

# 10\_KLASIFIKASI DOKUMEN DENGAN METODE NAIVE BAYES

*by* Lastri Suastri

---

**Submission date:** 11-Apr-2023 01:46AM (UTC+0700)

**Submission ID:** 2060788154

**File name:** 10\_KLASIFIKASI\_DOKUMEN\_DENGAN\_METODE\_NAIVE\_BAYES.pdf (370.74K)

**Word count:** 2219

**Character count:** 14203

## KLASIFIKASI DOKUMEN DENGAN METODE NAIVE BAYES TERHADAP PUTUSAN KASASI PENGADILAN TENTANG MERK

Sulastr<sup>1</sup>, Eri Zuliarso<sup>2</sup>

<sup>1</sup>Progdi Sistem Informasi, Fakultas Teknologi Informasi Universitas Stikubank Semarang

<sup>2</sup>Progdi Teknik Informatika, Fakultas Teknologi Informasi Universitas Stikubank Semarang

e-mail : <sup>1</sup>sulastr<sup>1</sup>@edu.unisbank.ac.id, <sup>2</sup>eri299@edu.unisbank.ac.id

### ABSTRAK

Aplikasi SIPP merupakan bagian dari Sistem Manajemen Informasi di Pengadilan, dimana masyarakat dapat mencari informasi yang diinginkan termasuk yang berhubungan tentang merk. Pada penelitian ini akan dilakukan klasifikasi terhadap putusan pengadilan tentang merk dengan menggunakan metode Naive Bayes dengan variabel klasifikasi putusan tolak atau kabul.

Dari hasil analisa yang dilakukan terhadap 50 data hasil putusan pengadilan tentang merk didapatkan model klasifikasi dengan tingkat akurasi sebesar 80%. Analisa text yang dilakukan terhadap putusan pengadilan terdapat 5 kata yang paling banyak muncul yaitu kasasi, pemohon, pengadilan, permohonan dan peninjauan.

**Kata Kunci** : Putusan Pengadilan, Merk, Klasifikasi, Naive Bayes

### 1. PENDAHULUAN

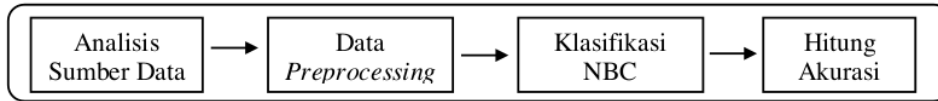
Klasifikasi termasuk teknik pembelajaran mesin atau biasa disebut supervised learning. Menurut Manning et al. (2008), supervised learning adalah proses pembelajaran mengenai ciri dari tiap-tiap kategori yang ada. Teknik ini membangun sebuah classifier yang mempelajari ciri tiap kategori berdasarkan dokumen latih yang dimiliki. Beberapa metode klasifikasi yang dapat digunakan dalam proses pembelajaran, yaitu multinomial naive bayes, multivariate Bernoulli model, Rocchio classification, k-Nearest Neighbor (KNN), dan support vector machine (SVM). Peningkatan dokumen akan mempengaruhi kinerja klasifikasi yang menyebabkan kerja sistem classifier akan semakin berat. Hal tersebut dikarenakan sistem klasifikasi mengambil isi dari uraian setiap dokumen. Salah satu cara untuk meningkatkan kinerja dari sistem klasifikasi dengan menerapkan teknik pemilihan fitur dokumen.

Penelitian yang dilakukan oleh Lorosae, Prakosa, Saifudin dan Kusrin (2018), analisis sentimen pada media sosial terkhusus twitter dalam pelayanan yang disediakan oleh penyedia jasa ekspedisi barang merupakan proses untuk mengompakkan respon seseorang terhadap pelayanan pengiriman barang yang sedang atau telah dilakukan. Permasalahan yang terjadi dalam penelitian ini terdapat kesulitan untuk menentukan opini yang bersifat positif, negatif ataupun netral. Penelitian ini bertujuan untuk mengembangkan aplikasi yang mampu melakukan analisis sentimen konsumen, dimana *Naive Bayes Classifier* digunakan sebagai metode klasifikasi. Sistem yang dibuat mampu mengklasifikasikan sentimen pada twitter kedalam sentimen positif, netral atau negatif. Hasil yang didapatkan dari akurasi *naive bayes* pada data uji positif memperoleh ketepatan 84%.

