TUGAS AKHIR – SS141501

KLASIFIKASI BERITA *ONLINE* MENGGUNAKAN METODE *SUPPORT VECTOR MACHINE* DAN *K-NEAREST NEIGHBOR*

SITI NUR ASIYAH
NRP 1314 105 016

Dosen Pembimbing
Dr. Kartika Fithriasari, M.Si

PROGRAM STUDI S1
JURUSAN STATISTIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA 2016

FINAL PROJECT – SS141501

ONLINE NEWS CLASSIFICATION USING SUPPORT VECTOR MACHINE AND K-NEAREST NEIGHBOR

SITI NUR ASIYAH
NRP 1314 105 016

Supervisor
Dr. Kartika Fithriasari, M.Si

UNDERGRADUATE PROGRAMME
DEPARTMENT OF STATISTICS
FACULTY OF MATHEMATICS AND NATURAL SCIENCES
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA 2016

LEMBAR PENGESAHAN

KLASIFIKASI BERITA ONLINE MENGGUNAKAN METODE *SUPPORT VECTOR MACHINE* DAN *K-NEAREST NEIGHBOR*

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Sains
pada
Program Studi S-1 Jurusan Statistika
Fakultas Matematika dan Ilmu Pengetahuan Alam
Institut Teknologi Sepuluh Nopember

Oleh :
SITI NUR ASIYAH
NRP. 1314 105 016

Disetujui oleh Pembimbing Tugas Akhir :

Dr. Kartika Fithriasari, M.Si (.....)
NIP. 19691212 199303 2 002

Mengetahui,
Ketua Jurusan Statistika FMIPA-ITS

Dr. Suhartono
NIP. 19710929 199512 1 001

SURABAYA, JULI 2016

V

Klasifikasi Berita *Online* Menggunakan Metode *Support Vector Machine* dan *K- Nearest Neighbor*

Nama	: Siti Nur Asiyah
NRP	: 1314 105 016
Jurusan	: Statistika
Pembimbing	: Dr. Kartika Fithriasari, M.Si

ABSTRAK

Teknologi informasi merupakan salah satu hal yang tidak akan lepas dari kehidupan manusia. Tanpa adanya teknologi, manusia akan kesulitan dalam berkomunikasi dan menyampaikan informasi. Perlu adanya sistem yang secara otomatis yang dapat mengelompokkan berita sesuai dengan kategori berita dengan menggunakan text mining. Dalam penelitian ini, metode yang digunakan dalam klasifikasi adalah SVM dan KNN. KNN memiliki kelebihan dalam hal data training yang cukup banyak. Sebagai komparasi, dalam penelitian ini juga menggunakan SVM karena metode ini merupakan salah satu metode yang banyak digunakan untuk klasifikasi data, khususnya data teks. Kedua metode ini akan dibandingkan untuk mengetahui hasil ketepatan klasifikasi yang paling baik. Hasil dari penelitian ini bahwa SVM kernel linier dan kernel polynomial menghasilkan ketepatan klasifikasi yang paling baik adalah kernel linier. Apabila dibandingkan dengan KNN maka SVM lebih baik daripada KNN dengan hasil nilai rata-rata akurasi total, recall, precision dan F-Measure sebesar 93.2%, 93.2%, 93.63% dan 93.14%.

Kata Kunci: K-nearest neighbor,Support vector machine,Text Mining

Online News Classification Using Support Vector Machine and K-Nearest Neighbor

Name	: Siti Nur Asiyah
NRP	: 1314 105 016
Department	: Statistics
Supervisor	: Dr. Kartika Fithriasari, M.Si

ABSTRACT

Information technology is one thing that will not be separated from human life. Without the technology, humans would have difficulty in communicating and conveying information. It needs a system that can automatically categorize the news and the news category by using text mining. In this study, the method used in the classification is SVM and KNN. KNN has advantages in terms of training data that quite a lot. As a comparison, in this study also uses SVM because this method is one of many methods used for classification of data, in particular text data. Both of these methods will be compared to determine the accuracy of the classification results of the most good. Results from this study that the SVM linear kernel and kernel polynomial generating accuracy of the classification is best linear kernel. When compared with KNN then SVM is better than KNN with the results of the average value of total accuracy, recall, precision and F-Measure amounted to 93.2%, 93.2%, 93.63% and 93.14%.

Keywords: K-nearest neighbor, Support vector machine, Text Mining

DAFTAR ISI

Halaman

HALAMAN JUDUL i
TITLE PAGE iii
LEMBAR PENGESAHAN v
ABSTRAK vii
ABSTRACT ix
KATA PENGANTAR xi
DAFTAR ISI xiii
DAFTAR GAMBAR xv
DAFTAR TABEL xvii
DAFTAR LAMPIRAN xix

BAB I PENDAHULUAN
1.1 Latar Belakang 1
1.2 Rumusan Masalah 3
1.3 Tujuan Penelitian 3
1.4 Manfaat 3
1.5 Batasan Masalah 4

BAB II TINJAUAN PUSTAKA
2.1 *Text Mining* 5
2.2 *Pre Processing Text* 6
2.3 *Term Frequency Inverse Document Frequency* 7
2.4 *Support Vector Machine* 7
2.4.1 SVM pada *Linearly Separable Data* 8
2.4.2 SVM pada *Nonlinearly Separable Data* 11
2.4.3 Metode Kernel 12
2.4.4 *Multi Class* SVM 14
2.5 *K- Nearest Neighbor* 16
2.6 Pengukuran Performa 17
2.7 *K-Fold Cross Validation* 19
2.8 Penelitian Sebelumnya 20

BAB III METODOLOGI PENELITIAN
3.1 Sumber Data 21
3.2 Langkah Analisis 21

BAB IV ANALISIS DAN PEMBAHASAN
4.1 *Pre Processing Text* 25
4.2 *Support Vector Machine* dalam *Text Mining* 28
4.2.1 SVM Menggunakan Kernel Linier pada Data *Training* 28
4.2.2 SVM Menggunakan Kernel Polynomial pada Data *Training* 30
4.2.3 SVM Menggunakan Kernel Linier pada Data *Testing* 31
4.2.4 SVM Menggunakan Kernel Polynomial pada Data *Testing* 33
4.2.5 Model SVM *Multi Class* Menggunakan Kernel Polynomial Data Testing 35
4.3 K-*Nearest Neighbor* dalam *Text Mining* 37
4.3.1 K-NN pada Data *Training* 37
4.3.2 K-NN pada Data *Testing* 39
4.4 Perbandingan SVM dan K-NN 41

BAB V KESIMPULAN DAN SARAN
5.1 Kesimpulan 43
5.2 Saran 43
DAFTAR PUSTAKA 45
LAMPIRAN 49
BIODATA PENULIS 73

DAFTAR TABEL

Halaman

Tabel 2.1 SVM dengan Metode *One-against-one* 15
Tabel 2.2 *Confusion Matrix*.....18
Tabel 3.1 Struktur Data 21
Tabel 4.1 Contoh Hasil Proses *Stemming*.....25
Tabel 4.2 Contoh Hasil Proses *Stopword* dan *Case Folding*.....26
Tabel 4.3 Contoh Hasil *Tokenizing*.....27
Tabel 4.4 Frekuensi Kemunculan Kata Tertinggi Tiap Kategori.....27
Tabel 4.5 Ketepatan Klasifikasi SVM Kernel Linier pada Data *Training* 29
Tabel 4.6 Ketepatan Klasifikasi SVM Kernel Polynomial pada Data *Training*.....30
Tabel 4.7 Performansi Kernel Linier tiap Kategori pada Data *Testing* 31
Tabel 4.8 Performansi Kernel Linier tiap Fold pada Data *Testing* 32
Tabel 4.9 *Confusion Matrix* Kernel Linier 32
Tabel 4.10 Performansi Kernel Polynomial tiap Kategori pada Data *Testing* 33
Tabel 4.11 Performansi Kernel Polynomial tiap Fold pada Data *Testing* 34
Tabel 4.12 *Confusion Matrix* Kernel Polynomial.....34
Tabel 4.13 Pengukuran Performansi SVM.....35
Tabel 4.14 Ketepatan Klasifikasi KNN pada Data *Training*.....37
Tabel 4.15 Performansi KNN tiap Kategori pada Data *Training*.....37
Tabel 4.16 Performansi KNN tiap Fold pada Data *Training*.....38
Tabel 4.17 Performansi KNN tiap Kategori pada Data *Testing*.....39

