International Journal of Industrial Innovation and Mechanical Engineering
Volume.1, No.3 Agustus 2024
e-ISSN: 3047-4515, p-ISSN : 3047-4507, Hal 10-18
DOI : https://doi.org/10.61132/ijiime.v1i3.38
Available online at: https://international.aritekin.or.id/index.php/IJIIME

Implementation of VGG6 for Image Processing of Aromatic Plants Using Python

Adriana Sari Aryani [1*], Irfan Wahyudin [2], Kotim Subandi [3]
[1,2,3] Program Studi Ilmu Komputer, FMIPA, Universitas Pakuan, Indonesia

Address: Jl. Pakuan, RT.02/RW.06, Tegallega, Kecamatan Bogor Tengah, Kota Bogor, Jawa Barat 16129
Email correspondence: adriana.aryani@gmail.com

Abstract. *Big Data Analytics has gained significant popularity in recent years, with many companies integrating it into their information technology roadmaps to enhance business performance. However, surveys indicate that Big Data Analytics demands substantial resources, including technology, costs, and talent, which often leads to failures in the initial stages of implementation. This study proposes a VGG6 architecture approach, intended to provide a framework for the initial implementation of Big Data Analytics. The study's outcomes include the implementation of the VGG6 architecture for processing images of aromatic plants using Python. Furthermore, this approach enabled the development of a Minimum Viable Product (MVP) solution that adheres to general Big Data principles, such as the 3Vs (Volume, Velocity, and Variety), and encompasses key technological components: 1) Data Storage and Analysis, 2) Knowledge Discovery and Computational Complexity, 3) Scalability and Data Visualization, and 4) Information Security.*

Keywords*: VGG6, aromatic plant image processing, Big Data Analytics.*

1. INTRODUCTION

Since the advent of the Web 2.0 and Web 3.0 eras, the internet has become one of the primary sources for decision-making in many organizations. This shift occurred because internet content contributors are no longer limited to companies or groups providing information about their profiles, as was the case during the Web 1.0 era. The Web 2.0 era marked the beginning of dynamic content that could be uploaded at any time, with ease and without requiring significant modifications to web pages, thanks to asynchronous technologies like AJAX. This was followed by the Web 3.0 era, which saw a change in user behavior from being passive visitors of web pages to becoming more active participants.

These developments also inspired scientists to develop more advanced methods for processing and analyzing data more quickly. In 2004, the MapReduce method was introduced by Google engineers Jeffrey Dean and Sanjay Ghemawat to process large-scale data in parallel, and the Hadoop framework was created by Doug Cutting in 2005. Since then, MapReduce and Hadoop have become the standard for large-scale data processing. Subsequently, several Big Data frameworks emerged, inspired by MapReduce, one of which is Spark, developed in 2009.

However, the worldwide adoption of Big Data remains relatively low. This is evidenced by a NewVantage survey in 2018, which found that nearly 30% of surveyed companies had not implemented Big Data. Moreover, of the approximately 70% of companies that had implemented Big Data, 20% failed to derive any benefits from it. According to a survey

Received: Juni 12, 2024; Revised: Juni 28, 2024; Accepted: Agustus 11, 2024; Published: Agustus 12, 2024;

conducted by Espinosa et al. in 2019, four main issues contributed to the failure of Big Data implementation, with technology-related issues such as storage and tools being the most dominant, followed by management and talent-related issues.

In terms of algorithms, K-means is a popular clustering method in Big Data implementations (https://link.springer.com/article/10.1007/s10115-007-0114-2). K-means is a distance-based algorithm that divides data into several clusters, and it only works on numeric attributes. K-means attempts to partition existing data into one or more clusters or groups so that data with similar characteristics are grouped into the same cluster, while data with different characteristics are grouped into different clusters. The K-means algorithm essentially performs two processes: detecting the center of each cluster and finding the members of each cluster.

