

LAPORAN
TUGAS BESAR MATAKULIAH
PEMROSESAN BAHASA ALAMI

KLASIFIKASI SENTIMEN PADA ULASAN PRODUK DI WEBSITE
FEMALE DAILY



Disusun Oleh:

Chlaudiah Julinar Soplero L. 1301150434

Vina Fadriani Effendi 1301154560

PROGRAM STUDI S1 TEKNIK INFORMATIKA
FAKULTAS INFORMATIKA
UNIVERSITAS TELKOM
AGUSTUS 2018

ABSTRAK

Klasifikasi sentimen merupakan hal yang menjadi sorotan dalam bidang ilmu Pemrosesan Bahasa Alami. Sering sekali didapati bahwa kelas dari sebuah ulasan tidak sesuai dengan isi yang dituliskan. Sistem ini merupakan salah satu sistem klasifikasi sentimen dengan data berupa ulasan produk yang ada pada situs Female Daily. Data ulasan-ulasan tersebut diolah melalui beberapa metode *preprocessing*, seperti penghapusan *stop word*, *stemming*, *tokenization*, pengolahan *punctuation*, pengolahan *emoji*, dan penanganan *negative words*. Setelah data melalui preprocessing, hasilnya akan di klasifikasi dengan metode Naive Bayes. Untuk ekstraksi fitur sistem ini memiliki 2 jenis ekstraksi fitur, yaitu *Contains Words* dan TF-IDF. Masing-masing ekstraksi fitur memiliki tingkat akurasi yang cukup tinggi, yaitu kurang lebih 80%.

Kata Kunci : Klasifikasi Sentimen, *Preprocessing*, Naive Bayes, *Contains Words*, TF-IDF

BAB 1. PENDAHULUAN

1.1. Latar Belakang

Saat ini, pengguna internet sering menggunakan forum sebagai pertimbangan untuk membeli sebuah barang. Female Daily merupakan salah satu situs yang ada di internet yang berisikan semua tentang produk yang berkaitan dengan wanita. Pada situs ini, pengguna dapat membaca ulasan dan memberikan ulasan produk-produk yang ada.

Umumnya, untuk memberikan sebuah ulasan produk, pengguna diminta untuk mengisi kalimat ulasan dan hasil rating yang biasanya berupa bintang. Pengguna diminta untuk menuliskan pengalamannya menggunakan produk dan menilai apakah pengalaman tersebut pantas diberikan berapa bintang. Tetapi, pengguna juga tak jarang memberi hasil ulasan dan rating yang justru berbanding terbalik. Contohnya, sering pengguna menuliskan betapa puasny ia dengan produk tersebut tetapi malah memberikan bintang berjumlah 1, dimana yang artinya ulasan tersebut negatif.

Dalam tugas besar ini, penulis membuat sebuah sistem yang dapat mengklasifikasi ulasan pengguna kedalam kelas negatif dan positif. Sistem ini dapat membaca sebuah inputan berupa ulasan produk. Dari hasil model klasifikasi yang telah dilatih, sistem akan mengeluarkan hasil klasifikasi kelas yaitu apakah ulasan tersebut termasuk kedalam kelas negatif atau kelas positif.

BAB 2. DATASET

2.1. Deskripsi Dataset

Dataset yang digunakan pada sistem ini sebagai data latih ataupun data validasi adalah merupakan ulasan yang ada pada situs Female Daily (femaledaily.com). Ulasan yang penulis gunakan adalah tulisan asli dari banyak pengguna situs ini. Dataset yang digunakan berasal dari banyak produk tetapi dari satu kategori, yaitu *facial wash*.

Contoh satu ulasan dari dataset yang penulis gunakan adalah sebagai berikut, “tertarik nyobain karna baru aja lepas skincare dr klinik kecantikan . awal pemakaian tuh sumpah lembut ,ga kering dan kaya kelembaban kejaga bangettt dan ga bikin merah2 atau gatal2. tapi pas dirutinin setiap pagi ,siang/sore(kalo pergi aja), dan sbelum tidur malah bikin kulit merah2 gatal didagu dan sumpah sakit2 jerewi w:((((. akhirnya stopp bener2. dan menurut aku kalo pake ini kayanya harus pake first cleanser juga deh ,karna kaya kurang bersih gitu. apalagi kalo kalian pake obat jerawat malemnya, dan obat jerawatnya berbahan sulfur gitu di muka. bener2 gabisa bersihin dan harus nuangin banyak..... big no deh sejak pake ini malah ada jerawat gede nononono padahl dulu jerawat cuma yg bump bump doang :(ada yg sama ga kaya aku?:(mau dong rekomendasi second cleanser yg oke. ga bikin kulit kering kerontang karna kulit ku yg kombinasi susah banget cari skincare yg cocok:(kalo ga ketarik bngt mukanya, ya jadi kilang minyak”. Pada contoh ulasan ini dapat dinilai bahwa pengguna situs ini menuliskan ulasan tidak dengan bahasa baku dan seringnya menggunakan banyak simbol.