Tabel 4.18 Performansi KNN tiap Fold pada Data *Testing* 40
Tabel 4.19 *Confusion Matrix* dengan Metode KNN 40
Tabel 4.20 Perbandingan Hasil Ketepatan Klasifikasi antara SVM dan KNN 41

DAFTAR GAMBAR

Halaman

Gambar 2.1 Konsep *Hyperplane* pada SVM.....8
Gambar 2.2 Data Spiral yang Menggambarkan Data *Nonlinier*.....12
Gambar 2.3 Tranformasi dari *Input Space* ke *Feature Space*.....13
Gambar 2.4 Contoh Klasifikasi dengan Metode *One-against-one*.....15
Gambar 3.1 Diagram Alir 22

BAB I
PENDAHULUAN

1.1 Latar Belakang

Teknologi informasi merupakan salah satu hal yang tidak akan lepas dari kehidupan manusia. Tanpa adanya teknologi, manusia akan kesulitan dalam berkomunikasi dan menyampaikan informasi. Teknologi informasi meliputi segala hal yang berkaitan dengan proses, penggunaan sebagai alat bantu, dan pengelolaan informasi. Sedangkan teknologi komunikasi adalah segala sesuatu yang berkaitan dengan penggunaan alat bantu untuk memproses dan mentransfer data dari perangkat satu ke perangkat lainnya. Awalnya, banyak instansi menyalurkan informasi kepada masyarakat melalui media televisi, koran, majalah atau radio. Kini, seiring berkembangnya teknologi, informasi disampaikan menggunakan sistem berbasis *web* secara *update.* Kementerian Komunikasi dan Informatika (Kemkominfo, 2014) menyatakan bahwa pengguna internet di Indonesia hingga saat ini telah mencapai 82 juta orang. Dengan capaian tersebut, Indonesia berada pada peringkat ke-8 di dunia.

Pada umumnya, berita yang disampaikan dalam *website* terdiri dari beberapa kategori seperti berita politik, olahraga, ekonomi, kesehatan, dan lain-lain (sebagai contoh pada website kompas.com, detik.com, dan vivanews.com). Sejauh ini, mengelompokkan berita dalam beberapa kategori tersebut dilakukan oleh editor secara manual. Prosesnya, sebelum diunggah harus terlebih dahulu diketahui isi berita secara keseluruhan untuk selanjutnya dikelompokkan dalam kategori yang tepat. Jika jumlah artikel berita yang diunggah semakin banyak, hal ini akan merepotkan bagi pengunggah berita. Terlebih jika dokumen sangat banyak dengan kategori yang cukup beragam. Hal tersebut akan menjadi beban kerja editor dalam mengelompokkan kategori berita. Permasalahan lain muncul ketika dokumen yang akan dikelompokkan dalam masing-masing kategori memiliki kemiripan isi. Hal ini

membutuhkan ketelitian dan waktu yang tidak sebentar dalam sistem pengelompokkan. Oleh karena itu, perlu adanya sistem yang secara otomatis dapat mengelompokkan berita sesuai dengan kategori berita dengan menggunakan *text mining*.

Metode *text mining* merupakan pengembangan dari metode *data mining* yang dapat diterapkan untuk mengatasi masalah tersebut. Algoritma-algoritma dalam *text mining* dibuat untuk dapat mengenali data yang sifatnya semi terstruktur misalnya sinopsis, abstrak maupun isi dari dokumen-dokumen (Gupta & Lehal, 2009). Sebelum suatu data teks dianalisis menggunakan metode dalam *text mining* perlu dilakukan *pre processing text* diantaranya adalah *tokenizing*, *case folding*, *stopwords*, dan *stemming.* Setelah dilakukan *pre processing*, selanjutnya dilakukan metode klasifikasi dalam mengelompokkan dalam masing-masing kategori. Klasifikasi merupakan suatu metode untuk memprediksi kategori atau kelas dari suatu item atau data yang telah didefinisikan sebelumnya. Berbagai macam metode klasifikasi banyak digunakan dalam melakukan klasifikasi berupa teks diantaranya adalah *Naïve Bayes Classifier* (NBC), *K-Nearest Neigbour* (KNN), *Artificial Neural Network* (ANN), dan *Support Vector Machines* (SVM).

Penelitian sebelumnya yang berkaitan adalah oleh Ariadi (2015) tentang klasifikasi berita Indonesia menggunakan metode NBC dan SVM dengan *Confix Stripping Stemmer* menghasilkan ketepatan klasifikasi sebesar 88,1%. Selain itu oleh Buana dan Putra (2012) tentang kombinasi KNN dan K-Mean untuk klasifikasi Koran Indonesia menghasilkan ketepatan klasifikasi sebesar 87%.

Dalam penelitian ini, metode yang digunakan dalam klasifikasi adalah SVM dan KNN. KNN memiliki kelebihan dalam hal data training yang cukup banyak. Sebagai komparasi, dalam penelitian ini juga menggunakan SVM karena metode ini merupakan salah satu metode yang banyak digunakan untuk klasifikasi data, khususnya data teks. Menurut Nugroho (2003) salah satu kelebihan SVM dapat diimplementasikan relatif

mudah, karena proses penentuan *support vector* dapat dirumuskan dalam QP problem. Selanjutnya akan dilakukan perbandingan dari kedua metode tersebut pada artikel berita *online.*

1.2 Rumusan Masalah

Klasifikasi artikel berita secara umum banyak menggunakan metode SVM sedangkan terdapat metode baru yang dirasa lebih baik dari metode tersebut yaitu KNN. Kedua metode tersebut akan dibandingkan, metode mana yang menghasilkan tingkat galat paling kecil. Berdasarkan penjelasan tersebut maka permasalahan yang akan dibahas yaitu Bagaimana hasil ketepatan klasifikasi artikel berita *online* dengan menggunakan metode SVM dan KNN.

1.3 Tujuan Penelitian

Tujuan dari penelitian yang ingin dicapai berdasarkan rumusan masalah ini adalah sebagai berikut.

1. Mengetahui hasil ketepatan klasifikasi artikel berita *online* dengan menggunakan metode SVM.
2. Mengetahui hasil ketepatan klasifikasi artikel berita *online* dengan menggunakan metode KNN.
3. Mengetahui hasil ketepatan yang paling baik diantara dua metode SVM dan KNN.

1.4 Manfaat

Hasil penelitian ini diharapkan dapat bermanfaat dalam bidang klasifikasi teks secara umum dengan menggunakan metode SVM dan KNN. Penelitian ini juga diharapkan dapat membantu editor dalam efisiensi waktu dan efisiensi kerja dalam pengkategorian artikel maupun berita sehingga dapat dilakukan secara otomatis.

1.5 Batasan Masalah

Dalam penelitian ini terdapat beberapa batasan masalah yang digunakan adalah sebagai berikut.

1. Data yang digunakan merupakan berita pada situs berita *online* www.detik.com.
2. Kategori yang digunakan sebanyak 5 kategori, yaitu *news, finance, hot, sport* dan *oto*.
3. Artikel berita diambil pada bulan Februari hingga Juni 2016.
4. Pengambilan sampel artikel berita dilakukan sesuai keinginan peneliti.

BAB II
TINJAUAN PUSTAKA

Pada penelitian ini *text mining* digunakan untuk mengelompokkan berita sesuai dengan kategori berita. Klasifikasi merupakan suatu metode untuk memprediksi kategori atau kelas dari suatu item atau data yang telah didefinisikan sebelumnya.Beberapa contoh metode yang dapat digunakan untuk klasifikasi suatu data teks adalah SVM dan KNN.

