

**PERBANDINGAN ALGORITMA *SUPPORT VECTOR MACHINE* (SVM) DAN
MULTINOMIAL NAIVE BAYES (MNB) DALAM KLASIFIKASI
ABSTRAK TUGAS AKHIR
(STUDI KASUS: FAKULTAS TEKNIK UNIVERSITAS MUHAMMADIYAH
JEMBER)**

Feni Shofiya¹, Deni Arifianto, M.Kom.², Habibatul Azizah Al Faruq, M.Pd.³

Program Studi Teknik Informatika, Fakultas Teknik, Universitas Muhammadiyah Jember

Fenishofiya17@gmail.com¹, deniarifianto@unmuhjember.ac.id², habibatulazizah@unmuhjember.ac.id³

ABSTRAK

Abstrak adalah representasi yang ringkas tetapi akurat dari isi suatu dokumen. Dalam penentuan kategori abstrak Tugas Akhir masih sulit dalam mengelompokkannya. Jika dikelompokkan dengan kemampuan manusia masih mengalami kesulitan karena isi dari abstrak Tugas Akhir yang sangat banyak. Hal tersebut akan membutuhkan waktu yang lama. Pengelolaan kata yang terdapat pada abstrak memiliki suatu kemiripan antara abstrak satu dengan yang lainnya sehingga terjadi ambiguitas pengelompokan beberapa kategori. Pada penelitian ini dilakukan klasifikasi dokumen terhadap abstrak Tugas Akhir mahasiswa Fakultas Teknik Universitas Muhammadiyah Jember. Data yang digunakan dalam penelitian ini adalah abstrak Tugas Akhir pada Teknik Elektro, Teknik Sipil, Teknik Informatika, Teknik Mesin, dan Manajemen Informatika. Metode klasifikasi pada penelitian ini adalah membandingkan algoritma *Support Vector Machine* dan *Multinomial Naive Bayes*. Hasil pengujian dari penelitian ini adalah pada algoritma SVM didapatkan hasil akurasi sebesar 82%, presisi 84%, dan *recall* 82%. Pada algoritma MNB didapatkan hasil akurasi sebesar 62%, presisi 68%, dan *recall* 62%.

Kata Kunci : Klasifikasi dokumen, Abstrak, *Support Vector Machine*, *Multinomial Naive Bayes*

PENDAHULUAN

Latar Belakang

Abstrak adalah representasi yang ringkas tetapi akurat dari isi suatu dokumen. Dalam penentuan kategori abstrak tugas akhir masih sulit dalam mengelompokkannya. Jika dikelompokkan dengan kemampuan manusia masih mengalami kesulitan karena isi dari abstrak tugas akhir yang sangat banyak.

Hal tersebut akan membutuhkan waktu yang lama. Pengelolaan kata yang terdapat pada abstrak memiliki suatu kemiripan antara abstrak satu dengan yang lainnya sehingga terjadi ambiguitas pengelompokan beberapa kategori.

Universitas Muhammadiyah Jember memiliki sepuluh fakultas, salah satunya adalah Fakultas Teknik yang di dalamnya terdapat lima jurusan yaitu Teknik Informatika, Teknik Sipil, Teknik Mesin, Teknik Elektro dan Manajemen Informatika. Namun tugas akhir yang ada di Fakultas Teknik Universitas Muhammadiyah Jember belum ada yang mengklasifikasikan berdasarkan program studi.

Salah satu metode yang digunakan dalam klasifikasi dokumen adalah metode *text mining* merupakan pengembangan dari metode *data mining* yang diterapkan untuk mengatasi masalah tersebut. Untuk melakukan proses klasifikasi teks, ada beberapa algoritma yang dapat digunakan salah satu diantaranya *Support Vector Machine* (SVM).

Rumusan Masalah

Berdasarkan latar belakang yang telah dikemukakan di atas, maka permasalahan yang akan dibahas dalam penelitian ini adalah sebagai berikut: Berapa tingkat akurasi, presisi, dan *recall*

perbandingan algoritma *Support Vector Machine* (SVM) dan *Multinomial Naive Bayes* (MNB) dalam klasifikasi abstrak Tugas Akhir Fakultas Teknik Universitas Muhammadiyah Jember?

Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah membandingkan tingkat akurasi, presisi, dan *recall* perbandingan algoritma *Support Vector Machine* (SVM) dan *Multinomial Naive Bayes* (MNB) dalam klasifikasi abstrak Tugas Akhir Fakultas Teknik Universitas Muhammadiyah Jember

Manfaat Penelitian

Manfaat dari penelitian ini yang dilakukan sebagai berikut: (1) Memberikan kontribusi secara keilmuan berupa perbandingan algoritma *Support Vector Machine* dan *Multinomial Naive Bayes* dalam klasifikasi abstrak tugas akhir. (2) Dapat digunakan sebagai referensi untuk penelitian-penelitian selanjutnya di bidang klasifikasi dengan metode *text mining*.

METODE PENELITIAN

Pengumpulan Data

Pengambilan data pada Repository Universitas Muhammadiyah Jember dilakukan dengan cara manual atau mengambil satu per satu file PDF abstrak tiap judul skripsi.

Pre-processing

Pre-processing dilakukan untuk mempersiapkan data abstrak sehingga bersih dari *noise* dan sesuai dengan ketentuan klasifikasi teks. Menurut Triawati dkk. (2009), beberapa proses yang dilakukan antara lain:

1. *Case folding*, yaitu proses mengonversi keseluruhan tulisan dalam data menjadi huruf kecil.
2. *Tokenizing*, yaitu proses pemotongan tulisan berupa huruf untuk dipisahkan menjadi kata per kata.
3. *Filtering*, yaitu proses menghapus kata-kata yang kurang penting seperti kata “yang”, “dan”, “di”, “dengan” dari hasil pemotongan tulisan.
4. *Stemming*, yaitu proses mencari kata dasar dari tiap kata pembentuknya.

Pembobotan

Pembobotan TF-IDF dilakukan dengan cara pemberian bobot hubungan suatu kata atau fitur (t) sebanyak m terhadap data (d) sebanyak n , serta w merupakan hasil pembobotan TF-IDF.

Beberapa tahapan yang dilakukan pada pembobotan menggunakan TF-IDF antara lain:

1. *Term frequency* atau TF, merupakan perhitungan frekuensi suatu term atau kata yang terdapat pada setiap data berbentuk dokumen.
2. *Inverse document frequency* atau IDF, merupakan hubungan ketersediaan suatu term dalam seluruh dokumen. Persamaan yang digunakan pada perhitungan IDF ditunjukkan pada persamaan 1. (Havrlant dan Kreinovich, 2014)

$$idf(t, d) = \log \frac{N}{df(t)} \quad (1)$$

3. *Term Weighting* TF-IDF, merupakan penggabungan dari rumus TF dengan IDF dengan mengalikan kedua rumus tersebut menjadi sebuah nilai pembobotan. Perhitungan tersebut ditunjukkan pada persamaan 2.

$$w(t, d) = tf(t, d) \times idf(t, d) \quad (2)$$

Support Vector Machine

Metode klasifikasi yang digunakan pada penelitian ini menggunakan metode *Support Vector Machine* (SVM) dengan konsep non linear. *Support Vector Machine* adalah mencari fungsi pemisah (*hyperplane*) terbaik atau paling optimum yang berfungsi sebagai pemisah dua kelas pada ruang vektor *input*. Untuk mengklasifikasikan data pada permasalahan non linear, formula SVM harus dimodifikasi, yaitu dengan

Setiap data dinotasikan sebagai \vec{x}_i , untuk $i = 1, 2, \dots, l$ (banyak data), sedangkan untuk setiap kelas pada data dinotasikan sebagai $y_i \in \{-1, +1\}$. Semua data dijabarkan sebagai *pattern* yang merupakan anggota dari dua kelas, yaitu kelas 1 dan kelas 2. *Pattern* yang letaknya paling dekat dengan *hyperplane* disebut *support vector*. Kemudian jarak antara *support vector* tiap kelas dengan *hyperplane* disebut sebagai *margin*.