2.2. Analisis Dataset

Dari total 500 dataset, 250 merupakan data dengan kelas positif dan 250 data lain merupakan kelas negatif. Hampir semua ulasan ditulis tidak menggunakan bahasa yang baku bahkan banyak ulasan berisi beberapa kalimat dengan bahasa Inggris. Pengguna juga banyak menggunakan simbol sebagai penanda emosi. Selain itu, atribut panjang teks tidak dapat dijadikan penanda kelas sentimen ulasan-ulasan tersebut.

Atribut yang paling berpengaruh terhadap sentimen kelas sejauh analisis penulis adalah simbol emoji yang digunakan pengguna. Tetapi karena tidak semua ulasan mengandung emoji, maka faktor lain yang paling berpengaruh adalah kata-kata negatif seperti, ‘tidak’, ‘gak’, ‘ga’, dan lain-lain.

BAB 3. ANALISIS FITUR DAN METODE KLASIFIKASI

3.1. Bahasa Pemrograman dan *Library*

Bahasa yang penulis gunakan adalah python. Untuk membangun sistem ini library yang digunakan, sebagai berikut

- a. csv
- b. random
- c. re
- d. sys
- e. nltk
- f. sastrawi
- g. sklearn

3.2. Dataset

Dataset yang digunakan merupakan data ulasan dengan ekstensi CSV(.csv). Dari dataset tersebut, penulis akan membagi dataset menjadi 80% sebagai data latih dan 20% sebagai data validasi.

3.3. Praproses Data

Sebelum mengklasifikasi ulasan, penulis melakukan praproses data. Ada beberapa metode yang digunakan untuk praproses data, yaitu:

- a. Penghapusan *Stop Word*
Pada metode ini, ulasan akan dihilangkan kata-kata yang umum dan sering muncul yang tidak memiliki pengaruh yang signifikan pada kalimat. Contoh *stop word* yang akan dihilangkan dari ulasan-ulasan pada dataset adalah ‘aku’, ‘ada’, ‘adalah’, dan lainnya.
- b. Penanganan *Emoji*
Pada metode ini, diberikan sebuah tanda dimana jika ada emoji bahagia (‘:D’ atau ‘:’) maka tanda yang diberikan adalah ‘POS’ sedangkan jika ada emoji sedih (‘:(’ atau ‘T_T’) maka akan diberi tanda ‘NEG’.
- c. Penanganan *Punctuation*
Pada metode ini, ulasan akan dihilangkan tanda bacanya.
- d. *Stemming*
Stemming merupakan sebuah metode yang digunakan untuk mengubah kata menjadi kata dasar dengan menghilangkan afiks dan sufiks pada kata. Untuk metode ini, penulis menggunakan *library* yang ada di python, yaitu Sastrawi.
- e. Penanganan *Negative Words*
Pada metode ini semua kata yang berada di depan kata negatif seperti, ‘tidak’, ‘ga’, ‘engga’, dan lain-lain akan diberikan penanganan yaitu penambahan ‘tidak_’

sebelum kata tersebut. Contohnya, jika ada kata ‘tidak cocok’ maka fungsi tersebut akan mengubah kata-kata tersebut menjadi ‘tidak tidak_cocok’. Penanganan ini berfungsi untuk membedakan makna kata ‘cocok’ jika diawali oleh kata negatif dan yang tidak diawali oleh kata negatif.

f. *Tokenization*

Fungsi ini berguna untuk membagi kalimat menjadi token. Token dapat dibentuk dalam kata-kata, frasa atau elemen bermakna lainnya. Pada sistem ini, metode *tokenization* menggunakan fungsi `word_tokenize` dari library `nltk.tokenize`.