2.1 *Text Mining*

Text mining merupakan salah satu cabang ilmu *data mining* yang menganalisis data berupa dokumen teks. Menurut Han, Kamber, dan Pei (dalam Prilianti dan Wijaya, 2014), *text mining* adalah satu langkah dari analisis teks yang dilakukan secara otomatis oleh komputer untuk menggali informasi yang berkualitas dari suatu rangkaian teks yang terangkum dalam sebuah dokumen. Ide awal pembuatan *text mining* adalah untuk menemukan pola-pola informasi yang dapat digali dari suatu teks yang tidak terstruktur (Hamzah, 2012). Dengan demikian, *text mining* mengacu juga kepada istilah *text data mining* (Hearst,1997) atau penemuan pengetahuan dari basis data teks (Feldman dan Dagan dalam Hamzah, 2012). Saat ini, *text mining* telah mendapat perhatian dalam berbagai bidang, antara lain dibidang keamanan, biomedis, pengembangan perangkat lunak dan aplikasi, media *online*, pemasaran, dan akademik. Seperti halnya dalam *data mining*, aplikasi *text mining* pada suatu studi kasus, harus dilakukan sesuai prosedur analisis. Langkah awal sebelum suatu data teks dianalisis menggunakan metode-metode dalam *text mining* adalah melakukan *pre-processing* teks. Selanjutnya, setelah didapatkan data yang siap diolah, analisis *text mining* dapat dilakukan.

Text mining dapat digunakan untuk proses penemuan *rule* baru dengan algoritma pengelompokan, asosiasi, dan *ranking*. Diantara beberapa fungsi tersebut, yang paling banyak dilakukan adalah proses pengelompokan. Terdapat dua jenis metode

pengelompokan teks, yaitu *text clustering* dan *text classification*. Menurut Darujati dan Gumelar (2012), *text clustering* berhubungan dengan proses menemukan sebuah struktur kelompok yang belum terlihat (tak terpandu atau *unsupervised*) dari sekumpulan dokumen. Sedangkan *text classification* dapat dianggap sebagai proses untuk membentuk golongan-golongan (kelas-kelas) dari dokumen berdasarkan pada kelas kelompok yang sudah diketahui sebelumnya (terpandu atau *supervised*). Berdasarkan pengertian ini, dapat dinyatakan bahwa proses klasifikasi (*supervised*) merupakan proses yang lebih mudah dilakukan *monitoring*, karena terdapat target kelas yang akan dituju dalam analisisnya.

2.2 *Pre Processing Text*

Tahapan *pre processing* ini dilakukan agar dalam klasifikasi dapat diproses dengan baik. Tahapan dalam *pre processing text* adalah sebagai berikut:

a. *Case Folding*, merupakan proses untuk mengubah semua karakter pada teks menjadi huruf kecil. Karakter yang diproses hanya huruf ‚a" hingga ‚z" dan selain karakter tersebut akan dihilangkan seperti tanda baca titik (.), koma (,), dan angka. (Weiss, 2010)
b. *Tokenizing*, merupakan proses memecah yang semula berupa kalimat menjadi kata-kata atau memutus urutan string menjadi potongan-potongan seperti kata-kata berdasarkan tiap kata yang menyusunnya.
c. *Stopwords*, merupakan kosakata yang bukan merupakan kata unik atau ciri pada suatu dokumen atau tidak menyampaikan pesan apapun secara signifikan pada teks atau kalimat (Dragut, Fang, Sistla, Yu, & Meng, 2009). Kosakata yang dimaksudkan adalah kata penghubung dan kata keterangan yang bukan merupakan kata unik misalnya “sebuah”, “oleh”, “pada”, dan sebagainya.
d. *Stemming*, yakni proses untuk mendapatkan kata dasar dengan cara menghilangkan awalan, akhiran, sisipan, dan *confixes* (kombinasi dari awalan dan akhiran).

2.3 *Term Frequency Inverse Document Frequency*

Term Frequency Inverse Document Frequency (TF-IDF) merupakan pembobot yang dilakukan setelah ekstrasi artikel berita. Proses metode TF-IDF adalah menghitung bobot dengan cara integrasi antara *term frequency* (*tf*) dan *inverse document frequency* (*idf*). Langkah dalam TF-IDF adalah untuk menemukan jumlah kata yang kita ketahui (*tf*) setelah dikalikan dengan berapa banyak artikel berita dimana suatu kata itu muncul (*idf*) (Jamhari, Noersasongko, & Subagyo,2014). Rumus dalam menentukan pembobot dengan TF-IDF adalah sebagai berikut :

$$w_{ij} = tf_{ij} \times idf$$
$$idf = \log\left(\frac{N}{df_j}\right) \tag{2.1}$$

dengan : i = 1,2,…,p (Jumlah variabel)
j =1,2,…,N (Jumlah data)

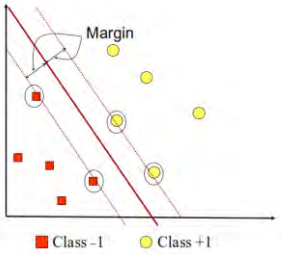
Dimana w_{ij} adalah bobot dari kata i pada artikel ke j, N merupakan jumlah seluruh dokumen, tf_{ij} adalah jumlah kemunculan kata i pada dokumen j, df_j adalah jumlah artikel j yang mengandung kata i. TF-IDF dilakukan agar data dapat dianalisis dengan menggunakan *support vector machine*.

2.4 *Support Vector Machine*

Support Vector Machine (SVM) adalah sistem pembelajaran yang menggunakan hipotesis fungsi linear dalam ruang berdimensi tinggi dan dilatih dengan algoritma berdasarkan teori optimasi dengan menerapkan *learning bias* yang berasal dari teori statistik. Liliana (dalam Cristianini & Shawe-Taylor, 2000). Tujuan utama dari metode ini adalah untuk membangun OSH (*Optimal Separating Hyperplane*), yang membuat fungsi pemisahan optimum yang dapat digunakan untuk klasifikasi.

2.4.1 SVM pada *Linearly Separable Data*

Linearly separable data merupakan data yang dapat dipisahkan secara linier. Misalkan $\mathbf{x}_i = \{\mathbf{x}_i, \ldots, \mathbf{x}_n\}$, $\mathbf{x}_i \in \Re^n$ adalah data set dan $y_i \in \{+1, -1\}$ adalah label kelas dari data x_i. Pada Gambar 2.1 dapat dilihat berbagai alternatif bidang pemisah yang dapat memisahkan semua data set sesuai dengan kelasnya. Bidang pemisah terbaik tidak hanya dapat memisahkan data tetapi juga memliki margin yang paling besar.



Gambar 2.1 Konsep *Hyperplane* pada SVM (Sumber : Nugroho,2003)

Data yang berada pada bidang pembatas disebut dengan *support vector*. Dalam Gambar 2.1, dua kelas dapat dipisahkan oleh sepasang bidang pembatas yang sejajar. Bidang pembatas pertama membatasi kelas pertama sedangkan bidang pembatas kedua membatasi kelas kedua, sehingga diperoleh:

$$\begin{aligned} \mathbf{x_i w} + b \geq +1, y_i = +1 \\ \mathbf{x_i w} + b \leq -1, y_i = -1 \end{aligned} \qquad (2.2)$$