Hyperplane klasifikasi linear didefinisikan pada persamaan berikut:

$$\langle \vec{w}, \vec{x}_i \rangle + b = 0 \quad (3)$$

\vec{w} merupakan normal bidang dan b merupakan posisi bidang relatif terhadap pusat koordinat. Data \vec{x}_i yang tergolong ke dalam kelas 1 merupakan data yang memenuhi pertidaksamaan berikut:

$$\langle \vec{w}, \vec{x}_i \rangle + b \leq -1 \quad (4)$$

Sedangkan data \vec{x}_i yang tergolong ke dalam kelas 2 merupakan data yang memenuhi pertidaksamaan berikut:

$$\langle \vec{w}, \vec{x}_i \rangle + b \geq 1 \quad (5)$$

Margin terbesar dapat dicari dengan cara memaksimalkan jarak antar bidang pembatas kedua kelas dan titik

terdekatnya, yaitu $2/|w|$. Hal ini dirumuskan sebagai permasalahan *quadratic programming* (QP) *problem* yaitu mencari titik minimal persamaan (6) dengan memperhatikan persamaan (7) berikut:

$$\min \tau(w) = \frac{1}{2} \|w\|^2 \quad (6)$$

$$y_i(w * x_i + b) - 1 \geq 0, (i = 1, \dots, n) \quad (7)$$

Permasalahan ini dapat dipecahkan dengan berbagai teknik komputasi. Lebih mudah diselesaikan dengan mengubah persamaan (6) ke dalam fungsi *Lagrangian* pada persamaan (8), dan menyederhanakannya menjadi persamaan (9) berikut:

$$L(w, b, a) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^n a_i (y_i((w^T x_i + b) - 1)) \quad (8)$$

$$l(w, b, a) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^n a_i y_i (w^T x_i + b) + \sum_{i=1}^n a_i \quad (9)$$

Dimana a_i adalah *lagrange multiplier* yang bernilai nol atau positif ($a_i \geq 0$). Nilai optimal dari persamaan (9) dapat dihitung dengan meminimalkan L terhadap w , b dan a . Dapat dilihat pada persamaan (10) sampai (12) berikut:

$$\frac{\partial L}{\partial w} = w - \sum_{i=1}^n a_i y_i x_i = 0 \quad (10)$$

$$\frac{\partial L}{\partial b} = \sum_{i=1}^n a_i y_i = 0 \quad (11)$$

$$\frac{\partial L}{\partial a} = \sum_{i=1}^n a_i y_i (w^T x_i + b) - \sum_{i=1}^n a_i = 0 \quad (12)$$

Maka masalah *Lagrange* untuk klasifikasi dapat dinyatakan pada persamaan (13) berikut:

$$\begin{aligned} \text{Min } L(w, b, a) = & \\ \frac{1}{2} \|w\|^T \sum_{i=1}^n a_i y_i (w^T x_i + b) - & \\ \sum_{i=1}^n a_i = 0 & \end{aligned} \quad (13)$$

Sedangkan dengan memaksimalkan L terhadap a_i , persamaannya menjadi persamaan (14) berikut:

$$\begin{aligned} \text{Max } \sum_{i=1}^n a_i - & \\ \frac{1}{2} \sum_{i=1, j=1}^n a_i a_j y_i y_j x_i x_j & \end{aligned} \quad (14)$$

Untuk mendapatkan nilai a_i , langkah pertama adalah mengubah setiap kalimat menjadi nilai vektor (*support vector*) = $\begin{pmatrix} x \\ y \end{pmatrix}$. Kemudian nilai vektor dari setiap kalimat dimasukkan ke persamaan (15) berikut:

$$\begin{aligned} S_i = \phi \begin{pmatrix} x \\ y \end{pmatrix} = & \\ \begin{cases} \sqrt{x^2 + y^2} > 2 \rightarrow \begin{pmatrix} 4 - y + (x - y) \\ 4 - x + (x - y) \end{pmatrix} \\ \sqrt{x^2 + y^2} \leq 2 \rightarrow \begin{pmatrix} x \\ y \end{pmatrix} \end{cases} & \end{aligned} \quad (15)$$

Nilai x didapatkan dari persamaan (16) kernel linear untuk x berikut:

$$\sum_{i=1, j=1}^n x_i x_j^T, (i, j = 1, \dots, n) \quad (16)$$

Nilai y didapatkan dari persamaan (17) kernel linear untuk y berikut:

$$\sum_{i=1, j=1}^n y_i y_j^T, (i, j = 1, \dots, n) \quad (17)$$

Untuk mendapatkan jarak tegak lurus yang optimal dengan mempertimbangkan vektor positif, maka hasil perhitungan dari substitusi nilai x dan nilai y ke persamaan (15) diberi nilai bias = 1. Kemudian cari parameter a_i , dengan terlebih dahulu mencari nilai fungsi setiap kalimat menggunakan persamaan (18), lalu mencari nilai a_i pada persamaan linear menggunakan persamaan (19) dengan memperhatikan $i, j=1, \dots, n$ berikut:

$$\sum_{i=1, j=1}^n a_i S_i^T S_j \quad (18)$$

$$\sum_{i=1, j=1}^n a_i S_i^T S_j = y_i \quad (19)$$

Setelah parameter a_i didapatkan, kemudian masukkan ke persamaan (20) sebagai berikut:

$$W = \sum_{i=1}^n a_i S_i \quad (20)$$

Selanjutnya, digunakan persamaan (21) untuk mendapatkan nilai w dan b :

$$y = wx + b \quad (21)$$

Pada awalnya SVM dikembangkan untuk persoalan klasifikasi dua kelas, kemudian dikembangkan kembali untuk klasifikasi multikelas. Dalam klasifikasi kasus multikelas, *hyperplane* yang terbentuk adalah lebih dari satu. Salah satu metode pendekatan yang digunakan adalah *One Against All* (OAA). Metode OAA untuk kasus klasifikasi k -kelas, menemukan k *hyperplane* dimana k adalah banyak kelas dan ρ adalah *hyperplane*. Dalam metode ini $\rho(\ell)$ diujikan dengan semua data dari kelas ℓ dengan label +1, dan semua data dari kelas lain dengan label -1. Kemudian dicari *hyperplane* dengan algoritma SVM dua kelas. Maka akan didapat *hyperplane* untuk masing-masing kelas. Kemudian kelas dari suatu data baru x ditentukan berdasarkan nilai terbesar dari *hyperplane*.

Multinomial Naive Bayes

Multinomial Naive Bayes adalah *conditional probability* yang dilakukan tanpa memperhitungkan urutan kata dan informasi yang ada dalam kalimat atau dokumen secara umum. Dan pada algoritma ini juga memperhitungkan jumlah kata yang muncul dalam dokumen (Destuardi dan Surya, 2009). *Algoritme conditional probability Multinomial Naive Bayes* dalam melakukan perhitungan peluang sebuah kata i masuk ke dalam kategori j dapat dilakukan dengan menggunakan persamaan berikut:

$$P(w_k | v_j) = \frac{W_{ct} + 1}{(\sum W' \in V W'_{ct}) + B'} \quad (22)$$

Evaluasi Dan Validasi

Proses evaluasi menggunakan *K-Fold Cross Validation* membagi atau memartisi *dataset* menjadi beberapa bagian. Penentuan nilai *fold* K yang digunakan pada penelitian (2011) bernilai 2, 4, 5 dan 10. Proses validasi dilakukan terhadap kumpulan *dataset* abstrak Tugas Akhir mahasiswa Fakultas Teknik Universitas Muhammadiyah Jember untuk mendapatkan performansi akurasi yang optimal. *Dataset* yang digunakan sebanyak 50 data abstrak Tugas Akhir.

HASIL DAN PEMBAHASAN

Data

Data yang dikumpulkan berasal dari abstrak Tugas Akhir Fakultas Teknik Universitas Muhammadiyah Jember yang ada *repository* Universitas Muhammadiyah Jember. Teknik pengumpulan data menggunakan studi literatur dengan cara mengambil abstrak Tugas Akhir mahasiswa Fakultas Teknik saja. Dari hasil studi literatur didapatkan data abstrak Tugas Akhir mahasiswa Fakultas Teknik berjumlah 100 abstrak Tugas Akhir, dengan masing-masing kelas atau prodi memiliki abstrak berjumlah 20 abstrak.

Pre-processing Data

Pre-processing data abstrak dilakukan *case folding*, *tokenizing*, *filtering*, dan *stemming* pada setiap abstrak Tugas Akhir tersebut.

Menggunakan modul Sastrawi untuk mengubah kata – kata menjadi kebentuk dasarnya. Setelah dilakukan *pre-processing* data, dilakukan pelabelan abstrak yang awalnya berupa Teknik Elektro, Teknik Sipil, Teknik Informatika, Teknik Mesin, dan Manajemen Informatika, diubah menjadi nilai numerik,

yaitu nilai “1” untuk Teknik Elektro, nilai “2” untuk Teknik Sipil, nilai “3” untuk Teknik Informatika, nilai “4” untuk Teknik Mesin, dan nilai “5” untuk Manajemen Informatika.