Penelitian yang dilakukan oleh Hanggara, Akhriza dan Husni (2017), sentimen dapat diketahui dari opini konsumen atas produk yang di sampaikan melalui website yang menjual produk tersebut. Beberapa aplikasi berbasis web untuk mengevaluasi sentimen konsumen sudah tersedia di internet. Analisis sentimen biasanya mengklasifikasi opini menjadi tiga kelas: Sentimen positif, negatif dan netral. Namun demikian aplikasi yang dapat melakukan proses pembuatan training dan testing set (dari data set opini konsumen terhadap produk), menganalisis sentimen konsumen sekaligus mengukur akurasi hasil analisis secara dinamis masih jarang tersedia. Penelitian ini bertujuan untuk mengembangkan aplikasi yang mampu melakukan analisis sentimen konsumen dengan fasilitas seperti yang disebutkan sebelumnya, dimana *Naive Bayes Classifier* digunakan sebagai metode klasifikasi. Hasil analisis pada testing set yang diperoleh dari Goodreads, Twitter, dan Tokopedia menunjukkan bahwa aplikasi yang dikembangkan mampu mengklasifikasikan opini ke dalam tiga kelas positif, negatif dan netral dengan tingkat akurasi rata-rata 70,43% - 88,03%.

### 2. METODOLOGI PENELITIAN

Obyek penelitian ini berupa dokumen putusan Pengadilan tentang merk yang diambil dari <https://putusan.mahkamahagung.go.id/pengadilan/mahkamah-agung/direktori/perdata-khusus/Merk>. Berikut adalah gambaran penyelesaian masalah dalam penelitian ini dapat dilihat pada gambar 1 :



Gambar 1. Alur Sistem Klasifikasi

Berikut adalah gambaran penyelesaian analisis klasifikasi putusan Pengadilan tentang Merk :

1. Analisis Sumber Data yaitu melakukan penentuan putusan apakah ditolak atau diterima secara manual terhadap data yang sudah diunduh. Jumlah data 50 data putusan terdiri dari 40 data latih dan 10 data uji. Untuk data latih terdiri dari 20 putusan terima dan 20 putusan kabul.
2. Data *Preprocessing*. Pada tahap ini, putusan yang telah dikumpulkan akan melalui beberapa tahap pemrosesan teks yang terdiri dari *case folding*, *tokenizing* dan *stopword removal*. Tujuan dari pemrosesan teks adalah supaya data yang didapat akan lebih terstruktur agar lebih mudah untuk dilakukan pengolahan data.
3. Klasifikasi Sentimen Naïve Bayes Classifier dengan menghitung probabilitas dari tiap putusan pengadilan untuk menentukan data mana yang termasuk dalam ditolak atau diterima.
4. Hitung Akurasi. Data yang diuji adalah data yang telah diketahui putusan pengadilannya, sehingga setelah dilakukan klasifikasi putusan pengadilan maka akan dilakukan perhitungan akurasi atau ketepatan sistem dalam melakukan klasifikasi. Dalam pengukuran akurasi metode yang digunakan dalam penelitian ini adalah *Confusion Matrix*.

### 3. HASIL DAN PEMBAHASAN

Tools yang digunakan untuk penelitian ini adalah Bahasa R, dimana langkah-langkahnya adalah :

1. Instalasi Packages yang dibutuhkan untuk proses klasifikasi yaitu `e1071`, `RtextTools`, `readxl`, `tm`, `dplyr`, `install.packages(caret, doMC, readxl, SnowballC, wordcloud, RColorBrewer)`.
2. Memanggil Library Packages yang berisi kumpulan fungsi R, data, dan kode.
3. Membaca Data dari file yang sudah didownload dari <https://putusan.mahkamahagung.go.id/pengadilan/mahkamah-agung/direktori/perdata-khusus/Merk>. Dan disimpan dalam bentuk excell yang sudah diberi text kabul atau tolak. Contoh datanya sebagai berikut :