K-means has several drawbacks, one of which is that the determination of the centroid is stochastic, requiring multiple repeated trials to obtain a convergent final clustering solution. Several complementary algorithms for centroid determination have been developed to address this issue, resulting in various K-means algorithm variants, such as the Pillar Algorithm, K-means with Genetic Algorithm, K-means++, and Scalable K-means, an extension of K-means++. K-means++ is a relatively new and popular variant of K-means that is easy to implement, as it is available in several popular data analytics libraries such as scikit-learn in Python and tidymodels in R.

The goal of this research is to develop a Minimum Viable Product (MVP) solution for Big Data Analytics implementation. The outcome of this research is a framework that can be used for Big Data Analytics implementation with minimal resources. To illustrate this, we implemented a data lineage framework to acquire data from the internet in near real-time, which was then processed using the Scalable K-means clustering algorithm to identify the top headlines from various news portals in Indonesia. To achieve optimal results, this algorithm was run on the Apache Spark Big Data platform.

The benefit of this research is to demonstrate that Big Data implementation, which can positively impact an organization's business processes, can be initiated with minimal budget and effort, without compromising the quality of the analysis that is valuable for decision-making within an organization.

2. RESEARCH METHODOLOGY

This research began by examining the pain points of Big Data Analytics implementation, as identified in several surveys conducted by consulting firms, which were discussed in the first section. Following this, we introduced an approach using a Minimal

Viable Product (MVP) for the implementation of Big Data Analytics. According to a survey by Lenarduzzi and Taibi (2017), there are two main reasons why a Minimum Viable Product should be developed, especially in a startup company: to quickly utilize the product and to bring it to market as soon as possible. In this study, we focused on the first reason, considering that Big Data Analytics is a technique that integrates massive datasets, statistical and mathematical methods, and parallel computing technologies. The second reason was not applicable, as Big Data Analytics is not a product marketed to customers unless the company is a Big Data technology provider, which is beyond the scope of this research.

In terms of implementation, according to a survey conducted by Acharjya and Ahmed in 2016, Big Data Analytics must meet the following four categories: 1) Data Storage and Analysis, 2) Knowledge Discoveries and Computational Complexities, 3) Scalability and Visualization of Data, and 4) Information Security. We mapped the technologies that can be used to meet these categories and provided examples of their implementation to satisfy the MVP criteria. For the data object, we selected news content from the internet, based on several considerations to meet the 3Vs criteria of Big Data:

a. Volume : Internet content is freely available and virtually unlimited in size.
b. Variety: Internet content is available in various formats, including text, images, and videos. However, within the scope of this research, we chose to focus only on text data formats.
c. Velocity: The speed at which internet content grows is undeniably the fastest today.

3. RESULT

Based on the criteria described in the previous section, we have designed a data lineage, which serves as a representation of the CRISP-DM framework. The research will primarily focus on collecting data in the form of online news content from at least three major mainstream online media outlets in Indonesia. This data will undergo thorough preprocessing, which includes cleaning, transforming, and organizing the data to prepare it for subsequent analysis. Following this, a cluster analysis will be performed using the distributed processing framework provided by Apache Spark.

The programming language selected for this research is Python, which will be executed on a parallel computing platform with Apache Hadoop as the underlying infrastructure. The choice of this platform allows for efficient processing of large datasets, which is essential given the volume of news content being analyzed. The results of the cluster analysis will be rigorously evaluated using the Silhouette Function technique. This evaluation will help determine the optimal number of clusters that accurately represent the underlying news topics, ensuring the robustness of the analysis.

Furthermore, one of the key outputs of this research will be a web-based dashboard application. This application is designed to enable users to easily visualize and interact with the analysis results, providing an intuitive interface for exploring the identified clusters and their corresponding news topics. The research flowchart, which outlines the entire process from data collection to result visualization, is illustrated in Figure 1."

