutnya dilakukan *stop word* dimana pada tahap ini dilakukan penghilangan kata sambung dalam bahasa Bali yang tidak begitu penting dalam pemrosesan *clustering*, kata-kata tersebut meliputi: *anggen*, *sane*, *ring*, *miwah*, *puniki* dan *olih*. Setelah jumlah data direduksi, selanjutnya dilakukan proses *stemming* untuk meluluhkan kata berafiks ke bentuk dasarnya, sebagai contoh: kata *ngajeng* diubah ke bentuk *ajeng*. Tujuan dari dilakukannya proses *stemming* adalah untuk meningkatkan nilai *recall* [10]. Untuk *stemmer* yang digunakan adalah *Balinese Stemmer*, dimana dalam metode *stemmer* tersebut dibangun menggunakan metode *rule-based* dan *n*-gram *string similarity matching* dalam proses meluluhkan kata berafiks.



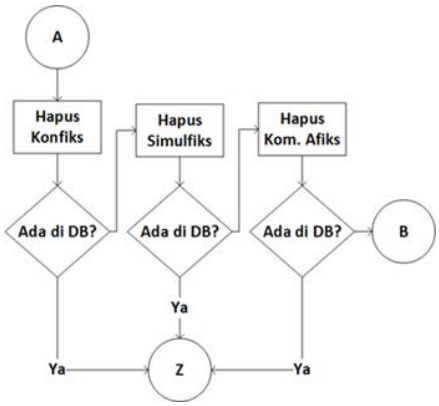
Gambar 3. Alur Proses Pengumpulan Data

Pada gambar 4-6 merupakan alur tahapan proses pada metode *Balinese Stemmer* menggunakan kombinasi metode *rule-based* dan *n*-gram *stemming*, dimana pada tahap awal dilakukan perbandingan apakah kata input merupakan kata dasar atau bukan dengan cara membandingkannya pada kamus kata dasar yang tersimpan pada *database*, apabila kata input merupakan kata dasar, maka kata dasar ditampilkan, apabila kata input bukan merupakan kata dasar, maka dilakukan proses meluluhkan afiks, dimulai dari meluluhkan prefiks, sufiks, infiks, konfiks, simulfiks, dan kombinasi afiks apabila *rules* yang tersedia tidak dapat mengenali kata input, maka untuk memperoleh kata dasar dilakukan proses *string similarity* menggunakan metode *n*-gram *stemming*, apabila tingkat kemiripannya memenuhi nilai ambang batas maka kata dasar ditampilkan. Adapun alur proses perancangan sistem dilakukan dalam [11]. Tahap terakhir pada praproses dilakukan *term weighting* merupakan tahap untuk mengetahui

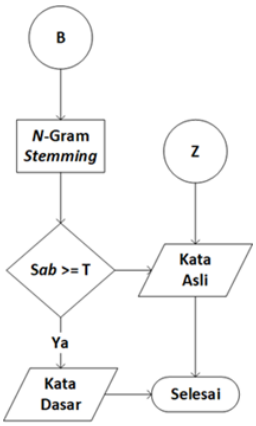
seberapa sering suatu kata atau *term* muncul dalam setiap dokumen. Pada proses *term weighting* menghasilkan *term* matriks yang nantinya digunakan untuk pemrosesan *clustering*. Metode *term weighting* yang digunakan adalah $TF \cdot IDF$ [12].



Gambar 4. *Flowchart Balinese Stemmer*



Gambar 5. *Flowchart On Page Connector A Balinese Stemmer*



Gambar 6. *Flowchart On Page Connector B Balinese Stemmer*

C. Clustering

Setelah tahapan praproses dilakukan selanjutnya dilakukan proses *clustering* menggunakan metode *mini batch k*-means dan *k*-means++. Penggunaan metode *k*-means++ digunakan pada saat inisialisasi awal nilai pusat *cluster* pada metode *mini batch k*-means, dimana pada metode *mini batch k*-means, nilai awal pusat *cluster*, sebelumnya yang dipilih secara acak atau *random*, hal ini mengakibatkan hasil *clustering* sangat tergantung pada pemilihan nilai awal pusat *cluster*, ketika nilai setiap *k cluster* ditentukan dengan baik maka hasil *cluster* akan optimal, sebaliknya apabila penentuan nilai awal tidak tepat maka hasil *cluster* tidak optimal.