$i = 1,2,\ldots, p$

w adalah normal bidang dan *b* adalah posisi bidang alternatif terhadap pusat koordinat. Nilai margin (jarak) antara bidang pembatas (berdasarkan rumus jarak garis ke titik pusat) adalah $\frac{1-b-(-1-b)}{\|\mathbf{w}\|} = \frac{2}{\|\mathbf{w}\|}$. Nilai margin ini dimaksimalkan dengan tetap memenuhi persamaan (2.2). Dengan mengalikan *b* dan **w**

dengan sebuah konstanta, akan dihasilkan nilai margin yang dikalikan dengan konstata yang sama. Oleh karena itu, konstrain pada persamaan (2.2) merupakan *scaling constraint* yang dapat dipenuhi dengan *rescaling* b dan $\mathbf{w}$. Selain itu karena memaksimalkan $\frac{1}{\|\mathbf{w}\|}$ sama dengan meminimumkan $\|\mathbf{w}\|^2$ dan jika kedua bidang pembatas pada persamaan (2.2) direpresentasikan dalam pertidaksamaan (2.3),

$$y_i\left(\mathbf{x_i}\mathbf{w}+b\right)-1\geq 0 \tag{2.3}$$

maka pencarian bidang pemisah terbaik dengan nilai margin terbesar dapat dirumuskan menjadi masalah optimasi konstrain, yaitu:

$$\begin{aligned}&\min \frac{1}{2}\|\mathbf{w}\|^2 \\ &\text{dengan } y_i\left(\mathbf{x_i}\mathbf{w}+b\right)-1\geq 0\end{aligned} \tag{2.4}$$

Persoalan (2.4) ini akan lebih mudah diselesaikan jika diubah ke dalam formula *lagrangian* yang menggunakan *lagrange multiplier*. Dengan demikian permasalah optimasi konstrain dapat diubah menjadi:

$$\min_{w,b} L_p(\mathbf{w},b,\alpha)=\frac{1}{2}\|\mathbf{w}\|^2-\sum_{i=1}^{n}\alpha_i y_i\left(x_i\mathbf{w}+b\right)+\sum_{i=1}^{n}\alpha_i \tag{2.5}$$

dengan tambahan konstrain, $\alpha_i\geq 0$ (nilai dari koefisien *lagrange*). Dengan meminimumkan L_p terhadap $\mathbf{w}$ dan b, maka dari $\frac{\partial L_p(\mathbf{w},b,\alpha)}{\partial \mathbf{w}}=0$ dan dari $\frac{\partial L_p(\mathbf{w},b,\alpha)}{\partial b}=0$,diperoleh persamaan (2.5).

$$w=\sum_{i=1}^{n}\alpha_i y_i \mathbf{x}_i \quad \text{dan} \quad \sum_{i=1}^{n}\alpha_i y_i=0 \tag{2.6}$$

vektor $\mathbf{w}$ sering kali bernilai besar (tak terhingga), tetapi nilai α_i terhingga. Untuk itu formula *lagrangian* L_P (*primal problem*)

diubah ke dalam L_D (*dual problem*). Dengan mensubtitusikan persamaan (2.6) ke persamaan (2.5), diperoleh L_D.

$$L_D = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \mathbf{x}_j \tag{2.7}$$

Jadi persoalan pencarian bidang pemisah terbaik dapat dirumuskan pada persamaan sebagai berikut:

$$\text{dengan } \max_{\alpha} L_D = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \mathbf{x}_j \tag{2.8}$$

$$\sum_{i=1}^{n} \alpha_i y_i = 0, \alpha_i \geq 0$$

Dengan demikian, dapat diperoleh dari nilai α_i yang nantinya digunakan untuk menemukan **w**. Terdapat nilai α_i untuk setiap data *training*, data *training* yang memiliki nilai $\alpha_i > 0$ adalah *support vector* sedangkan sisanya memiliki nilai $\alpha_i = 0$. Dengan demikian fungsi keputusan yang dihasilkan hanya dipengaruhi oleh *support vector*.

Formula pencarian bidang pemisah terbaik ini adalah permasalah *quadratic programming* sehingga nilai maksimum global dari α_i selalu dapat ditemukan. Setelah solusi permasalahan *quadratic programming* ditemukan (nilai α_i), maka kelas dari suatu data x testing dapat ditentukan dengan persamaan sebagai berikut:

$$f(x_d) = \sum_{i=1}^{ns} \alpha_i y_i \mathbf{x_i} \mathbf{x_d} + b \tag{2.9}$$

Dengan x_i adalah *support vector*, ns adalah jumlah *support vector*, dan x_d adalah data yang akan diklasifikasikan. Sedangkan nilai *b* didapatkan dengan persamaan

$$b = \frac{1}{2} \mathbf{w} \left[\mathbf{x_r} + \mathbf{x_s} \right] \tag{2.10}$$

$\mathbf{x}_r$ dan $\mathbf{x}_s$ adalah *support vector* untuk tiap-tiap kelas dengan syarat persamaan $\alpha_r, \alpha_s > y_r = 1, y_s = -1$ (Sembiring,2007).

2.4.2 SVM pada *Nonlinearly Separable Data*

Untuk mengklasifikasikan data yang tidak dapat dipisahkan secara linier formula SVM harus dimodifikasi karena tidak akan ada solusi yang ditemukan. Oleh karena itu, kedua bidang pembatas (2.2) harus diubah sehingga lebih fleksibel dengan penambahan variabel $\xi_i (\xi_i \geq 0, \forall_i : \xi_i = 0$ jika x_i diklasifikasikan dengan benar) menjadi $\mathbf{x_i w} + b \geq 1 - \xi_i$ untuk kelas 1 dan $\mathbf{x_i w} + b \leq -1 + \xi_i$ untuk kelas 2. Pecarian bidang pemisah terbaik dengan penambahan variabel ξ_i sering disebut dengan *soft margin hyperplane*. Dengan demikian formula pencarian bidang pemisah terbaik berubah menjadi:

$$\min \frac{1}{2}\|\mathbf{w}\|^2 + C\left(\sum_{i=1}^{n} \xi_i\right) \tag{2.11}$$

dengan $y_i(\mathbf{x_i w} + b) \geq 1 - \zeta_i$

$\xi_i \geq 0$

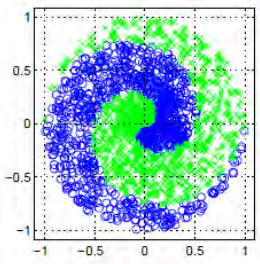
Menurut Osuna (dalam Sembiring,2007) *C* adalah parameter yang menentukan besar penalti akibat kesalahan dalam klasifikasi data dan nilainya ditentukan oleh pengguna. Sehingga peran dari *C* adalah meminimalkan kesalahan pelatihan dan mengurangi kompleksitas model. Selanjutnya bentuk *primal problem* sebelumnya berubah menjadi:

$$\min_{w,b} L_p(\mathbf{w}, b, \alpha) = \frac{1}{2}\|\mathbf{w}\|^2 + C\left(\sum_{i=1}^{n} \xi_i\right) - \sum_{i=1}^{n} \alpha_i \left\{y_i\left(\mathbf{x_i w} + b\right) - 1 + \zeta_i\right\} + \sum_{i=1}^{n} \mu_i \xi_i \tag{2.12}$$

Pengubahan Lp ke dalam *dual problem*, menghasilkan persamaan yang sama dengan (2.6) sehingga pencarian bidang pemisah terbaik dilakukan dengan cara yang hampir sama dengan data linier, tetapi rentang nilai α_i adalah $0 \geq \alpha_i \geq C$.

2.4.3 Metode Kernel

Metode lain untuk mengklasifikasikan data yang tidak dapat dipisahkan secara linier adalah dengan metode kernel. Metode kernel mentransformasikan data ke dalam dimensi ruang fitur (*feature space*) sehingga dapat dipisahkan secara linier pada *feature space*. Sebagai contoh data yang tidak dipisahkan secara linier adalah Gambar 2.2

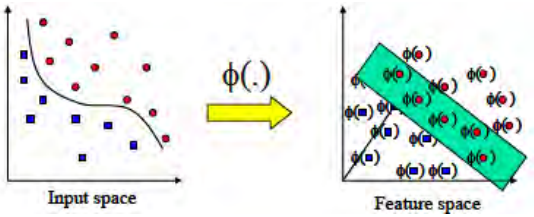


Gambar 2.2 Data Spiral yang Menggambarkan Data *Nonlinier* (Sumber : Sembiring,2007)

Suatu data *x* di input space ke feature space dengan menggunakan fungsi transformasi $\mathbf{x}_k \rightarrow \phi(\mathbf{x}_k)$. Sehingga nilai $\mathbf{w} = \sum_{i=1}^{ns} \alpha_i y_i \phi(\mathbf{x_i})$ dan fungsi hasil pembelajaran yang dihasilkan adalah

$$f(x_d) = \sum_{i=1}^{ns} \alpha_i y_i \phi(\mathbf{x_i}) \phi(\mathbf{x_d}) + b \qquad (2.13)$$

Feature space dalam prakteknya biasanya memiliki dimensi yang tinggi dari vektor *input space*. Hal ini mengakibatkan komputasi pada *feature space* mungkin sangat besar, karena ada kemungkinan *feature space* dapat memiliki jumlah *feature* yang tidak terhingga.