Tabel1 Contoh putusan pengadilan tentang merk

Kabul	Bahwa Penggugat sangat berkeberatan atas pendaftaran merk Tergugat tersebut di atas karena merk tersebut di atas mempunyai persamaan pada pokoknya dengan merk sudah dikenal & Lukisan milik Penggugat
Kabul	Bahwa ternyata pendaftaran merk dagang Lavera yang dilakukan oleh Tergugat mempunyai persamaan pada pokoknya dengan merk dagang Lavera milik Penggugat yang sudah dikenal di berbagai negara (Pasal 6ayat (1a) Undang Undang Nomor 15 tahun 2001 tentang Merk)kabul
Tolak	Bahwa berdasarkan fakta-fakta yang disampaikan dalam uraian di atas sudah jelaslah bahwa Penggugat adalah Pencipta dan Pemegang Hak Cipta atas Seni Motif Kertas Nomor 020607
Tolak	Eksepsi Tentang Penggugat tidak mempunyai hak dan kapasitas untuk menggugat (non legitima persona standi in iudicio) Karena Penggugat tidak ada hubungan hukum dengan Tergugat atas merk dan Penggugat a quo tidak mempunyai hak tunggal/khusus untuk dan atas merk tersebut

4. Membuat Corpus. *Corpus* mewakili setiap kata dalam dokumen sebagai token atau fitur dan setiap dokumen sebagai vektor fitur. Disini urutan kata diabaikan dan fokus hanya pada jumlah kemunculan setiap kata. Data yang digunakan untuk membuat corpus adalah data `datamerk.xlsx` dengan variabel text. Sedangkan `df$label` digunakan untuk menentukan sebagai label klasifikasi apakah keputusan ditolak atau dikabulkan.

```
corpus <- tm_map(corpus, content_transformer(tolower))
corpus <- tm_map(corpus, removePunctuation)
corpus <- tm_map(corpus, removeNumbers)
corpus <- tm_map(corpus, removeWords, c(stopwordID))
corpus.clean <- tm_map(corpus, stripwhitespace)
```

Gambar 2 Script Corpus

Hasil corpus dapat dilihat pada gambar 3

```
> corpus
<<SimpleCorpus>>
Metadata: corpus specific: 1, document level (indexed): 0
Content: documents: 50
> ## <<VCorpus>>
> ## Metadata: corpus specific: 0, document level (indexed): 0
> ## Content: documents: 2000
```

Gambar 3 Hasil Corpus

Hasil Inspect Corpus adalah sebagai berikut :

```
> inspect(corpus[1:3])
<<SimpleCorpus>>
Metadata: corpus specific: 1, document level (indexed): 0
Content: documents: 3
```

[1] Bahwa berdasarkan alasan-alasan yg ada, dengan tidak perlu mempertimbangkan alasan peninjauan kembali lainnya, Mahkamah Agung berpendapat terdapat cukup alasan untuk mengabulkan permohonan peninjauan kembali yang diajukan oleh Pemohon Peninjauan Kembali PT GUDANG GARAH, TBK. Dan dinyatakan bahwa Merek Gudang Garah milik Penggugat adalah merek terkenal. Karena permohonan peninjauan kembali dari Pemohon Peninjauan Kembali dikabulkan, maka Termohon Peninjauan Kembali dihukum untuk membayar biaya perkara dalam semua tingkat peradilan dan pemeriksaan peninjauan kembali.

[2] Bahwa merek tersebut dinyatakan batal demi hukum karena hingga sekarang belum ada aturan pelarangan dari barang yang bukan sejenis, maka hal tersebut belum dapat diterapkan dalam putusan tersebut. Bahwa dengan belum adanya aturan pelaksanaan dari barang yang bukan sejenis maka ketentuan tersebut belum dapat dinyatakan berlaku. Bahwa keberatan tersebut dapat dibenarkan, oleh karena setelah meneliti secara seksama memori kasasi tanggal 7 Januari 2016 dan kontra memori kasasi tanggal 21 Januari 2016 dihubungkan dengan pertimbangan Juxde Facti. Bahwa berdasarkan pertimbangan tersebut di atas, Mahkamah Agung berpendapat, terdapat cukup alasan untuk mengabulkan permohonan kasasi dari Pemohon Kasasi LINAMATY HARDJONO tersebut dan membatalkan Putusan Pengadilan Niaga pada Pengadilan Negeri Jakarta Pusat Nomor 47/Pdt.Sus-Merek/2015/PN Niaga Jkt. Pst. tanggal 18 November 2015.