Metode *mini batch k*-means adalah penggunaan kumpulan data acak kecil sehingga dapat disimpan dalam memori. Dalam setiap iterasi, sejumlah kecil kumpulan data acak baru diperoleh dan digunakan untuk pemutakhiran pusat *cluster* dan terus berulang hingga konvergensi. Setiap *mini batch* memperbarui pusat *cluster* menggunakan metode *gradient descend*. Misalkan dataset dan pusat *cluster* sama ambil b data acak sebanyak b dari dataset X, yang disimbolkan M, dimana variabel b menggambarkan ukuran *mini batch* [2]. Dengan demikian, fungsi tujuan dari metode ini adalah:

$$\min_{\mu k} f(\mu, x) = \sum_{x_n \in M} \sum_K \|\mu_x - k_n\|^2 \qquad (1)$$

Sedangkan fungsi yang digunakan untuk memperbarui pusat *cluster* adalah:

$$\mu_{k_{i+1}} = (1 - \eta_k)\mu_{k_i} + \eta_k x_n \qquad (2)$$

Tahapan metode *mini batch k*-means dapat dilihat pada Gambar 7:

Input:
k: jumlah *cluster*
b: ukuran *mini batch*
t: batas pengulangan
X: $\{x_1, x_2, x_3, \ldots, x_n\}$ *dataset*
Output:
C: $\{c_1, c_2, c_3, \ldots, c_n\}$
1 Pilih pusat *cluster* x dari *dataset* X secara *random*
2 $v \leftarrow 0$
3 *for* $i = 0$ *to* t *do*
  $M \leftarrow b$
  for $x_n \in M$ *do*
    $d[x_n] \leftarrow \min_{\mu k} f(\mu, x)$
  end for

  for $x_n \in M$ *do*
    $\mu_k \leftarrow d[x_n]$
    $v[\mu_k] \leftarrow v[\mu_k] + 1$
    $\eta_k \leftarrow \frac{1}{v[\mu_k]}$
    $\mu_{k_{i+1}} = (1 - \eta_k)\mu_{k_i} + \eta_k x_n$
  end for
end for

Gambar 7. Tahapan Metode *Mini Batch k*-Means

Hasil *clustering* yang baik mampu mengelompokkan data menjadi beberapa *cluster* sehingga data dalam satu *cluster*

memiliki kemiripan yang maksimal dan sebaliknya data antar *cluster* memiliki kemiripan yang minimal. Secara intuitif, merupakan pilihan bijak untuk memilih pusat klaster yang saling berjauhan. Algoritma *k*-means++ mengikuti ide ini, tetapi titik terjauh tidak selalu dipilih untuk menjadi *centered cluster*, *k*-means++ sangat sederhana dan mudah diimplementasikan. Pada dasarnya, kecuali pusat klaster awal yang dipilih secara acak dari titik data, setiap pusat klaster berikutnya dipilih berdasarkan perhitungan jarak terdekat antara titik data dengan pusat klaster yang telah dipilih sebelumnya. Beri $D(x)$ jarak *euclidean* antara x dan pusat klaster yang telah dipilih [3], [8], [9]. Detail metode *k*-means++ digambarkan pada Gambar 8.

Input:
k: jumlah cluster
X: $\{x_1, x_2, x_3, \ldots, x_n\}$ dataset
Output:
C: $\{c_1, c_2, c_3, \ldots, c_k\}$
1 $C \leftarrow \emptyset$
2 Pilih pusat cluster x dari dataset X secara random, $C = C \cup \{x\}$
3 Ulangi
4 Pilih $x \in X$ dengan probabilitas $D(x)^2 / \sum_{x \in X} D(x)^2$
5 $C = C \cup \{x\}$
6 Hingga cluster pusat k terisi
7 Proses clustering dengan Mini Batch K-Means