Gambar 2.3 Transformasi dari *Input Space* ke *Feature Space*

(Sumber : Sembiring,2007)

Selain itu, sulit mengetahui fungsi transformasi yang tepat. "*Kernel Trick*" digunakan untuk mengatasi masalah ini pada SVM. Dari persamaan (2.13) dapat dilihat *dot product* $\phi(x_i)\phi(x_d)$. Jika terdapat sebuah fungsi kernel K sehingga $K(\mathbf{x_i}, \mathbf{x_d}) = \phi(x_i)\phi(x_d)$, maka fungsi transformasi $\phi(\mathbf{x_k})$ tidak perlu diketahui secara persis. Dengan demikian fungsi yang dihasilkan dari training adalah

$$f(x_d) = \sum_{i=1}^{ns} \alpha_i y_i K(\mathbf{x_i}, \mathbf{x_d}) + b \qquad (2.14)$$

syarat sebuah fungsi menjadi fungsi kernel adalah memenuhi teorema Mercer yang menyatakan bahwa matriks kernel yang dihasilkan harus bersifat *positive semi-definite*. Fungsi kernel yang umum digunakan pada metode SVM adalah

1. Kernel Linier
   $K(\mathbf{x_i}, \mathbf{x}) = \mathbf{x_i^T x}$
2. Kernel Polynomial
   $K(\mathbf{x_i}, \mathbf{x}) = (\gamma \mathbf{x_i^T x} + r)^p, \gamma > 0$
3. Kernel Radial Basis Function (RBF)
   $K(\mathbf{x_i x}) = \exp(-\gamma \|\mathbf{x} - \mathbf{x_i}\|^2)$
4. Sigmoid Kernel
   $K(\mathbf{x_i}, \mathbf{x}) = \tanh(\gamma \mathbf{x_i^T x} + r)$

Dari keempat metode kernel yang ada, fungsi kernel RBF direkomendasikan untuk diuji pertama kali. Fungsi kernel RBF memiliki performansi yang sama dengan kernel linier pada parameter tertentu, memiliki perilaku seperti fungsi kernel *sigmoid* dengan parameter tertentu dan rentang nilai kecil [0,1] (Hsu, Chang, & Lin, 2010). Sedangkan dalam kasus data teks mayoritas peneliti menggunakan ketiga jenis kernel pertama. Diantaranya adalah membandingkan ketepatan klasifikasi dengan kernel linier dan kernel RBF dimana hasilnya lebih baik kernel linier (Guduru, 2006) dan membandingkan antara kernel polynomial dan kernel RBF hasilnya adalah RBF yang lebih baik (Joachims, 1998).

2.4.4 *Multi Class* SVM

Dalam kasus klasifikasi berita ini terdiri dari lebih 2 kategori. Terdapat dua pendekatan yang sering digunakan dalam mengimplementasikan *multi class* SVM yaitu *One Against All* (OAA) dan *One Against One* (OAO). Pendekatan dengan metode OAO, diperlukan untuk menemukan fungsi pemisah sebanyak $k(k\text{-}1)/2$, dimana setiap model klasifikasi dilatih pada data dari dua kelas. Untuk data pelatihan kelas ke-i dan kelas ke-j, dilakukan pencarian solusi untuk persoalan optimasi konstrain sebagai berikut:

$$\min_{w^{ij} b^{ij} \xi^{ij}} \frac{1}{2}\left(w^{ij}\right)^T w^{ij} + C\sum_t \xi_t^{ij} \tag{2.15}$$

$$\text{dengan } \left(w^{ij}\right)^T \phi\left(x_t\right) + b^{ij} \geq 1 - \xi_t^{ij} \rightarrow y_t = i,$$

$$\left(w^{ij}\right)^T \phi\left(x_t\right) + b^{ij} \geq -1 + \xi_t^{ij} \rightarrow y_t = j,$$

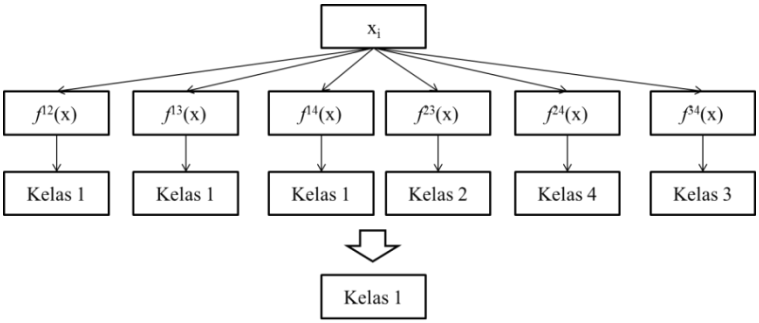
$$\xi_t^{ij} \geq 0$$

Terdapat beberapa metode melakukan pengujian setelah keluruhan $k(k\text{-}1)/2$ model klasifikasi selesai dibangun. Salah satunya adalah metode voting (Hsu, Chih-wei, dan Lin,2002).

Jika data x dimasukkan ke dalam fungsi pelatihan $\left(f(x)=\left(w^{ij}\right)^T\phi(x)+b\right)$ dan hasilnya menyatakan x adalah kelas *i*, maka suara untuk kelas *i* ditambah satu. Kelas dari data x akan ditentukan dari jumlah suara terbanyak. Jika terdapat dua buah kelas yang jumlah suaranya sama, maka kelas yang indeksnya lebih kecil dinyatakan sebagai kelas dari data. Metode voting akan diilustrasikan dalam Tabel 2.1 dan Gambar 2.4.

Tabel 2.1 SVM dengan Metode *One-against-one*

$y_i = 1$	$y_i = -1$	Hipotesis
Kelas 1	Kelas 2	$f^{12}(x)=\left(w^{12}\right)x+b^{12}$
Kelas 1	Kelas 3	$f^{13}(x)=\left(w^{13}\right)x+b^{13}$
Kelas 1	Kelas 4	$f^{14}(x)=\left(w^{14}\right)x+b^{14}$
Kelas 2	Kelas 3	$f^{23}(x)=\left(w^{23}\right)x+b^{23}$
Kelas 2	Kelas 4	$f^{24}(x)=\left(w^{24}\right)x+b^{24}$
Kelas 3	Kelas 4	$f^{34}(x)=\left(w^{34}\right)x+b^{34}$



Gambar 2.4 Contoh Klasifikasi dengan Metode *One-against-one*

Jadi pada pendekatan ini terdapat $k(k\text{-}1)/2$ buah permasalahan *quadratic programming* yang masing-masing memiliki $2n / k$ variabel (n adalah jumlah data pelatihan). Contohnya terdapat permasalahan dengan 4 buah kelas. Oleh karena itu, digunakan 6 buah SVM biner seperti pada Tabel 2.1 dan contoh penggunaanya dalam memprediksi kelas data baru dapat dilihat pada Gambar 2.4

2.5 *K-Nearest Neighbor*

KNN merupakan salah satu pendekatan yang sederhana untuk diimplementasikan dan merupakan metode lama yang digunakan dalam pengklasifikasian. Menurut Y. Hamamoto, dkk dan E.Alpaydin pada tahun 1997 menyebutkan bahwa KNN memiliki tingkat efisiensi yang tinggi dan dalam beberapa kasus memberikan tingkat akurasi yang tinggi dalam hal pengklasifikasian.

