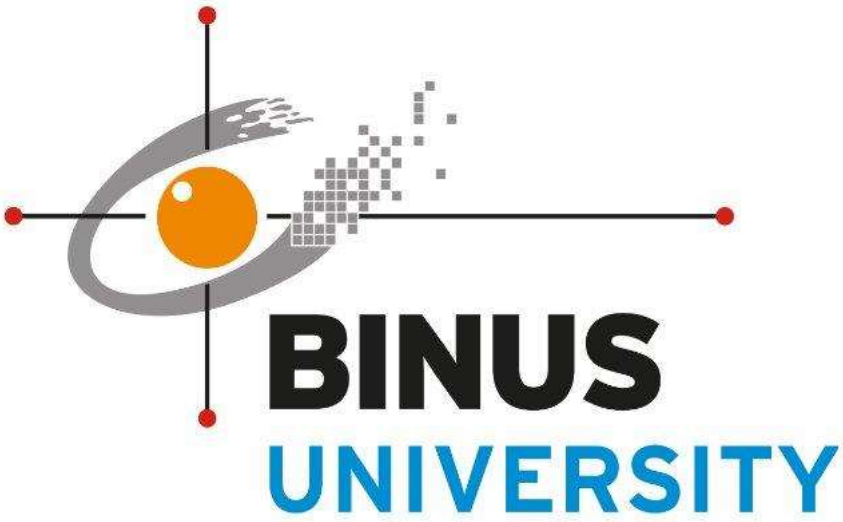
LAPORAN DAN DOKUMENTASI PROJEK "FASHION TREND FORECASTING"



Anggota Kelompok

Timotius Jason Lee	2301879955
Sovia Agustina Kusuma Ikka	2301956531
Vangelia Peace Palijama	2301961084

School of Computer Science
Program Studi Teknik Informatika
Universitas Bina Nusantara
2020/2021

BAB 1. PENDAHULUAN

Dalam keseharian hidup, manusia tidak terlepas dari pakaian. Pakaian terus berkembang seiring dengan berkembangnya zaman. Eksistensi desainer busana/ *fashion* telah ada sejak abad ke-19, sebagai perancang model busana. Karya desainer pertama di dunia, ada tepatnya pada 1858 oleh Charles Frederick Worth. Hingga kini, para desainer fashion terus berlomba-lomba untuk merancang model busana yang unik dan berbeda, yang menarik perhatian dunia.

Dewasa ini, perilaku konsumen dalam membeli busana telah menjadi indikator utama dalam mencari dan menentukan tren fashion. Kebanyakan desainer fashion bergantung pada prediksi apa yang akan menjadi trend di seluruh dunia [1]. Mulai dari beberapa atribut fashion yang paling berpengaruh dalam proses prediksi trend itu sendiri, seperti warna, style, pola, material, dan shape [2]. Dilansir dari jdinstitute.co mengenai pentingnya prediksi fashion, menyatakan bahwa prediksi fashion merupakan hal yang mendasar bagi dunia design. Hal ini menjadi titik pacu para designer untuk membuat karya-karyanya.

Oleh karena itu, implementasi teknologi mulai dimanfaatkan untuk melakukan prediksi akan trend pakaian yang akan berkembang di masa depan. Teknologi ini berbasis Komputer dimana menggunakan implementasi *machine learning* untuk membantu para desainer ataupun pengguna memprediksi trend *fashion* ke depannya [3].

Maka dari itu, dengan adanya prediksi *fashion trend* di masa depan, maka para desainer dan pembuat pakaian mampu menyesuaikan dengan hasil prediksi dari hasil *Machine Learning* tersebut. Sehingga, membantu industri maupun pengguna untuk menyesuaikan kebutuhan bahan baku serta desain pakaian jadi yang sesuai dengan prediksi trend yang diproses oleh komputer.

BAB 2. TINJAUAN PUSTAKA

2.1. Pengertian Sentiment Analysis

Sentiment Analysis merupakan cabang teknologi dari bidang komputer Natural Language Processing dimana Sentiment Analysis merupakan metode yang digunakan untuk menentukan suatu tulisan cenderung bersifat positif atau negative.

Penggunaan dari Sentiment Analysis ini sangat berguna untuk menentukan persepsi orang melalui kata-kata atau text yang mereka tulis. Oleh karena itu metode ini seringkali digunakan untuk menentukan rating dari suatu produk ataupun jasa karena dinilai mampu untuk mengklasifikasikan feedback pengguna berbentuk positif atau negative.

2.2. Kelebihan dan Kekurangan Sentiment Analysis

Metode Sentiment Analysis ini memiliki kelebihan seperti:

- Kemampuan untuk mengetahui persepsi pengguna melalui text
- Kemampuan untuk dijadikan tolak ukur evaluasi
- Kemampuan untuk di implementasikan dalam segala jenis text

Namun, metode Sentiment Analysis ini juga memiliki kelemahan yaitu adanya kesalahan intepretasi maksud dari pengguna. Karena terkadang penggunaan Sentiment Analysis tidak selalu seraturs persen tepat dalam mengiterpretasikan persepsi pengguna.