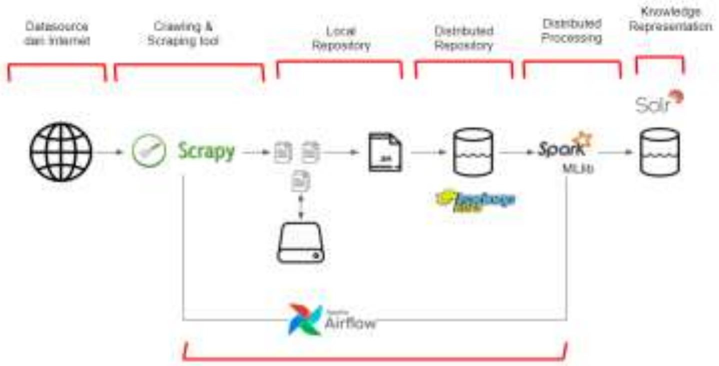
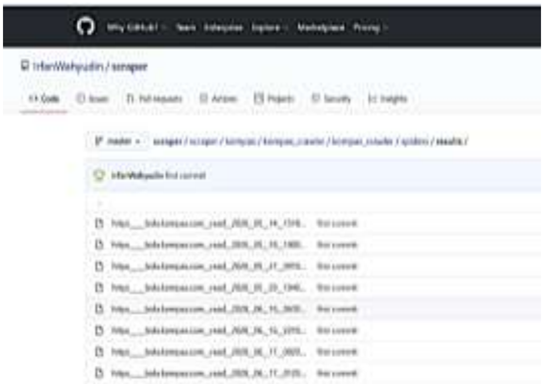


Figure 1. schedule process

In this research activity, the web scraping phase has been meticulously conducted. Web scraping refers to the process of extracting data from websites, which is a crucial step in gathering large volumes of relevant information for analysis. This process is executed using tools such as web scrapers, bots, web spiders, or web crawlers. Specifically, a web scraper is a program designed to systematically access a webpage, download its content, extract pertinent data, and then store this data in a structured format, such as a file or database, for further analysis.

During this phase, data scraping was focused on retrieving content related to the keywords: New Normal, Covid, and Vaccine. These keywords were selected to capture a wide range of relevant articles and discussions surrounding these topics. The web scraping was carried out on three major online media outlets: CNN, Kompas, and Detik, which are known for their extensive coverage and influence in the Indonesian media landscape.

The scraped data includes various types of information, such as headlines, publication dates, article bodies, and metadata, which will be used in the subsequent stages of data processing and analysis. This data collection process is critical in ensuring that the research is based on a comprehensive and representative dataset. An example of the data obtained through web scraping can be seen in Figure 2, which illustrates the structure and content of the collected data.



IrfanWahyudin / scraper

scraper / scraper / kompas / kompas_crawler / kompas_crawler / spiders / results /

IrfanWahyudin first commit

Figure 2. Scrapping process

Modeling was conducted using the Databricks platform, a cloud service offering (Platform as a Service - PaaS) specifically tailored for big data analytics. Databricks provides a collaborative and scalable environment that integrates seamlessly with Apache Spark, facilitating high-performance data processing and complex analytics. The platform's architecture supports both batch and real-time data processing, enabling efficient handling of large datasets. In this phase, Databricks was employed to perform advanced data modeling and analysis, leveraging its robust features for distributed computing and data management. The platform's interactive notebooks and advanced data visualization tools played a crucial role in developing and refining models, as well as in interpreting the results. Databricks also supports collaborative work by allowing multiple users to simultaneously work on and review data workflows, which enhances productivity and ensures accuracy.