Dalam istilah lain, K-*Nearest Neighbor* merupakan salahsatu metode yang digunakan dalam pengklasifikasian. Prinsip kerja K-*Nearest Neighbor* (KNN) adalah melakukan klasifikasi berdasarkan kedekatan lokasi (jarak) suatu data dengan data yang lain (Prasetyo, 2012). Dekat atau jauhnya lokasi (jarak) bisa dihitung melalui salah satu dari besaran jarak yang telah ditentukan yakni jarak *Euclidean*, jarak *Minkowski*, dan jarak *Mahalanobis* (Deokar, 2009). Namun dalam penerapannya seringkali digunakan jarak *Euclidean* karena memiliki tingkat akurasi dan juga *productivity* yang tinggi (Dasarathy, 1990). Konsep jarak *Euclidean* ini memperlakukan semua peubah adalah bebas (tidak berkorelasi). Transformasi baku yang dilakukan berarti menghilangkan pengaruh keragaman data atau dengan kata lain semua peubah akan memberikan kontribusi yang sama untuk jarak. Jarak *Euclidean* adalah besarnya jarak suatu garis lurus yang menghubungkan antar objek. Rumus jarak *Euclidean* adalah sebagai berikut (Yang dan Liu, 1999):

$$d(x_i, x_j) = \sqrt{\sum_{n=1}^{p} (x_{ip} - x_{jp})^2} \quad (2.16)$$

Dengan:

x_{ip} = data *testing* ke-*i* pada variabel ke-*p*

x_{jp} = data *training* ke-*j* pada variabel ke-*p*

$d(x_i, x_j)$ = jarak *euclidean*

p = dimensi data variabel bebas

2.6 Pengukuran Performa

Pengukuran performa dilakukan untuk melihat hasil yang didapatkan dari klasifikasi. Terdapat beberapa cara untuk mengukur performa, beberapa cara yang sering digunakan adalah dengan menghitung akurasi, *recall,* dan *precission* (Hotto dkk, 2005). Akurasi merupakan persentase dari total dokumen yang teridentifikasi secara tepat dalam proses klasifikasi. *Recall* mengindikasikan sebagian kecil dari dokumen yang relevan diambil. *Precision* mengkuantifikasi fraksi dokumen diambil yang sebenarnya relevan, dalam contoh milik kelas sasaran. Adapun *performance* algoritma klasifikasi *data mining* biasanya dinilai dari tingkat akurasi, yaitu persentase *tuples* yang berada pada kelas yang tepat(Dunham,2003). Misalkan, jika terdapat suatu kelas Cj dan *tuple* atau baris *database* ti, maka terdapat 4 kuadran kemungkinan penempatan *tuple* ti ke dalam kelas-kelas yang ada. Hal ini dapat dilihat pada Tabel 2.2 adalah sebagai berikut.

Tabel 2.2 *Confusion Matrix*

		Prediksi				
		kelas 1	kelas 2	kelas 3	kelas 4	kelas 5
Aktual	kelas 1	F_{11}	F_{12}	F_{13}	F_{14}	F_{15}
	kelas 2	F_{21}	F_{22}	F_{23}	F_{24}	F_{25}
	kelas 3	F_{31}	F_{32}	F_{33}	F_{34}	F_{35}
	kelas 4	F_{41}	F_{42}	F_{43}	F_{44}	F_{45}
	kelas 5	F_{51}	F_{52}	F_{53}	F_{54}	F_{55}

(Arifin,2015)

Rumus penentuan akurasi total, *recall,* dan *precision,* berdasarkan *table of confusion* adalah sebagai berikut

$$akurasi\ \ total = \frac{F_{11}+F_{22}+F_{33}+F_{44}+F_{55}}{F_{11}+F_{12}+F_{13}+F_{14}+F_{15}+\cdots+F_{51}+F_{52}+F_{53}+F_{54}+F_{55}} \quad (2.17)$$

Dari Tabel 2.2, ukuran performansi *recall* untuk kelas 1 dapat dihitung menggunakan persamaan (2.18).

$$recall = \frac{F_{aa}}{\sum_{b=1}^{B} F_{ab}}$$

Dimana a=1,2,…,A
b=1,2,…,B

$$recall\ \ kategori\ \ 1 = \frac{F_{11}}{F_{11}+F_{12}+F_{13}+F_{14}+F_{15}} \quad (2.18)$$

Persamaan 2.18 dapat diartikan proporsi ketepatan klasifikasi data aktual yang berasal dari data prediksi kelas 1. Perhitungan untuk *recall* pada kelas 2 sampai kelas 5 disesuaikan dengan Tabel 2.2. Selanjutnya digunakan pengukuran performansi yang lainnya yaitu *precision*. Berikut merupakan persamaan untuk kelas 1 dapat dilihat pada persamaan (2.19).

$$precision = \frac{F_{aa}}{\sum_{a=1}^{A} F_{ab}}$$

$$precision \quad kategori \quad 1 = \frac{F_{11}}{F_{11} + F_{21} + F_{31} + F_{41} + F_{51}} \tag{2.19}$$

Persamaan 2.19 dapat diartikan proporsi ketepatan klasifikasi data yang diprediksi pada kelas 1 yang berasal dari kelas data aktual pada seluruh kelas. Sama seperti *recall*, apabila ingin mencari kelas 2 sampai kelas 5 maka disesuaikan dengan Tabel 2.2.

F-measure merupakan kompromi dari *recall* dan *precision* untuk mengukur kinerja keseluruhan pengklasifikasi. Berikut merupakan cara perhitungan *f-measure*. (Hotho dkk, 2005)

$$F = \frac{2 \times recall \times precision}{recall + precision} \tag{2.20}$$

2.7 *K- Fold Cross Validation*

K - fold cross validation adalah sebuah teknik yang menggunakan keseluruhan *dataset* yang ada sebagai *training dan testing* (Bengio, 2004). Teknik ini mampu melakukan pengulangan data *training* dan data testing dengan algoritma *k* pengulangan dan partisi 1/*k* dari *dataset*, yang mana 1/*k* tersebut akan digunakan sebagai data testing. Sebagai analogi misalkan keseluruhan *dataset* dibagi menjadi *k* buah subbagian *Ak* dengan ukuran sama, yang mana *Ak* merupakan himpunan bagian dari *dataset*. Kemudian dari data itu dilakukan iterasi sebanyak *k* kali. Pada iterasi ke *k*, subset *Ak* menjadi data *testing*, sedangkan subbagian lain menjadi data *training*. Hal ini ditujukan agar mendapatkan tingkat kepercayaan yang tinggi karena semua *dataset* dilibatkan sebagai data *training* maupun *testing* (Schneider, 1997). Akurasi total yang dihasilkan dengan

menggunakan metode *k-fold cross validation* adalah sebagai berikut

$$CVA = \frac{1}{k}\sum_{i=1}^{k} A_i \tag{2.21}$$

Dimana k adalah banyaknya *fold* yang digunakan, dan A_i adalah hasil akurasi total yang didapatkan dari setiap *fold.*

2.8 Penelitian Sebelumnya

Ariadi (2015) telah melakukan penelitian mengenai klasifikasi berita menggunakan metode NBC dan SVM dengan *confix stripping stemmer*. Pada penelitian tersebut didapatkan Metode *Support Vector Machine* antara kernel RBF dan kernel linier didapatkan hasil kernel linier sama baiknya dengan kenel RBF pada *word vector* 10000 dalam melakukan klasifikasi berita Indonesia. Menggunakan data *testing* didapatkan untuk tiap pengukuran performa akurasi, *precision, recall,* dan *F-Measure* adalah 88,1%, 89,1%, 88,1%, dan 88,3%. Perbandingan antara kedua metode NBC dan SVM didapatkan hasil SVM kernel RBF dan linier lebih baik dibandingkan dengan NBC.