2.3. Sistem Kerja Sentiment Analysis

Cara Kerja Sentiment Analysis bisa dibagi menjadi tiga bagian besar yaitu:

a.) Pertama, komputer mengklasifikasikan data yang dinilai sebagai pendapat dari suatu tulisan. Ada tiga klasifikasi dalam metode analisis yaitu:

- *Machine learning*: mengenali persepsi seseorang dalam sebuah tulisan.
- *Lexicon-based*: menilai skor polaritas dari berbagai kata untuk mengetahui perspektif pengguna terhadap suatu topik.
- Campuran: Penggabungan antara *Machine learning* dengan *Lexicon-Based*, jarang ditemukan tetapi efektif dalam melakukan klasifikasi.

b.) Evaluasi

Sesudah data melalui proses klasifikasi, proses berikutnya adalah evaluasi menggunakan metrik seperti Precision, Recall, F-score, dan Accuracy.

c.) Visualisasi data

Berikutnya adalah visualisasi data, dimana dapat dilakukan sesuai kebutuhan orang yang menggunakan data ini. Kebanyakan orang menggunakan teknik yang familiar seperti histogram, matriks atau grafik.

BAB 3. TAHAP PELAKSANAAN

3.1.s Cara Kerja Fashion Trend Forecasting

Dari 19.652 ulasan/reviews yang terdapat pada dataset, kami hanya mengambil 5000 reviews untuk digunakan untuk tahap pengujian guna menghindari kasus overfit. Fitur-fitur yang kami gunakan antara lain “Clothing ID” (5nalisa5 unik yang membedakan tiap produk), “Review Text” (ulasan user terhadap produk tersebut), “Rating” (ulasan garis besar user dengan angka; rentang angka dari 1 sampai dengan 5), dan “Item Success” (hasil kalkukasi dari fitur rating).

Setelah menetapkan dataset yang akan dibawa ke tahap uji, dataset akan diproses terlebih dahulu. Pemrosesan data awal ini dinamakan sentiment analysis. Ulasan user yang berbentuk kalimat akan disaring dengan menghilangkan angka, tanda baca, tokenisasi (proses pembagian teks menjadi bagian-bagian tertentu), stopwords (kata umum yang sering muncul namun dianggap tidak memiliki makna), dan diakhiri dengan proses stemming (penguraian bentuk kata menjadi bentuk kata dasarnya).

Kelima tahap awal tersebut akan menghasilkan kalimat yang sudah dapat digunakan untuk pengolahan selanjutnya. Kata-kata yang terdapat pada tiap ulasan user akan diubah dari bentuk akta menjadi angka dengan fungsi vectorizer. Vectorizer yang digunakan pada tahap uji ini ada dua, diantaranya Count Vectorizer dan TF-IDF Vectorizer. Count Vectorizer akan mengubah data berbentuk teks ke dalam angka-angka dalam bentuk matriks, agar dapat diproses oleh komputer. Sementara itu, TF-IDF Vectorizer merupakan proses term-weighting atau proses pemberian bobot kata pada dokumen yang membandingkan jumlah kemunculan sebuah kata dalam sebuah dokumen dengan jumlah dokumen yang mana kata tersebut muncul.

Setelah proses term-weighting selesai, maka data teks yang telah diekstrak menjadi angka akan dilakukan training dengan menggunakan pengklasifikasi (classifier) One vs Rest. Hal ini dikarenakan kami ingin mengklasifikasi prediksi rating dari hasil 5nalisa ulasan yang user berikan, yaitu 1 sampai dengan 5. Tahap ini memberikan kami hasil tertinggi sebesar 63,4% menggunakan pengklasifikasi Support Vector Machine.

Setelah rating diprediksi menggunakan SVM, maka tahap selanjutnya ialah memprediksi apakah produk tersebut akan sukses atau tidak (trending atau tidak trending). Faktor yang memengaruhi/berhubungan dengan keberhasilan produk ialah jumlah ulasan, rata-rata rating produk, dan hubungan terbalik dengan standar deviasi produk. Arti dari hubungan terbalik ini ialah, jika target pasar menyukainya, tentu akan banyak user/pelanggan yang cenderung

memberikan nilai tinggi tanpa banyak variasi rating pada produk tersebut. Dengan pemahaman ini, didapatkan rumus "Rating Factor" yaitu:

$$Rating\ Factor = \frac{1 - e\left(\frac{Reviews\ proportion * Rata - rata\ rating}{Standar\ deviasi\ dari\ rating}\right)}{1 + e\left(\frac{Reviews\ proportion * Rata - rata\ rating}{Standar\ deviasi\ dari\ rating}\right)}$$

di mana, "Proporsi ulasan" adalah proporsi ulasan yang diterima item dari kumpulan data lengkap.

Terdapat juga rumus "Rating Strength" (untuk menghilangkan hasil minus dari Rating Factor) yaitu:

$$Rating\ Strength = (Rating\ Factor)^2$$

Rating Strength ini akan menghasilkan nilai antara 0 sampai dengan 1, yang mana nantinya akan berhubungan dengan nilai cutoff (pada pengujian ini, kami menggunakan angka 0.00037).