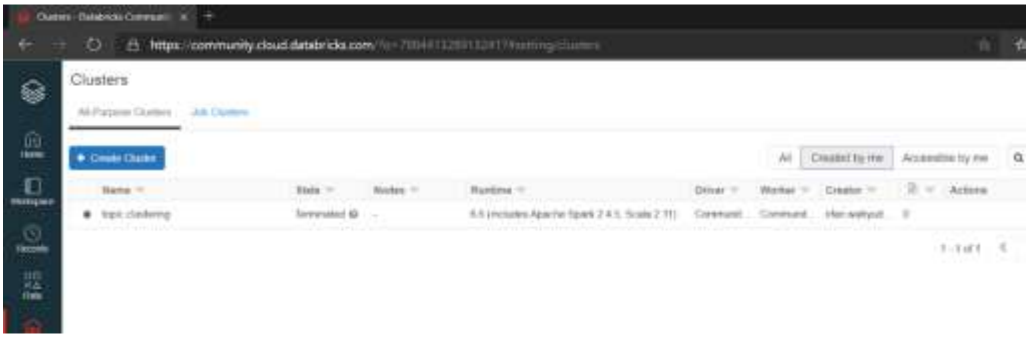
The user interface login screen for Databricks, which provides access to the platform's suite of features, is illustrated in Figure 3. This interface is designed to streamline user access to data management, algorithm execution, and result visualization functionalities. It allows users to efficiently navigate through their projects, execute complex queries, and manage computational resources. The platform's user-friendly design and powerful capabilities significantly contribute to the effectiveness of the modeling process, making it an invaluable tool for big data analytics

Sign In to Databricks Community Edition

irfan.wahyudin@unpak.ac.id

Figure 3. Interface login

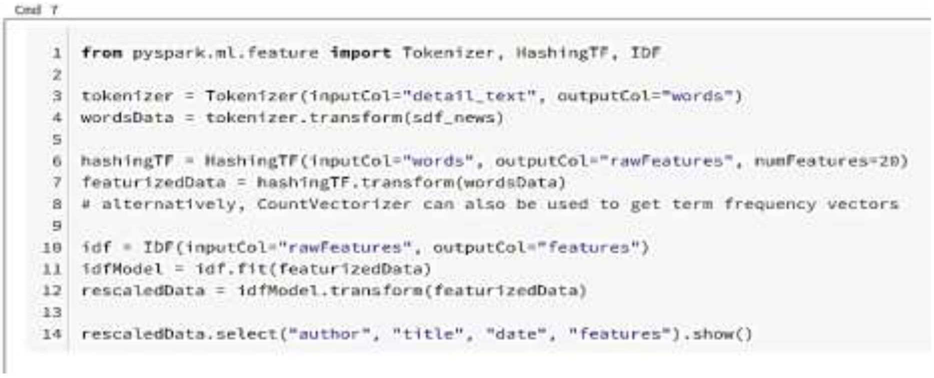
The Computing Platform will be accessible after a successful login. Once logged I n,users will encounter the main dashboard of the platform, which displays the datasets available for processing and analysis. These datasets are integral for performing clustering operations, which are essential for segmenting and analyzing the data based on various attributes. Within the Computing Platform, users can navigate through various modules and tools designed for data management, preprocessing, and modeling. The platform supports the importation of large datasets and provides functionalities for data cleaning, transformation, and visualization. The user interface of the computing platform, including the navigation panel and data processing tools, is illustrated in Figure 4. This interface allows users to interact with the datasets, configure clustering algorithms, and monitor the progress of data processing tasks. The design of the user interface aims to facilitate ease of use and efficient management of computational resources, contributing to an effective data analysis workflow."



Name	State	Nodes	Runtime	Driver	Worker	Creator	Actions
topic clustering	Terminated	-	[illegible]	[illegible]	[illegible]	[illegible]	