Buana dan Putra (2012) melakukan penelitian mengenai kombinasi K-NN dan K-Mean untuk klasifikasi koran Indonesia.. penelitian tersebut didapatkan hasil dengan mnggunakan k=5 didapatkan hasil akurasi dari *f-measure* sebesar 87% .

BAB III
METODOLOGI PENELITIAN

3.1 Sumber Data

Sumber data yang akan digunakan dalam penelitian ini adalah artikel berita pada koran online detik.com yang terdiri dari 5 kategori. Kategori tersebut adalah *news*, *finance*,*hot* , *sport*, dan *oto*. Tiap kategori akan diambil sebanyak 100 artikel sehingga data artikel keseluruhan berjumlah 500 dengan jumlah variabel sebanyak 3784 *word vector*. Struktur data artikel yang telah dilakukan tahapan *pre processing text* adalah seperti pada Tabel 3.1

Tabel 3.1 Struktur Data

No	Y	X_1	X_2	…	X_{3784}
1	1	$x_{1,1,1}$	$x_{1,1,2}$	…	$x_{1,1,3784}$
2	1	$x_{2,1,1}$	$x_{2,1,2}$	…	$x_{2,1,3784}$
3	1	$x_{3,1,1}$	$x_{3,1,2}$	…	$x_{3,1,3784}$
⋮	⋮	⋮	⋮	⋮	⋮
500	5	$x_{500,5,1}$	$x_{500,5,2}$	…	$x_{500,5,3784}$

3.2 Langkah Analisis

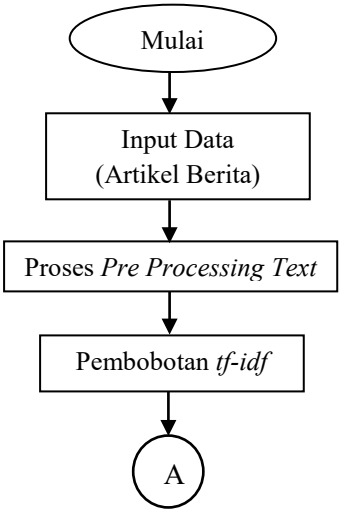
Langkah analisis data yang dilakukan pada penelitian ini adalah sebagai berikut.

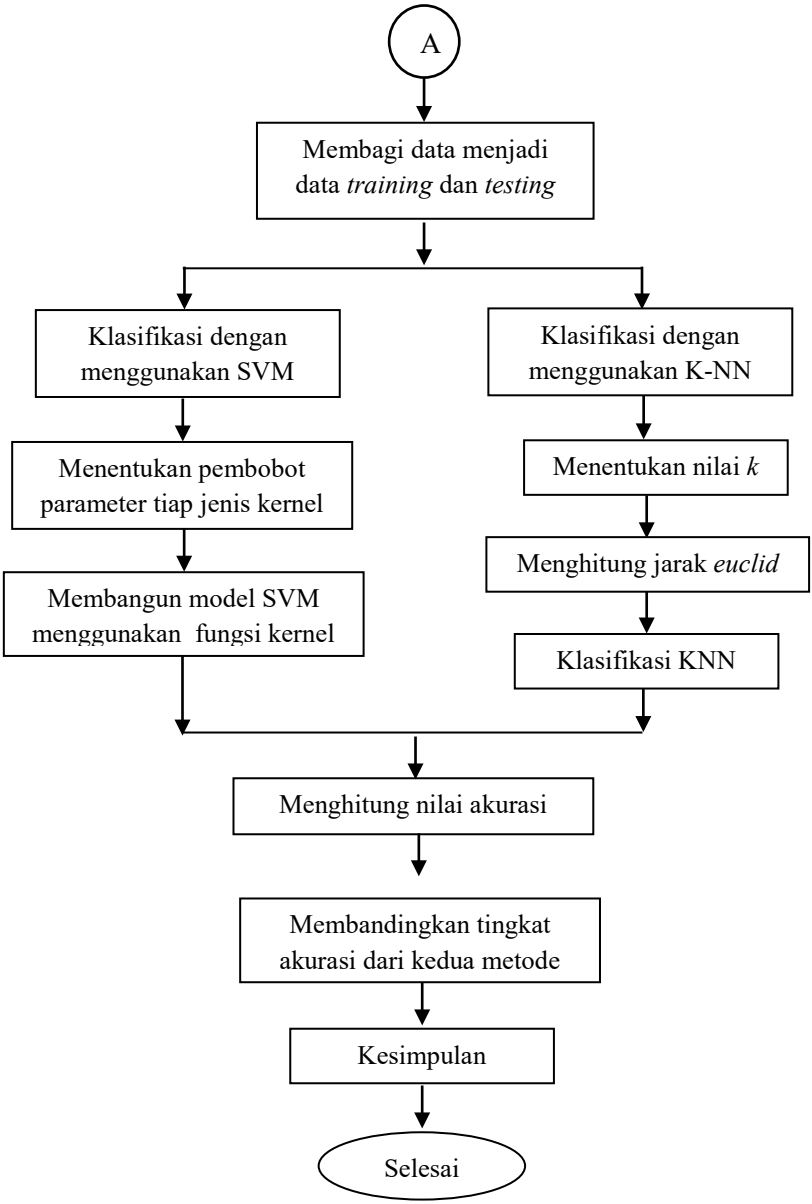
1. Menyiapkan data artikel
2. Melakukan *pre processing text* yaitu *stemming, stopword, casefolding* dan *tokenizing.*
   a) Proses *stemming* menyiapkan data artikel dalam bentuk excel kemudian dilakukan *running* dengan menggunakan *xampp*
   b) Tahap *stopword* dan *casefolding* yaitu hasil dari *stemming* di *running* menggunakan *software* R. Daftar *stopwords* diambil dari tesis F. Tala yang berjudul "*A Study of Stemming Effect on I nformation Retrieval in Bahasa Indonesia*".

c) Pada tahap *tokenizing* hasil dari *casefolding* dilakukan running data pada *software* Weka.
d) Merubah teks menjadi vector dan pembobotan kata dengan *tf-idf*.
e) Membagi data menjadi data *training* dan data *testing* .

3. Melakukan klasifikasi menggunakan SVM
   a) Menentukan pembobot parameter pada SVM tiap jenis kernel
   b) Membangun model SVM menggunakan fungsi kernel.
   c) Menghitung nilai akurasi dari model yang terbentuk.
4. Melakukan klasifikasi menggunakan KNN
   a) Menentukan nilai *k.*
   b) Menghitung kuadrat jarak *euclid(query instance)* masing-masing objek terhadap *training data* yang diberikan.
   c) Mengumpulkan label *class* Y (klasifikasi *Nearest Neighbours*).
5. Membandingkan performansi antara metode SVM dengan metode KNN berdasarkan tingkat akurasi ketepatan klasifikasi.

Berikut ini merupakan diagram alir dari langkah analisis.





Gambar 3.1 Diagram Alir

(Halaman Ini Sengaja Dikosongkan)

BAB IV
ANALISIS DAN PEMBAHASAN

Analisis dan pembahasan pada penelitian ini dilakukan *pre processing text* agar dapat dilakukan analisis menggunakan SVM dan KNN. Sebelum dilakukan analisis SVM dan KNN data akan dibagi menjadi *training* dan *testing*. Pada analisis SVM digunakan kernel linier dan polynomial. Untuk K-NN digunakan k 2,3 dan 5 untuk dilakukan analisis pada data *training*.

4.1 *Pre Processing Text*

Data artikel yang telah terkumpul akan dilakukan *pre processing text* yaitu *stemming, case folding, stopword* dan *tokenizing*. Proses *stemming* dilakukan dengan menggunakan program Sastrawi yang telah dimodifikasi. Tabel 4.1 merupakan contoh hasil proses *stemming* dari artikel yang telah diambil.