3.4. Fitur dan Metode Klasifikasi

Pada Klasifikasi Sentiment untuk Review Produk disini, kelompok kami menggunakan dua fitur yang berbeda dengan proses ekstraksi yang berbeda juga. Berikut fitur yang digunakan:

- a. Fitur seluruh kata dalam setiap dokumen setelah dokumen melalui pra-proses
Setelah seluruh dokumen melalui pra-proses maka dapat dikatakan setiap kata dalam tiap dokumen merupakan kata yang penting dalam suatu dokumen, inilah yang menjadi alasan kelompok kami memilih fitur ini.
Pada fitur ini, pada saat proses ekstraksi fitur data validasi maka setiap kata dalam tiap dokumen akan di lihat pada setiap kata dalam dokumen data train. Apabila dalam satu dokumen data validasi, memiliki jumlah kata yang masuk ke dalam suatu dokumen pada data train dengan label positive, maka dokumen untuk data validasi tersebut akan memiliki label yang sama yaitu label positive.
- b. Fitur Term Frequency-Inverse Document Frequency (TF-IDF)
Pada fitur ini, kami menggunakan library yang telah disediakan oleh Scikit-Learn yaitu *`from sklearn.feature_extraction.text import TfidfVectorizer`*. TF-IDF di pilih sebagai salah satu fitur karena TF-IDF merupakan salah satu fitur klasifikasi sentiment yang sangat sering digunakan, oleh sebab itu diharapkan dengan menerapkan fitur TF-IDF maka akurasi dari hasil prediksi semakin tinggi.
Proses ekstraksi fitur dengan TF-IDF yaitu berdasarkan bobot dari tiap jumlah kata dalam tiap dokumen. Fitur-fitur yang berhasil di ekstraksi adalah fitur yang memiliki nilai lebih dari nilai minimum fitur tersebut berada di dalam suatu dokumen dan fitur yang memiliki nilai kurang dari nilai maksimum fitur tersebut berada di dalam suatu dokumen.

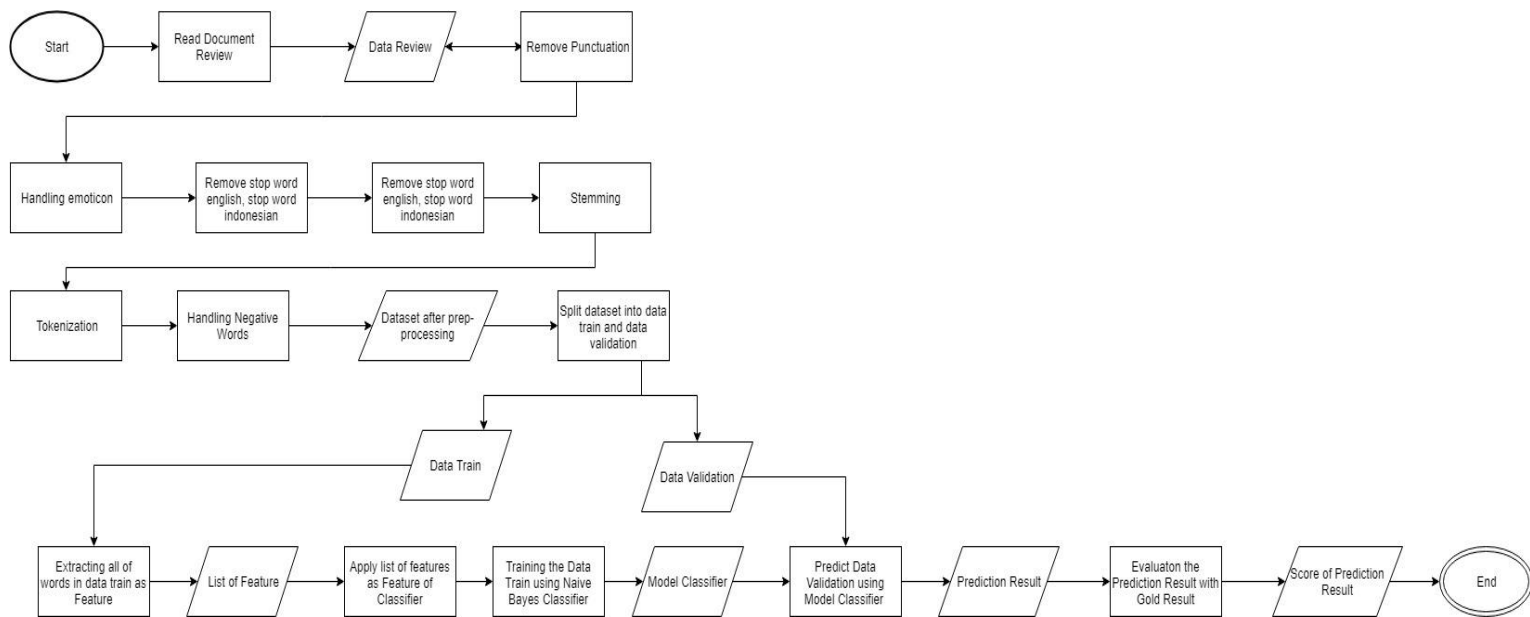
Setelah proses ekstraksi fitur telah selesai, maka dilakukan klasifikasi. Metode Klasifikasi yang kami pilih untuk digunakan adalah metode Klasifikasi menggunakan Naïve Bayes. Klasifikasi Naïve Bayes merupakan salah satu metode yang populer dalam melakukan klasifikasi sentiment teks. Kedua fitur akan di implementasikan ke dalam Naïve Bayes Classifier sehingga kami dapat menganalisis fitur yang memiliki tingkat akurasi yang paling sesuai untuk digunakan dengan Naïve Bayes Classifier.