Implementasi Pembobotan

Pembobotan pada data abstrak Tugas Akhir dilakukan menggunakan algoritma TF-IDF. Pembobotan ini dilakukan untuk mencari nilai dari tiap-tiap kata yang ada pada sekumpulan data yang akan dibentuk menjadi sebuah vektor pada proses klasifikasi. Dengan teknik pembobotan ini diimplementasikan ke dalam *Jupyter Notebook*, jumlah fitur atau kata pada semua abstrak Tugas Akhir didapatkan 2659 kata yang digunakan.

Tabel 4.1 Hasil Pembobotan pada Abstrak

	absensi	mesin	...	zat	Zone
Data 1	5,2	0	...	3,7	0
Data 2	5,2	0	...	0	0
...
Data 99	0	0	...	3,7	0
Data 100	0	0	...	0	0

Hasil Pengujian

Pengukuran dilakukan dengan mengambil model pada skenario masing-masing algoritma yang kemudian model tersebut akan diujikan sebagai data latih untuk data baru yang akan di validasi. Untuk algoritma *Support Vector Machine (SVM)* menggunakan model pada *fold k 4* skenario 3, dan untuk algoritma *Multinomial Naive Bayes (MNB)* menggunakan model pada *fold k 10* skenario 1.

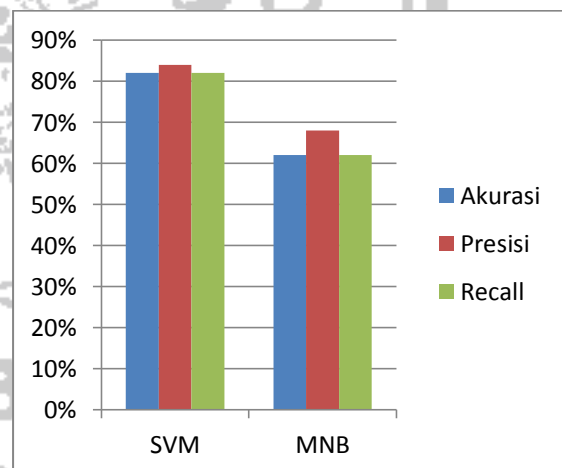
Untuk data uji diambil dari data baru sebanyak 50 data abstrak Tugas Akhir yang terdiri dari 10 data abstrak Tugas Akhir Teknik Sipil, 10 abstrak Tugas Akhir

Teknik Elektro, 10 data abstrak Tugas Akhir Teknik Informatika, 10 data abstrak Tugas Akhir Teknik Mesin, dan 10 abstrak Tugas Akhir Manajemen Informatika. Berikut hasil klasifikasi data validasi yang ditampilkan pada Tabel 4.2.

Tabel 4.2 Hasil klasifikasi data validasi

	AKURASI	PRESISI	RECALL
SVM	82%	84%	82%
MNB	62%	68%	62%

Berdasarkan hasil klasifikasi dengan menggunakan data validasi untuk algoritma *Support Vector Machine (SVM)* didapatkan hasil akurasi sebesar 82%, presisi 84%, dan *recall* 82%. Sedangkan untuk algoritma *Multinomial Naive Bayes (MNB)* didapatkan hasil akurasi sebesar 62%, presisi 68%, dan *recall* sebesar 62%.



KESIMPULAN DAN SARAN

Kesimpulan

Berdasarkan penelitian yang telah dilakukan, dapat diambil kesimpulan sebagai berikut:

1. Hasil akurasi paling tinggi dalam klasifikasi abstrak Tugas Akhir menggunakan algoritma *Support Vector Machine* didapatkan hasil sebesar 82%,

- untuk hasil akurasi dalam klasifikasi abstrak Tugas Akhir menggunakan algoritma *Multinomial Naive Bayes* didapatkan hasil sebesar 62%.
2. Hasil presisi paling tinggi dalam klasifikasi abstrak Tugas Akhir menggunakan algoritma *Support Vector Machine* didapatkan hasil sebesar 84%, untuk hasil presisi dalam klasifikasi abstrak Tugas Akhir menggunakan algoritma *Multinomial Naive Bayes* didapatkan hasil sebesar 68%.
 3. Hasil *recall* paling tinggi dalam klasifikasi abstrak Tugas Akhir menggunakan algoritma *Support Vector Machine* didapatkan hasil sebesar 82%, untuk hasil *recall* dalam klasifikasi abstrak Tugas Akhir menggunakan algoritma *Multinomial Naive Bayes* didapatkan hasil sebesar 62%.
 4. Secara keseluruhan kinerja algoritma *Support Vector Machine* lebih baik dibandingkan dengan *Multinomial Naive Bayes* dalam klasifikasi abstrak Tugas Akhir.