Figure 4. Computational interaction interface

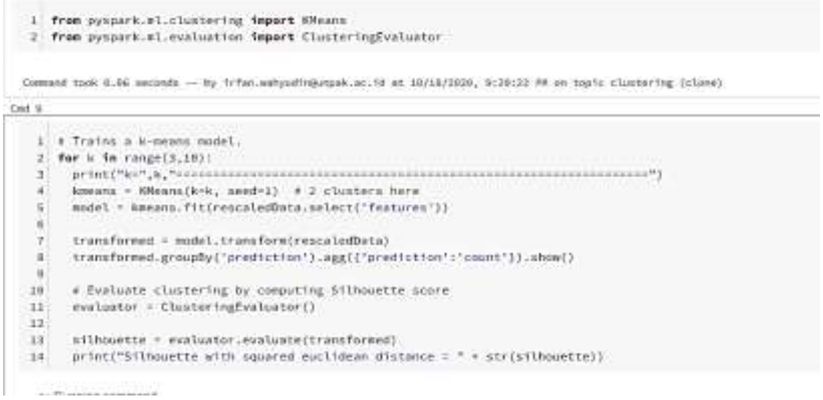
The preprocessing stage prior to clustering involves utilizing the Tokenizer and TF-IDF methods. Tokenization is the process of breaking down text into individual tokens, such as words or phrases, which are essential for transforming raw text data into a format suitable for analysis. The TF-IDF (Term Frequency-Inverse Document Frequency) method is then applied to quantify the importance of each token within the dataset by evaluating its frequency in individual documents relative to its frequency across the entire corpus. This preprocessing step is crucial for preparing the data for clustering algorithms, as it converts textual information into numerical features that can be effectively processed. The Tokenizer and TF-IDF methods help in extracting meaningful patterns and reducing dimensionality, which improves the performance and accuracy of the clustering analysis. A snippet of the source code used for the preprocessing stage, which includes the implementation of these methods, is provided in Figure 5. This code illustrates the steps involved in applying Tokenization and TF-IDF to the dataset, demonstrating how text data is transformed into a structured format ready for clustering."



```
Cmd 7

from pyspark.ml.feature import Tokenizer, HashingTF, IDF

tokenizer = Tokenizer(inputCol="detail_text", outputCol="words")
wordsData = tokenizer.transform(sdf_news)

hashingTF = HashingTF(inputCol="words", outputCol="rawFeatures", numFeatures=20)
featurizedData = hashingTF.transform(wordsData)
# alternatively, CountVectorizer can also be used to get term frequency vectors

idf = IDF(inputCol="rawFeatures", outputCol="features")
idfModel = idf.fit(featurizedData)
rescaledData = idfModel.transform(featurizedData)

rescaledData.select("author", "title", "date", "features").show()
```

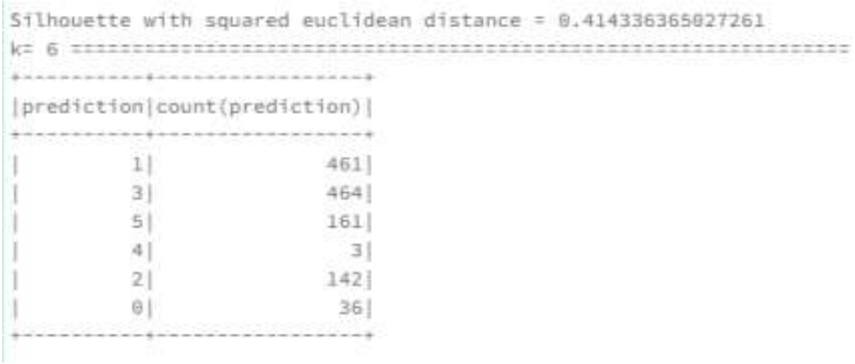
Figure 5. Preprocessing stage

Modeling with K-Means involves the creation of clusters following the preprocessing stage. K-Means is a widely used clustering algorithm that groups data points into \(K \) clusters based on their features. The objective is to minimize the variance within each cluster and maximize the variance between clusters. A snippet of the source code used for the K-Means modeling stage is displayed in Figure 6, demonstrating the implementation of this algorithm. While a high average silhouette score is a useful metric for evaluating cluster quality, it does not always guarantee the optimality of the clusters. It is crucial to also examine the distribution of data across clusters to ensure that the clusters are not only well-separated but also have uniformly distributed populations. This means checking whether the number of data points in each cluster is balanced, which contributes to the effectiveness and practical utility of the clustering results.In our analysis, (K = 6) was determined to be the most appropriate number of clusters. This choice was supported by the observation that (K = 6) provided the highest silhouette score and also resulted in a uniform distribution of data across the clusters. The uniformity of cluster populations helps in achieving more stable and reliable clusters. The average silhouette scores for different values of K are presented in Figures 6 and 7, which illustrate the comparison and support the conclusion that (K = 6) is the optimal number of clusters for this dataset."



```
from pyspark.ml.clustering import KMeans
from pyspark.ml.evaluation import ClusteringEvaluator

Command took 0.06 seconds -- by irfan.wahyudin@utpak.ac.id at 10/18/2020, 9:20:22 PM on topic clustering (clone)
Cmd 9

# Trains a k-means model.
for k in range(3,10):
  print("k=",k,"================================================================")
  kmeans = KMeans(k=k, seed=1)  # 2 clusters here
  model = kmeans.fit(rescaledData.select('features'))

  transformed = model.transform(rescaledData)
  transformed.groupBy('prediction').agg({'prediction':'count'}).show()

  # Evaluate clustering by computing Silhouette score
  evaluator = ClusteringEvaluator()

  silhouette = evaluator.evaluate(transformed)
  print("Silhouette with squared euclidean distance = " + str(silhouette))
```