BAB 4. IMPLEMENTASI KLASIFIKASI DAN ANALISIS HASIL KLASIFIKASI

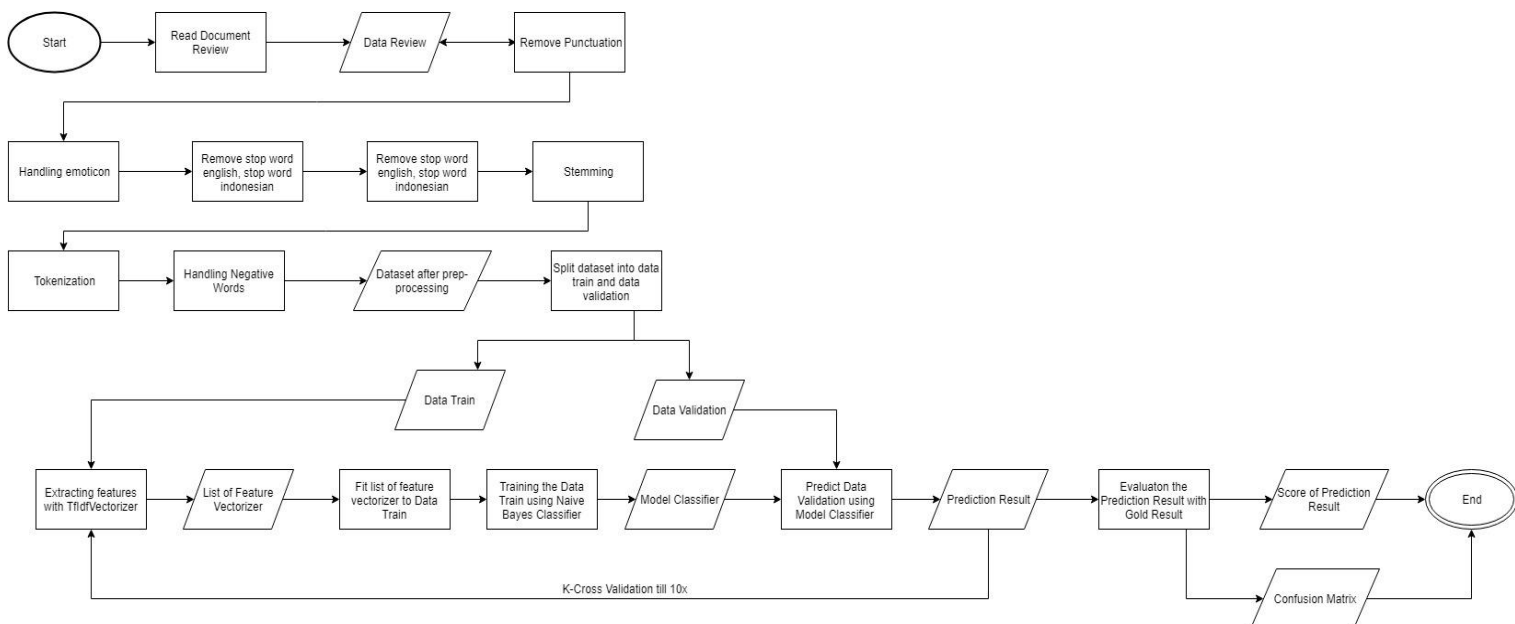
4.1. Diagram Alir Implementasi Klasifikasi

Pada Implementasi klasifikasi, terdapat dua implementasi yaitu implementasi klasifikasi pada Dataset dengan fitur klasifikasi yang digunakan adalah fitur seluruh kata pada setiap dokumen yang telah di pre-proses dan menggunakan TF-IDF juga sebagai fitur yang kedua.