Saran

Berdasarkan penelitian yang telah dilakukan, saran yang dapat dikembangkan untuk penelitian selanjutnya adalah sebagai berikut:

1. Perlu dikembangkan menggunakan jumlah data abstrak yang lebih banyak untuk dapat lebih mendapatkan banyak kata-kata dan keakuratan dalam pelabelan data sehingga hasil yang didapatkan lebih maksimal.
2. Diharapkan untuk peneliti selanjutnya bisa menerapkannya dalam bentuk sistem klasifikasi abstrak Tugas Akhir dengan penelitian ini sebagai acuan.

DAFTAR PUSTAKA

- Asiyah, S dan Fithriasari, K. 2016. "Klasifikasi Berita Online Menggunakan Metode Support Vector Machine dan K- Nearest Neighbor". *JURNAL SAINS DAN SENI ITS*, Vol. 5 No. 2. Institut Teknologi Sepuluh Nopember.
- Avila, D. dkk . 2020. *About Us: Some information about the Jupyter Project and Community* [online]. Tersedia di <https://jupyter.org/about>. Diakses Februari 2020.
- Chang, W. 2014. "Metodologi Penulisan Ilmiah" dalam Sabanise, Y. F. 2017. "Klasifikasi Abstrak Tugas Akhir Mahasiswa Politeknik Harapan Bersama Tegal". ISBN: 978-602-74355-1-3. Jurusan Teknik Komputer, Politeknik Harapan Bersama Tegal.
- Destuardi dan Surya, S. 2009. "Klasifikasi Emosi Untuk Teks Bahasa Indonesia Menggunakan Metode Naive Bayes". Surabaya: Teknik Elektro, Institut Teknologi Sepuluh Nopember.
- Feldman, R dan Sanger, J. 2007. *Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data*. New York: Cambridge University Press.
- Harvlant, L. dan Kreinovich, V. 2014. *A simple probabilistic explanation of term frequency-inverse document frequency (tf-idf) heuristic (and variations motivated by this explanation)*. University of Texas at El Paso.
- Hidayatullah, A.F dan Ma'arif, M.R. 2016. "Penerapan Text Mining dalam Klasifikasi Judul Skripsi," p.4.