Cutoff merupakan angka yang bersifat sebagai batas penentu apakah produk tersebut akan trending atau tidak. Jika nilai "Rating Strength" kurang dari nilai cutoff, maka produk akan dianggap sebagai produk yang tidak trending, begitu pula sebaliknya.

BAB 4. DOKUMENTASI

4.1. Library yang Digunakan

```
import nltk
import re
import string
from nltk.tokenize import WordPunctTokenizer
from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer
from nltk.stem import WordNetLemmatizer
from sklearn.feature_extraction.text import CountVectorizer as cv
from sklearn.feature_extraction.text import TfidfVectorizer as tfidf
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score

import numpy as np
import pandas as pd
```

4.2. Algoritma yang Digunakan

a) Count Vectorizer

→ Di dalam dunia NLP, Count Vectorizer termasuk ke dalam *feature extraction*. Di mana ia melakukan *tokenize* pada text bersama dengan tahap preprocessing. Teknik ini mengubah data berbentuk teks ke dalam angka-angka dalam bentuk matriks, agar dapat diproses oleh komputer.

b) TF-IDF Vectorizer

→ *Term Frequency-Inverse Document Frequency Vectorizer* merupakan proses term-weighting atau proses pemberian bobot kata pada dokumen. Berbeda dengan *count vectorizer*, teknik ini membandingkan jumlah kemunculan sebuah kata dalam sebuah dokumen dengan jumlah dokumen yang mana kata tersebut muncul.

c) Support Vector Machine (One vs. Rest Classifier)

→ SVM merupakan algoritma *supervised learning* yang dapat menangani permasalahan klasifikasi dan regresi secara linear maupun non-linear. Algoritma ini memaksimalkan jarak antar kelas untuk mencari *hyperplane* terbaik. *Hyperplane* sendiri merupakan fungsi yang dapat digunakan untuk memisahkan kelas-kelas dengan dimensi rendah (dua) sampai tinggi. Titik-titik/objek yang berada pada atau di luar dan

paling dekat dengan garis margin disebut sebagai *support vector*. Dalam SVM, *support vector* ini berguna dalam perhitungan untuk menemukan hyperplane terbaik/yang paling optimal untuk model yang diuji.

4.3. Hasil/Evaluasi Keberhasilan Project

a) Hasil klasifikasi rating berdasarkan bobot kata dalam dokumen/*vectorizer*:

SVM Classifier

1. CountVectorizer() Features

```
clf_cv = ovr(SVC()).fit(xcv_train, y_train)
cv_pred = clf_cv.predict(xcv_test)
```

```
print(f"SVM Classifier using CountVectorizer(): {accuracy_score(y_test, cv_pred)}")
```

```
SVM Classifier using CountVectorizer(): 0.634
```

b) Evaluasi model dengan RMSE dan akurasi ("Trending" / "Not trending")

MODEL ACCURACY

```
from sklearn import metrics
# print("nyoba: %r" % np.sqrt(np.mean((p) ** 2)))
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(yf, xf)))
```

```
Root Mean Squared Error: 0.4289522117905443
```

```
print('Accuracy Score:', np.sqrt(metrics.accuracy_score(yf, xf)))
```

```
Accuracy Score: 0.9033271832508971
```

4.4. Aplikasi

Berikut tautan terkait code dan penjelasan tentang pengaplikasian program yang kami buat:
https://deepnote.com/project/Untitled-Python-Project-ntiSnl4aSnqc9qpYYk3rzg/%2FFashionTrendForecasting.ipynb

4.5. Kesimpulan

Setelah proses pembuatan model selesai, didapatkan hasil akhir terbaik adalah dengan menggunakan CountVectorizer dengan SVM Classifier. Hal ini dapat dilihat dari akurasi yang lebih tinggi dari vectorizer TF-IDF. Hasil pembobotan kata/*vectorizer* ini nantinya akan diklasifikasi menjadi lima kelas, yaitu kelas 1 sampai dengan 5 (rentang *rating*). Pada akhirnya, model mampu memprediksi apakah suatu produk akan trending atau tidak dari review setiap produk yang didapatkan.

4.6. Daftar Pustaka

[1] S avitrie, D. (2008). Kebanyakan desainer fashion bergantung pada prediksi apa yang akan menjadi trend di seluruh dunia. Universitas Indonesia Fakultas Ekonomi. Published.

[2] Sastra Permata, N., Maria, A., & Asih, S. (n.d.). Identifikasi Atribut -Atribut yang Paling Berpengaruh Dalam Memprediksi Tren Fashion

[3] Chang, A. A., C., D., Ramadhan, J. F., Adnan, Z. K. S., Kanigoro, B., & Irwansyah, E. (2021). Fashion Trend Forecasting Using Machine Learning Techniques: A Review. School of Computer Science, Bina Nusantara University. Published.

[4] Vinay Arun, "Predict Product Success using NLP models - Towards Data Science," *Medium*, May 08, 2018. https://towardsdatascience.com/predict-product-success-using-nlp-models-b3e87295d97

[5] Github: https://github.com/vinayarun/BUSINESS-USE-CASE-FOR-NLP