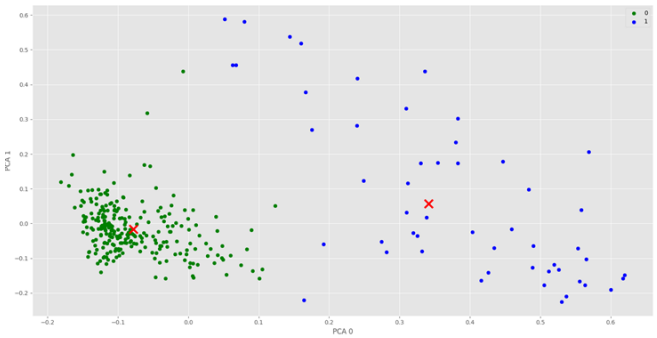
Gambar 8. Tahapan Metode *k*-Means++

D. Evaluasi

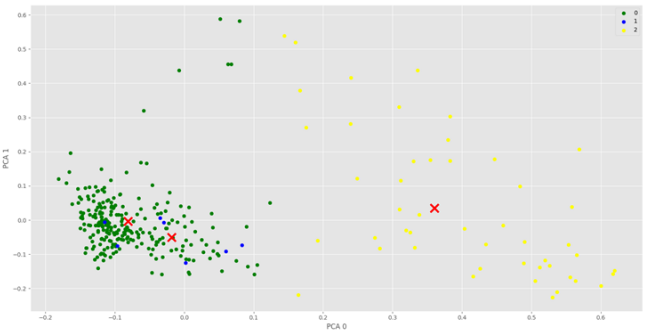
Proses evaluasi dilakukan dengam membandingkan hasil yang dihasilkan oleh metode usulan denngan metode *k*-means dan *k*-means++ dengan menggunakan beberapa *n cluster*.

III. HASIL DAN PEMBAHASAN

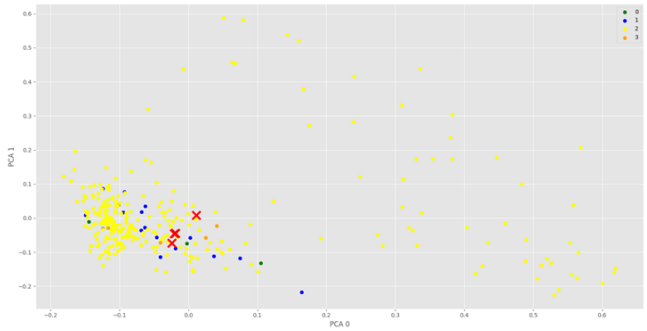
Berdasarkan hasil *clustering* pada 300 artikel menggunakan beberapa variasi *n cluster*, yaitu *n* =2, *n*=3, *n*=4 menggunakan metode *mini batch k*-means dan pemilihan pusat *cluster* menggunakan metode *k*-means++ terlihat pada Gambar 9-10 hasil *clustering* dari kombinasi metode *mini batch k*-means dan *k*-means++ menggunakan jumlah *cluster n* 2, 3, dan 4.



Gambar 9. *Scatter Plot Mini Batch k-Means* dan *k-Means++* 2 *Clusters*



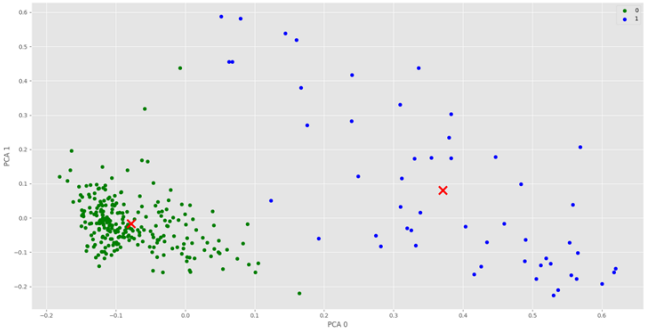
Gambar 10. *Scatter Plot Mini Batch k-Means* dan *k-Means++* 3 *Clusters*