[3] Bahwa berdasarkan bukti-bukti yang diajukan Penggugat, terbukti Penggugat adalah penakai dan pendaftar pertama di dunia dan di Indonesia juga telah terlebih dahulu didaftarkan yaitu pada tanggal 8 Januari 1996 dengan Nomor 3

Gambar 4. Hasil Inspect Corpus

## 5. Text Processing

Tahapan berikutnya adalah tokenisasi yaitu mengubah semua huruf menjadi huruf kecil, menghapus tanda baca, menghapus angka dan menghapus kata – kata (*stopword removal*) yang tidak memiliki makna sehingga tidak berpengaruh terhadap proses analisis sentimen. *Stopword* yang digunakan merupakan modifikasi *stopword (custom stopwords)*. Penggunaan *custom stopwords* karena dalam RStudio belum tersedianya *stopword* untuk bahasa Indonesia. Script sebagai berikut :

```
df <- read_excel("D:/datamerek.xlsx")
stopwordID <- "D:/stoplist.txt"
cstopwordID <- readLines(stopwordID)

glimpse(df)
set.seed(1234)
df <- df[sample(nrow(df)), ]
glimpse(df)
df$class <- as.factor(df$label)
corpus <- corpus(vectorSource(df$text))
```

Gambar 5 Custom Stopword

Hasil stopword :

```

cStopwordID      chr [1:758] "ada" "adalah" "adanya" "adapun" "agak" "agakny" "agar" ...
custom_stopwords chr [1:174] "i" "me" "my" "myself" "we" "our" "ours" "ourselves" "you" ...
custom_stopwords chr [1:187] "i" "me" "my" "myself" "we" "our" "ours" "ourselves" "you" ...
fit1.perf        'table' int [1:2, 1:2] 0 0 4 6
fit1.pred        Factor w/ 2 levels "terima", "tolak": 2 2 2 2 2 2 2 2 2
fivefreq         chr [1:148] "agung" "alasan" "alasanalasan" "berdasarkan" "berpendapat" ...
freq             Named num [1:442] 112 65 63 62 58 45 44 41 40 40 ...

```

Gambar 6 Hasil Custom Stopword

Berikutnya adalah menjalankan tahap *text processing* script yang digunakan gambar 7.

```

corpus.clean <- corpus %>%
  tm_map(content_transformer(tolower)) %>%
  tm_map(removePunctuation) %>%
  tm_map(removeNumbers) %>%
  tm_map(removewords, custom_stopwords) %>%
  tm_map(stripwhitespace)

```

Gambar 7. Script Text Processing

## 6. Document Term Matrix (DTM)

*Document Term Matrix* (DTM) adalah tahapan mempresentasikan kata-kata (*term*) menjadi matriks. Baris DTM sesuai dengan dokumen dalam dataset, kolom-kolom yang cocok dengan *term*, dan itu adalah unsur-unsur frekuensi *term*. Untuk membangun DTM menggunakan fungsi built-in dari *packages* 'tm'. Penyusunan *script* dapat dilihat pada gambar 8.

```

dtm <- DocumentTermMatrix(corpus.clean)
# Frequency
freq <- sort(colSums(as.matrix(dtm)), decreasing=TRUE)
wf <- data.frame(word=names(freq), freq=freq)

```

Gambar 8. Script Document Term Matrix

Gambar 9 adalah contoh dari beberapa *term* yang telah dibuat matriks. Data yang diambil berjumlah 3 dokumen dengan 10 *term*.

```

> #####
> tdm <- TermDocumentMatrix(corpus.clean)
> m <- as.matrix(tdm)
> v <- sort(rowSums(m), decreasing=TRUE)
> d <- data.frame(word = names(v), freq=v)
> head(d, 10)

```

word	freq
kasasi	112
pemohon	65
pengadilan	63
peninjauan	62
permohonan	58
pertimbangan	45
penggugat	44
merek	41
diajukan	40
hukum	40