- Ipmawati, Kusriani, dan Luthfi, E.T. 2017. "Komparasi Teknik Klasifikasi Teks Mining Pada Analisis Sentimen". STMIK AMIKOM Yogyakarta.
- Kamus Besar Bahasa Indonesia (KBBI) [online]. Tersedia di <https://kbbi.web.id/abstrak>. Diakses 22 Nopember 2019.
- Kaplan, R.M. 1995. "A Methode for Tokenizing Text. Palo Alto Research Center (Festchrift in The Honor of Prof. Kimmo Koskenniemi's 60 th Anniversary)" dalam Sulisty, W. 2008. "Klasifikasi Dokumen Berbahasa Inggris Berdasarkan Weighted-Term". *Jurnal Teknologi Informasi-Aiti*, Vol.5 No.1, Februari 2008: 1-100. Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana.
- Kunchady, M. 2006. *Text Mining Application Programming*. Thomson Learning Inc. ISBN 1-58450-460-9.
- Lidya, S.K., Sitompul, O.S. dan Efendi, S. 2015. "Sentiment Analysis Pada Teks Bahasa Indonesia Menggunakan Support Vector Machine (SVM) dan K-Nearest Neighbor(K-NN)". *SENTIKA* 2015 ISSN: 2089-9815. Universitas Sumatera Utara.
- Machine Learning: Support Vector Machines [online]. Tersedia di <https://www.megabagus.id/machine-learning-support-vector-machines/>
- Mauludiya, R.A. 2015. "Simulasi dan Analisis Klasifikasi Genre Musik Berbasis FFT dan Support Vector Machine". Universitas Telkom, S1 Teknik Telekomunikasi, Bandung.
- Muhammad Rangga A.N. dan Mardhiya, Hayati. 2019. "Perbandingan Akurasi dan Waktu Proses Algoritma K-NN dan SVM dalam Analisis Sentimen *Twitter*". *Jurnal Informatika*, Vol. 6 No. 2. Universitas Amikom Yogyakarta.
- Mulyana, I., Ramadana, S., dan Herfina. 2012. "Penerapan Term Frequency-Inverse Document Frequency pada Sistem Peringkasan Teks Otomatis Dokumen Tunggal Berbahasa Indonesia".
- Ningrum, E.L.W dan Widodo, A.P. 2018. "Implementasi Metode *Multinomial Naive Bayes Classifier* Untuk Analisis Sentimen". *JFMA* Vol. 1 No. 2. Universitas Diponegoro.
- Nugroho, A.S. 2008. Pengantar Support Vector Machine, [online]. Tersedia di <http://asnugroho.net/papers/ikcsvm.pdf>. Diakses November 2019.
- Octaviani, P.A., Wilandari, Y., dan Ispriyanti, D. 2014. "Penerapan Metode Klasifikasi *Support Vector Machine (SVM)* Pada Data Akreditasi Sekolah Dasar (SD) Di Kabupaten Magelang". *Jurnal Gaussian* Vol. 3 No. 4. ISSN: 2339-2541. Universitas Diponegoro.
- Perpuspedia – Abstract : Universitas Diponegoro Digital Library [online]. Tersedia di <https://digilib.undip.ac.id/v2/2012/05/14/abstract/>. Diakses Januari 2020.
- Porter, M.F. 2001. *Snowball: A language for Stemming Algorithms Computer Laboratory*. England: Cambridge University Press.
- Rahman, A. Wiranto dan Doewes, A. 2017. "Online News Classification

- Using Multinomial Naive Bayes*". *ITSMART* Vol. 6 No.1. Universitas Sebelas Maret.
- Repository Universitas Muhammadiyah Jember [online]. Tersedia di <http://repository.unmuhjember.ac.id/view/divisions/information/>. Diakses Nopember 2019.
- Sabanise, Y.F. 2017. "Klasifikasi Abstrak Tugas Akhir Mahasiswa Politeknik Harapan Bersama Tegal". ISBN: 978-602-74355-1-3.
- Salekhah, C. 2016. "Implementasi Metode Multi Class Support Vector Machine Untuk Klasifikasi Emosi pada Lirik Lagu Bahasa Indonesia". Universitas Komputer Indonesia.
- Santosa, B. 2015. *Tutorial Support Vector Machine*. Institut Teknologi Sepuluh Nopember.
- Setiawan, P. 2019. Pengertian Dan Manfaat Kalimat Abstrak Lengkap [online]. Tersedia di <https://www.gurupendidikan.co.id/kalimat-abstrak/>. Diakses Desember 2019.
- Setiyono, A dan Pardede, H. 2019. "Klasifikasi SMS Spam Menggunakan Support Vector Machine". *Jurnal Pilar Nusa Mandiri* Vol. 15 No 2. Sekolah Tinggi Manajemen Informatika dan Komputer Nusa Mandiri.
- Sianturi, M.F. 2017. "Klasifikasi Dokumen Menggunakan Kombinasi Algoritma Principal Component Analysis dan SVM". Universitas Telkom.
- Suyanto. 2018. *Machine Learning: Tingkat Dasar dan Lanjutan*. Informatika Bandung, Bandung. ISBN 978-602-6232-78-6.
- Triawati, dkk. 2009. "Metode Pembobotan Statistical Concept Based untuk Klustering dan Kategorisasi Dokumen Berbahasa Indonesia". Institut Teknologi Telkom, Bandung.
- Weissberg, R and Buker, S. 1990. *Writing up Research: Experimental Research Report Writing for Students of English*. New Jersey: PrenticeHall, Inc. Dalam Yanti, Y. 2014. "Cara Penulisan Abstrak Di Dalam Karya Ilmiah". Fakultas Ilmu Budaya (FIB) Universitas Bung Hatta.
- Zakky. 2018. Pengertian Abstrak Menurut Para Ahli, KBBI dan Secara Umum [online]. Tersedia di <https://www.zonareferensi.com/pengertian-abstrak/>. Diakses Januari 2020.