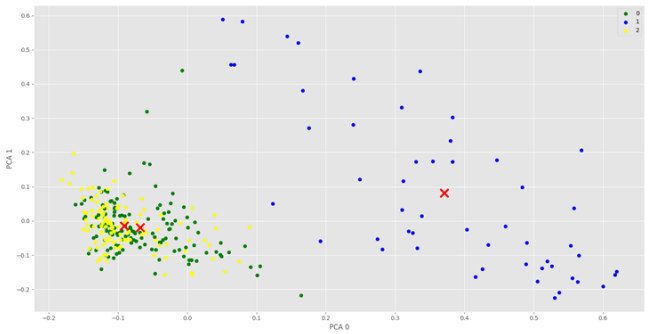
Gambar 11. *Scatter Plot Mini Batch k-Means* dan *k-Means++* 4 *Clusters*

Berdasarkan hasil yang diperoleh pada beberapa variasi *n cluster*, terlihat data pada *n*=2 dapat terpisah dengan baik, sedangkan untuk *n*=3 dan *n*=4 data tidak terpisah dengan baik. Pada Gambar 12 hingga Gambar 14 merupakan hasil *clustering* menggunakan metode *k*-means dan *k*-means++.

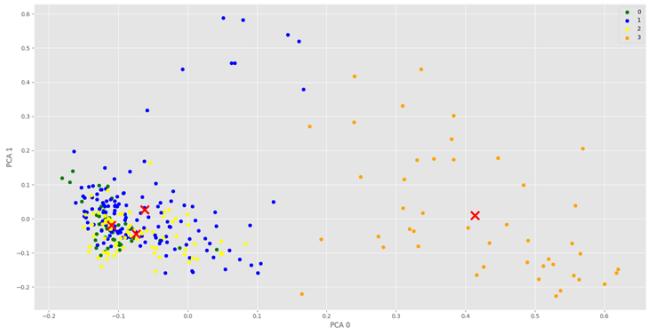
Sejalan dengan hasil dari metode sebelumnya pada penggunaan metode *k*-means dan *k*-means++ hasil terbaik ditunjukan oleh jumlah *n cluster n*=2, sedangkan *n*=3 dan *n*=4 menunjukkan hasil yang tidak begitu baik.



Gambar 12. *Scatter Plot k-Means* dan *k-Means++* 2 *Clusters*



Gambar 13. *Scatter Plot k-Means* dan *k-Means++* 3 *Clusters*



Gambar 14. *Scatter Plot k-Means* dan *k-Means++* 4 *Clusters*

Berdasarkan hasil *clustering* yang telah diperoleh, menunjukan hasil yang kurang begitu baik. Apabila diperhatikan pada Gambar 10, 11 dan Gambar 13, 14 yang menggunakan jumlah *n cluster* tiga dan empat memberikan hasil *cluster* yang kurang optimal, hal ini bisa dilihat pada tiga titik pusat *cluster* (disimbolkan *x* berwarna *merah*) saling berdekatan satu sama lain. Salah satu kendala atau penyebab burukmya hasil *clustering* yang dihasilkan adalah data yang digunakan tidak terdistribusi dengan baik. Distribusi data yang tidak merata atau *overlap* pada setiap *cluster* dapat menyebabkan metode usulan kesulitan dalam membedakan setiap *cluster* tersebut [13]. Penyebab terjadinya *overlap* dikarenakan tingkat kemiripan antar data yang digunakan sangat tinggi, mengingat mayoritas data berupa artikel pendek berbahasa Bali yang hanya terdiri dari beberapa kata, sebagai contoh berikut adalah salah satu artikel berita yang digunakan: "*kabupaten buleleng sampun sayaga ngmargiang vaksinasi covid-19 tahap kapertama vaksinasi puniki karencanayang kakawitin ring tanggal pitulikur rahinane mangkin*". Hal ini diperkuat dari ekstraksi fitur yang dilakukan pada tahap pembobotan kata menggunakan metode $TF \cdot IDF$. Mayoritas fitur yang dihasilkan dari setiap data atau artikel memiliki jumlah fitur yang sama.