Gambar 9. Contoh Document Term Matrix

## 7. Mempartisi Data

Data putusan kasasi pengadilan akan dibagi menjadi data latih dan data uji. Dari 14 data putusan kasasi pengadilan, data latih yang dipakai berjumlah 10 data putusan kasasi pengadilan yang dapat dilihat pada gambar 10, data latih berjumlah 10 data putusan kasasi pengadilan dapat dilihat pada gambar 9. Data latih terdiri dari 5 data *putusan kasasi Pengadilan* dengan keputusan ditolak dan 5 data *putusan kasasi Pengadilan* dengan keputusan kabul. Data latih menghasilkan 2.310 *term* yang dapat dilihat pada gambar 10. Setelah dilakukan *preprocessing*, *term* pada data latih berkurang menjadi 1.831 *term*.

```

> df.train <- df[1:40,]
> df.test <- df[41:50,]
> dtm.train <- dtm[1:40,]
> dtm.test <- dtm[41:50,]
> inspect(dtm.train)
<<DocumentTermMatrix (documents: 40, terms: 442)>>
Non-/sparse entries: 1268/16412
Sparsity : 93%
Maximal term length: 40
weighting : term frequency (tf)
Sample :
  Terms
docs diajukan hukum kasasi merek pemohon pengadilan penggugat peninjauan
  2      0      1      4      1      1      2      0      0
 21      2      1      2      3      2      2      3      0
 24      1      1      0      3      2      0      2      6
 25      1      2      2      3      1      2      1      0
 29      1      2      5      0      1      4      0      0
 3      1      0      6      1      3      0      4      0
 38      1      1      2      0      1      2      3      0
 4      1      4      4      0      1      4      4      0
 5      0      0      6      2      2      2      1      0
 6      2      1      0      1      2      0      1      8

```

Gambar 10. Term Data Latih Hasil Preprocessing

## 8 Klasifikasi Naïve Bayes

Untuk melatih model data menggunakan fungsi *naive bayes* dari packages 'e1071'. Karena *naive bayes* mengevaluasi produk probabilitas, maka memerlukan beberapa cara untuk menetapkan probabilitas non-nol ke kata-kata yang tidak terjadi dalam sampel. Gambar 11 merupakan *script* untuk model Naïve Bayes.

```

> corpus.clean.train <- corpus.clean[1:40]
> corpus.clean.test <- corpus.clean[41:50]
> fivefreq <- findFreqTerms(dtm.train, 2)
> length((fivefreq))
[1] 148
> dtm.train.nb <- DocumentTermMatrix(corpus.clean.train, control=list(dictionary = fivefreq))
> dim(dtm.train.nb)
[1] 40 148
> dtm.train.nb
<<DocumentTermMatrix (documents: 40, terms: 148)>>
Non-/sparse entries: 1037/4883
Sparsity : 82%
Maximal term length: 18
weighting : term frequency (tf)
> dtm.test.nb <- DocumentTermMatrix(corpus.clean.test, control=list(dictionary = fivefreq))
> dim(dtm.test.nb)
[1] 10 148
> # Function to convert the word frequencies to yes (presence) and no (absence) labels
> convert_count <- function(x) {

```

Gambar 11. Model Naïve Bayes.

Hasil prediksi :

```

> # Use the NB classifier we built to make predictions on the test set.
> system.time( pred <- predict(classifier, newdata=testNB) )
  user system elapsed
  0.06  0.00  0.06
> # Create a truth table by tabulating the predicted class labels with the actual class labels
> table("Predictions"= pred, "Actual" = df.test$class)
      Actual
Predictions Terima Tolak
Terima      1      0
Tolak      1      8

```

Gambar 12. Hasil Prediksi Data Uji

Gambar 12 merupakan hasil prediksi data uji menggunakan algoritma Naïve Bayes. Dari pengujian 10 data putusan kasasi Pengadilan algoritma *Naive Bayes* memberikan hasil klasifikasi putusan kasasi pengadilan ke dalam putusan di tolak sebanyak 8 dokumen dan ke dalam putusan Kabul sebanyak 1 dokumen.

## 10. Confusion Matrix (Menghitung Akurasi)

Dari hasil prediksi klasifikasi kelas putusan kasasi Pengadilan data uji pada tahap ini akan dihitung akurasi dari algoritma *Naive Bayes* dalam melakukan klasifikasi. Data hasil klasifikasi algoritma *Naive Bayes* akan dicocokkan dengan kelas putusan kasasi pengadilan yang telah diketahu sebelumnya.

```

> # Prepare the confusion matrix
> conf.mat <- confusionMatrix(pred, df.test$class)
> conf.mat
Confusion Matrix and Statistics

          Reference
Prediction Terima Tolak
Terima      1      0
Tolak      1      8

```

Gambar 13. Hasil *Confusion Matrix*

Dari gambar 13 hasil pencocokan data sentimen klasifikasi algoritma *naive bayes* dengan sentimen sebenarnya menghasilkan :