Figure 6. K-Mean



```
Silhouette with squared euclidean distance = 0.414336365027261
k= 6 ============================================================
+----------+-----------------+
|prediction|count(prediction)|
+----------+-----------------+
|         1|              461|
|         3|              464|
|         5|              161|
|         4|                3|
|         2|              142|
|         0|               36|
+----------+-----------------+
```

Figure 7. Comparison Silhoutte

4. CONCLUSION

This research on Cluster Analysis of Online News Content utilizes the K-Means algorithm to acquire and process data from the internet in near real-time. The K-Means algorithm is employed to perform clustering, a method aimed at partitioning data into distinct groups based on similarity. Specifically, K-Means seeks to group data points with similar characteristics into the same cluster, while segregating those with differing characteristics into separate clusters. The algorithm operates through two core processes: determining the centroids of each cluster and assigning data points to the nearest centroid.

The primary objective of this study is to develop a framework capable of near real-time data acquisition from online news sources and to apply the K-Means clustering algorithm to identify emerging trends and topics across various news portals in Indonesia. The research methodology includes several key stages: web scraping to collect relevant news content, preprocessing the data using Tokenizer and TF-IDF methods to convert textual information into a suitable format for analysis, and conducting clustering with K-Means.

Evaluation of the clustering outcomes indicates that K=6 only yields the highest silhouette score, reflecting optimal cluster separation, but also achieves a uniform distribution of data points across clusters. This uniformity in cluster size is significant for ensuring the reliability and validity of the clustering results. Thus, K=6 is determined to be the most appropriate number of clusters for this dataset, balancing the quality of the clustering solution with an equitable distribution of data."

REFERENCE

Big Data Redux: New Issues and Challenges Moving Forward https://scholarspace.manoa.hawaii.edu/bitstream/10125/59546/0106.pdf

Campaign Produk Perbankan", Laporan Akhir Hasil Penelitian Bersumber Dana Hibah Dikti, Bogor September 2018.

Dean and Ghemawat, (2005), "MapReduce: Simplified Data Processing on Large Clusters*",* Google Research, Google Inc. California, USA.

International Journal of Recent Technology and Engineering 8 (2S7), 25-29.

Mougalas R. 2005. *Big Data Analytics*. O'Reilly Media. Sebastopol, USA.

Pengelompokkan Penjualan Produk", Jurnal Media Infotama, Vol. 12, No.2, pp. 148-157, 2016.

Tosida et al. 2019. A hybrid data mining model for Indonesian telematics SMEs empowerment. IOP Conference Series: Materials Science and Engineering. IOP Science.

Wahyudin dan Salmah A. 2019. *A Big Data Architecture to Support Bank Digital Campaign.*

Wahyudin dan Salmah, "Perancangan Teknologi Big Data Untuk Mendukung Digital

Wahyudin, et al. 2016. *Cluster analysis for SME risk analysis documents based on Pillar K-Means*. Telkomnika, Ahmad Dahlan University

Yulia Darmi dan Agus Setiawan, "Penerapan Metode Clustering K-Means Dalam