4.1.1 Diagram Alir Implementasi Klasifikasi dengan Fitur Seluruh Kata pada Dokumen Pra-Proses



4.1.2 Diagram Alir Implementasi Klasifikasi dengan Fitur TF-IDF



4.2. Analisis Hasil Klasifikasi

4.2.1 Analisis Hasil Klasifikasi Fitur Contains Word (Seluruh Kata pada Tiap Dokumen)

Hasil klasifikasi dengan ekstraksi fitur menggunakan fitur ini yaitu sebesar 83%.

```
In [16]: num_true = 0
         for k, val in enumerate(validation_test):
             if val == sentiment_test[k]:
                 num_true += 1
         accuracy = (num_true / len(reviews_test)) * 100
         accuracy
```

Out[16]: 83.0

Hasil akurasi dikatakan memiliki akurasi yang cukup tinggi. Hal ini dikarenakan, proses data validasi telah dilakukan pre-proses terlebih dahulu sama seperti pada Data Train. Sehingga, kata negative seperti 'tidak bagus' juga ikut dilakukan preproses menjadi 'tidak' dan 'tidak_bagus', sehingga pada saat dilakukan pengecekan dengan dokumen yang memiliki label positive maka dokumen yang mengandung kata negative 'tidak bagus' tidak akan pernah termasuk kedalam dokumen dengan label positive.

Adapun dokumen yang pada hasil prediksinya memiliki label yang salah diakibatkan oleh masih banyak non-standard words yang terdapat dalam dokumen validasi yang bisa saja kata tersebut merupakan kata yang penting seperti kata 'bagus' untuk label positive tetapi ditulis menjadi 'bgs', dan pada data train sebelumnya seluruh dokumen dengan label positive belum pernah ada kata bagus ditulis dengan cara tersebut, maka kata 'bagus' yang merupakan kata berpengaruh akan diabaikan.

4.2.2 Analisis Hasil Klasifikasi Fitur TF-IDF

Hasil klasifikasi dengan ekstraksi fitur menggunakan TF-IDF adalah sebesar 80%. Berbeda 0.3% lebih rendah daripada fitur sebelumnya.

```
In [50]: print(totalMatNB, totalNB/len(data))
[[400. 350.]
 [450. 350.]] 0.8
```

Hasil akurasi TF-IDF dikatakan memiliki akurasi yang cukup tinggi, meskipun masih lebih dibawah hasil akurasi fitur sebelumnya. Hasil akurasi yang tinggi dapat diperoleh karena TF-IDF melakukan ekstraksi sesuai dengan fitur yang memiliki bobot terbaik dalam suatu dokumen. Sehingga, semakin tinggi bobot suatu fitur dari dokumen validasi didalam dokumen data train maka dapat dipastikan bahwa dokumen validasi tersebut sudah pasti memiliki label yang sama dengan dokumen train yang memiliki bobot fitur tinggi tersebut.

Tetapi, hal yang menyebabkan TF-IDF memiliki akurasi lebih rendah dibandingkan fitur contains words yaitu keberagaman kata pada tiap dokumen baik dalam data train maupun data validasi. Keberagaman kata tersebut mengakibatkan bobot suatu fitur tidak besar, sehingga berpengaruh dalam proses pemilihan label dokumen.

BAB 5. KESIMPULAN

5.1. Kesimpulan

Kesimpulan dari hasil pengerjaan tugas besar kami adalah dengan menggunakan Naïve Bayes sebagai Metode Klasifikasi yang digunakan dengan dua fitur yang berbeda adalah apabila menggunakan fitur Contains Words sebagai proses ekstraksi fitur maka harus dipastikan bahwa seluruh kata yang memiliki peranan penting untuk memutuskan label dalam suatu dokumen sudah di standarisasi dengan benar, sehingga kata tersebut dapat dipastikan sudah pernah muncul di dalam model klasifikasi yang dibangun.

Lalu, untuk klasifikasi dengan fitur TF-IDF sebagai proses ekstraksi fitur maka harus dipastikan bahwa keberagaman kata dalam dokumen dari data train maupun data validasi tidak terlalu tinggi. Contohnya adalah kata 'gue' yang dapat ditulis dengan berbagai cara seperti 'gua', 'gw', 'w', 'guw', dsb.

Kedua fitur yang digunakan memiliki kesamaan yang paling terlihat yaitu dokumen yang digunakan sebagai dokumen train maupun validasi harus sudah dilakukan pra-proses yang optimal sehingga dapat menghasilkan tingkat akurasi yang tinggi juga.

BAB 6. References

- [1] A. F. Hidayatullah and M. R. Ma'arif, "Pre-processing Tasks in Indonesian Twitter," in *Journal of Physics: Conference Series*, Edmonton, 2017.