1. *True* Terima 1 dokumen
2. *True* Tolak 8 dokumen
3. *False* Tolak 0 dokumen
4. *False* Terima 1 dokumen

```

Accuracy : 0.8
95% CI : (0.4439, 0.9748)
No Information Rate : 0.6
P-Value [Acc > NIR] : 0.1673

kappa : 0.5455

McNemar's Test P-Value : 0.4795

Sensitivity : 0.50
Specificity : 1.00
Pos Pred Value : 1.00
Neg Pred Value : 0.75
Prevalence : 0.40
Detection Rate : 0.20
Detection Prevalence : 0.20
Balanced Accuracy : 0.75

'Positive' Class : Terima

```

Gambar 14. Akurasi Klasifikasi Algoritma *Naive Bayes*

Pada Gambar 14 akurasi dari algoritma *Naive Bayes* dalam melakukan klasifikasi *putusan kasasi pengadilan* dalam penelitian ini sebesar  $0.8 = 80\%$ .

Tampilan Plot pada Data Putusan Kasasi Pengadilan :

Untuk visualisasi dari klasifikasi Data Putusan Kasasi Pengadilan dapat menggunakan wordcloud dan histogram. Adapun code untuk wordcloud adalah sebagai berikut :

```

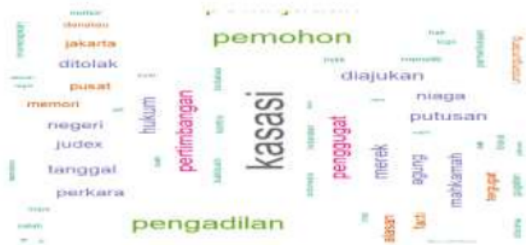
#####
tdm <- TermDocumentMatrix(corpus.clean)

m <- as.matrix(tdm)
v <- sort(rowSums(m), decreasing=TRUE)
d <- data.frame(word = names(v), freq=v)
head(d, 10)
wordcloud(words = d$word, freq = d$freq, min.freq = 2,
max.words=100, random.order=FALSE, rot.per=0.35,
colors=brewer.pal(8, "dark2"))

```

Gambar 15 Script untuk wordcloud Putusan Kasasi Pengadilan

Adapun hasilnya adalah sebagai berikut :



Gambar 16 Hasil Wordcloud Putusan Kasasi Pengadilan

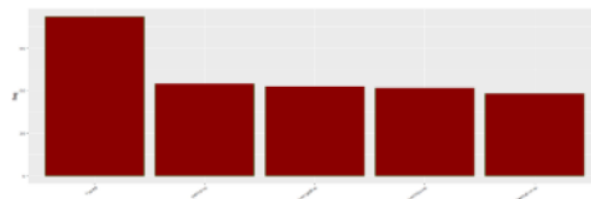
Pada gambar 16 menunjukkan bahwa kata-kata kasasi paling banyak muncul, hal ini ditunjukkan kata kasasi mempunyai ukuran font yang paling besar, dilanjutkan dengan kata pemohon dan pengadilan. Untuk visualisasi dengan histogram dapat menggunakan script sebagai berikut :

```
# Frequency
freq <- sort(colSums(as.matrix(dtm)), decreasing=TRUE)
wf <- data.frame(word=names(freq), freq=freq)
#Plot Histogram
subset(wf, freq>50) %>%
  ggplot(aes(word, freq)) +
  geom_bar(stat="identity", fill="darkred", colour="darkgreen") +
  theme(axis.text.x=element_text(angle=45, hjust=1))

#####
tdm <- TermDocumentMatrix(corpus_clean)
```

Gambar 17 Script untuk membuat histogram Putusan kasasi Pengadilan

Hasilnya adalah sebagai berikut :



Gambar 18 Histogram Klasifikasi Putusan Pengadilan

Gambar 18 menunjukkan bahwa 5 kata yang paling banyak muncul pada analisa text Putusan Kasasi Pengadilan yaitu kasasi, pemohon, pengadilan, permohonan dan peninjauan.