IV. KESIMPULAN

Berdasarkan hasil dari proses *clustering* menggunakan kombinasi metode *mini batch k*-means dengan pemilihan *cluster* pusat awal menggunakan metode *k*-means++, hasil yang diberikan menunjukkan ketiga ratus data yang digunakan memiliki tingkat kesamaan yang tinggi, terlihat dari pemberian *n cluster* apabila *n* lebih besar dari dua maka data tidak dapat terpisah dengan baik. Hasill *clustering* yang tidak begitu baik pada *n cluster* lebih dari dua, dikarenakan adanya *overlap* atau distribusi yang tidak merata pada data yang digunakan, mengingat ketiga ratus artikel bahasa Bali yang diperoleh memiliki karakteristik yang sama, dimana setiap artikel terdiri dari ± 30-100 kata, hal ini membuat fitur yang dihasilkan memiliki tingkat kemiripan yang tinggi sehingga menyulitkan algoritma yang digunakan untuk melakukan proses *clustering*.

UCAPAN TERIMA KASIH

Peneliti mengucapkan terima kasih kepada Institut Teknologi dan Bisnis STIKOM Bali yang telah memberi dukungan finansial terhadap kelancaran penelitian ini.

DAFTAR PUSTAKA

[1] I. B. G. W. Putra, M. Sudarma, and I. N. S. Kumara, "Klasifikasi Teks Bahasa Bali dengan Metode Supervised Learning Naive Bayes Classifier," *Teknologi Elektro*, vol. 15, no. 2, pp. 81–86, 2016.

[2] S. R. Fitriyani and H. Murfi, "The K-means with mini batch algorithm for topics detection on online news," in *4th International Conference on Information and Communication Technology (ICoICT)*, Bandung, 2016.

[3] M. Erisoglu, N. Calis, and S. Sakallioglu, "A new algorithm for initial cluster centers in k-means algorithm," *Pattern Recognition Letters*, vol. 32, no. 14, pp. 1701–1705, 2011.

[4] K. M. Kumar and A. R. M. Reddy, "An efficient k-means clustering filtering algorithm using density based initial cluster centers," *Information Sciences*, vol. 418–419, pp. 286–301, 2017.

[5] M. E. Celebi, H. A. Kingravi, and P. A. Vela, "A comparative study of efficient initialization methods for the k-means clustering algorithm," *Expert Systems with Applications*, vol. 40, no. 1, pp. 200–210, 2013.

[6] D. Sculley, "Web-scale k-means clustering," in *Proceedings of the 19th international conference on World wide web*, New York, 2010, pp. 1177–1178.

[7] A. Feizollah, N. B. Anuar, R. Salleh, and F. Amalina, "Comparative study of k-means and mini batch k-means clustering algorithms in android malware detection using network traffic analysis," in *International Symposium on Biometrics and Security Technologies (ISBAST)*, Kuala Lumpur, 2014.

[8] Y. Xu, W. Qu, Z. Li, G. Min, K. Li, and Z. Liu, "Efficient k-means++ approximation with MapReduce," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 12, pp. 3135–3144, 2014, doi: 10.1109/TPDS.2014.2306193.

[9] D. Arthur and S. Vassilvitskii, "k-means++: the advantages of careful seeding," in *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, Philadelphia, 2007, pp. 1027–1035.

[10] M. A. P. Subali and P. Wijaya, "Sistem Question Answering untuk Bahasa Bali menggunakan Metode Rule-Based dan String Similarity," *Techno.COM*, vol. 20, no. 2, pp. 300–308, 2021.

[11] M. A. P. Subali and C. Fatichah, "Kombinasi Metode Rule-Based dan N-Gram Stemming untuk Mengenali Stemmer Bahasa Bali," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, vol. 6, no. 2, 2019.

[12] M. A. Fauzi, A. Z. Arifin, and A. Yuniarti, "Term Weighting Berbasis Indeks Buku dan Kelas untuk Perangkingan Dokumen Berbahasa Arab," *Lontar Komputer*, vol. 5, no. 2, pp. 435–442, 2014.

[13] S. I. Murpratiwi, I. G. A. Indrawan, and A. Aranta, "Analisis Pemilihan Cluster Optimal dalam Segmentasi Pelanggan Toko Retail," *Jurnal Pendidikan Teknologi dan Kejuruan*, vol. 18, no. 2, pp. 152–163.