#### 4. KESIMPULAN

Berdasarkan hasil klasifikasi pada data dokumen Putusan Pengadilan pada Pengadilan Tentang Merk menggunakan metode Naïve Bayes, dapat diambil kesimpulan sebagai berikut :

1. Metode Naïve Bayes dalam melakukan klasifikasi putusan pengadilan kasasi Pengadilan ditolak dan diterima dengan 10 data latih dan 4 data uji mendapat hasil akurasi sebesar 80%.
2. Dengan hasil akurasi yang cukup tinggi yaitu 80% maka metode Naïve Bayes dapat digunakan untuk melakukan klasifikasi putusan pengadilan Pengadilan dengan sentimen diterima dan ditolak secara otomatis.

#### DAFTAR PUSTAKA

- [1] Afshoh, F., Pamungkas, E. W., Kom, S., & Kom, M. (2017) Analisa Sentimen Menggunakan Naïve Bayes Untuk Melihat Persepsi Masyarakat Terhadap Kenaikan Harga Jual Rokok Pada Media Sosial Twitter, *Disertasi*, Universitas Muhammadiyah, Surakarta.
- [2] Alzola, C. & Harrell, F.E. (2006) An introduction to S and the Hmisc and design libraries.. Website: <http://biostat.mc.vanderbilt.edu/RS>, diakses 12 April 2018.
- [3] Anjani, D. (2015) Analisis Kemiripan Dokumen Tugas Akhir Untuk Penilaian Originalitas, *Disertasi*, Doctoral, Universitas Widyatama, Bandung.
- [4] Faradhillah, N. Y., Kusumawardani, R. P., & Hafidz, I. (2016) Eksperimen Sistem Klasifikasi Analisa Sentimen Twitter pada Akun Resmi Pemerintah Kota Surabaya Berbasis Pembelajaran Mesin. *Seminar Nasional Sistem Informasi Indonesia*, 1(3), pp.16-24.
- [5] Hanggara, S., Akhriza, T. M., & Husni, M. (2017) Aplikasi Web Untuk Analisis Sentimen Pada Opini Produk Dengan Metode Naive Bayes Classifier, *Prosiding Seminar Nasional Inovasi dan Aplikasi Teknologi di Industri*, 3(1), pp.33-1.
- [6] Hidayat, A. (2016) Algoritma Naives Bayes, Website: <https://arfianhidayat.com/algoritma-naive-bayes>, diakses 14 April 2018.



- [7] Lorosae, T. A., Prakoso, B. D., Saifudin, S., & Kusrin, K. (2018) Analisis Sentimen Berdasarkan Opini Masyarakat pada Twitter Menggunakan Naive Bayes, *Seminar Nasional Teknologi Informasi & Multimedia Online*, 6(1), pp.1-10.
- [8] Marjuki, A. (2017) Analisis Klastering Lirik Lagu Indonesia, *Skripsi*, Program Studi Sistem Informasi FTI Unisbank, Semarang.
- [9] Maulana, A. (2016) Twitter Rahasiakan Jumlah Pengguna di Indonesia, Website: <https://www.cnnindonesia.com/teknologi/20160322085045-185-118939/twitter-rahasiakan-jumlah-pengguna-di-indonesia>, diakses 20 April 2018.
- [10] Prasetyo, Eko. (2012) *Data Mining Konsep dan Aplikasi Menggunakan Matlab*, Andi Yogyakarta, Yogyakarta.
- [11] Rozi, I. F., Pramono, S. H., & Dahlan, E. A. (2013). Implementasi Opinion Mining (Analisis Sentimen) untuk Ekstraksi Data Opini Publik pada Perguruan Tinggi, *Jurnal Electrical Power, Electronics, Communications, Controls, and Informatics Seminar*, 6(1), pp.37-43.
- [12] H. Riduan Syahrani, S.H., *Buku Materi Dasar Hukum Acara Perdata*, PT. Citra Aditya Bakti Bandung, Cet. V, 2009

# 10\_KLASIFIKASI DOKUMEN DENGAN METODE NAIVE BAYES

---

## ORIGINALITY REPORT

---

**21** %

SIMILARITY INDEX

**19** %

INTERNET SOURCES

**2** %

PUBLICATIONS

**4** %

STUDENT PAPERS

---

## MATCH ALL SOURCES (ONLY SELECTED SOURCE PRINTED)

---

4%

★ [repositori.unsil.ac.id](http://repositori.unsil.ac.id)

Internet Source

---

Exclude quotes  On

Exclude matches  < 2%

Exclude bibliography  On