Tabel 4.1 Contoh Hasil Proses *Stemming*

Judul	Sebelum	Sesudah
Luhut: SP3 Bukan untuk Perlemah, Penggunaannya Diputuskan Pimpinan KPK	Salah satu poin dalam revisi UU No 30 Tahun 2003 tentang KPK yakni KPK diberikan kewenangan	salah satu poin dalam revisi uu no 30 tahun 2003 tentang kpk yakni kpk beri wenang
Serangan Udara AS di Libya Tewaskan 2 Staf Kedutaan Serbia	Sedikitnya 43 orang tewas dalam serangan udara yang dilancarkan Amerika Serikat terhadap kamp pelatihan ISIS di Libya. Dua staf Kedutaan Serbia yang diculik	sedikit 43 orang tewas dalam serang udara yang lancar amerika serikat hadap kamp latih isis di libya dua staf duta serbia yang culik

Pada Tabel 4.1 da pat diketahui bahwa proses *stemming* yang dilakukan dapat mengubah kata berimbuhan menjadi kata dasar. Dapat dilihat dari contoh hasil *stemming*

tersebut dari artikel 1 kata “kewenangan” berubah menjadi “wenang”. Selanjutnya pada artikel 2 kata “serangan” berubah menjadi “serang” dan kata “dilancarkan” berubah menjadi “lancar”. Setelah dilakukan proses *stemming* langkah selanjutnya adalah *stopword* dan *case folding*, yaitu mengubah semua karakter pada teks menjadi huruf kecil dan menghilangkan tanda baca titik (.), koma (,) dan angka. Hasil proses *stopword* dan *case folding* ditampilkan pada Tabel 4.2.

Tabel 4.2 Contoh Hasil Proses *Stopword* dan *Case Folding*

Judul	Sebelum	Sesudah
Luhut: SP3 Bukan untuk Perlemah, Penggunaannya Diputuskan Pimpinan KPK	salah satu poin dalam revisi uu no 30 tahun 2003 tentang kpk yakni kpk beri wenang	salah poin revisi uu no kpk kpk wenang
Serangan Udara AS di Libya Tewaskan 2 Staf Kedutaan Serbia	sedikit 43 orang tewas dalam serang udara yang lancar amerika serikat hadap kamp latih isis di libya dua staf duta serbia yang culik	orang tewas serang udara lancar amerika serikat hadap kamp latih isis libya staf duta serbia culik

Tabel 4.2 menunjukkan bahwa adanya perubahan isi artikel sebelum dilakukan proses *stopword* dan *case folding.* Proses *stopword* melakukan penghapusan beberapa kata, sebagai contoh pada artikel 1 terdapat penghapusan kata”satu”, “dalam”, “tentang”, “yakni” dan “beri”. Pada artikel 2 terdapat penghapusan kata “sedikit”, “dalam”, “yang”, dan “di”. Dalam proses *case folding* yaitu menghilangkan angka. Langkah selanjutnya yaitu proses memecah yang semula berupa kalimat menjadi kata (*word vector*). Proses ini merupakan *tokenizing.* Hasil dari proses tersebut dapat dilihat pada Tabel 4.3

Tabel 4.3 Contoh Hasil Proses *Tokenizing*

Kategori	Kata Kunci					
	rekor	**remaja**	**rencana**	**rendah**	**resmi**	**revisi**
News	0	0	0	0	0	1
News	0	0	0	0	1	0

Tabel 4.3 menunjukkan bahwa pada artikel 1 terdapat kata "revisi" yang muncul sebanyak 1 kali dan artikel 2 terdapat kata "resmi" muncul sebanyak 1 kali . Apabila terdapat lebih dari satu artikel maka kemungkinan kata yang terbentuk dapat lebih dari 8000 kata.

Berikut merupakan lima kata dengan frekuensi tertinggi untuk tiap kategori berita pada *word vector* yang berjumlah 1009.

Tabel 4.4 Frekuensi Kemunculan Kata Tertinggi Tiap Kategori

News	**Jumlah**	**Finance**	**Jumlah**	**Hot**	**Jumlah**
Ahok	136	Harga	155	Jakarta	77
Laku	128	Bangun	147	Orang	76
Jakarta	127	Turun	132	Sang	70
Orang	104	Perintah	122	Lapor	67
DKI	64	Indonesia	111	Jazz	66

Untuk kategori *news* kata yang paling sering muncul adalah "Ahok", lalu selanjutnya ada kata "laku","Jakarta", "orang" dan "DKI". Untuk kategori *finance* kata yang paling sering muncul adalah "harga","bangun", "turun", "perintah" dan "Indonesia". Dalam bahasan kategori *finance* lebih banyak membahas tentang harga kebutuhan pokok dan harga minyak dll. Untuk kategori *hot* kata yang sering muncul jumlah kata tidak lebih banyak dari kategori sebelumnya. Karena dalam kategori *hot* pembahasan yang paling banyak mengenai tentang artis, musik dan film.

Lanjutan Tabel 4.4 Frekuensi Kemunculan Kata Tertinggi Tiap Kategori

Sport	Jumlah	Oto	Jumlah
Balap	224	Mobil	329
Rio	192	Motor	191
Indonesia	142	Indonesia	181
Musim	136	Honda	112
Tim	132	Kendara	108

Kategori *sport* kata yang memiliki frekuensi paling banyak muncul adalah "balap","Rio", "Indonesia", "musim" dan "tim". Dan kategori *oto* kata yang memiliki frekuensi banyak muncul misalnya adalah "mobil", "motor", "Indonesia", "Honda" dan "kendara". Dari Tabel 4.6 k ata "Indonesia" adalah kata yang sering muncul dari beberapa kategori.

4.2 *Support Vector Machine* dalam *Text Mining*

Pada penelitian klasifikasi berita *online* digunakan metode *support vector machine*. Fungsi kernel yang akan digunakan adalah kernel linier dan polynomial. Berikut merupakan pembahasan dari kernel linier dan kernel polynomial pada data *training* dan data *testing*.

4.2.1 SVM Menggunakan Kernel Linier pada Data Training

Data *training* kelas pada artikel berita telah diketahui. Dimana tujuan data *training* adalah untuk menghasilkan ketepatan klasifikasi. Dalam penelitian ini akan dilakukan beberapa percobaan *word vector* untuk memeberikan hasil yang paling optimum. Tiap *word vector* dicari ketepatan klasifikasi yang paling baik dengan menggunakan kernel linier. Dengan menggunakan parameter *c* dengan rentang nilai 10^{-3} sampai 10^{3} didapatkan hasil seperti pada Tabel 4.5

Tabel 4.5 Ketepatan Klasifikasi SVM Kernel Linier pada Data *Training*

		Ketepatan Klasifikasi (%)						
	C	0.001	0.01	0.1	**1**	10	100	1000
Word Vector	1009	98.95	100	100	**100**	100	100	100
	1595	99.04	100	100	**100**	100	100	100
	2220	99.4	100	100	**100**	100	100	100
	2595	99.49	100	100	**100**	100	100	100
	3038	99.56	100	100	**100**	100	100	100
	3784	99.62	100	100	**100**	100	100	100

Berdasarkan Tabel 4.5 dapat diketahui bahwa dengan menggunakan kernel linier untuk setiap *word vector* dengan menggunakan *k-fold cross validation* sebesar 10 *fold* didapatkan nilai ketepatan paling besar 100% pada semua *word vector* dengan menggunakan c=0.01 sampai c=1000. Pada c=0.001 didapatkan hasil ketepatan klasifikasi yang berbeda-beda. Parameter c=1 akan digunakan pada data testing. Selanjutnya akan melihat pengukuran performansi pada data *training word vector* 3784 untuk tiap *fold*.

Dari hasil pengukuran performansi kernel linier tiap *fold* pada data *training* didapatkan hasil rata-rata dari 10 *fold* untuk akurasi total ,*recall* total, *precision* total dan *F-Measure* total pada tiap fold dalam data *training* menghasilkan nilai 100%. Selanjutnya hasil pengukuran performansi tiap kategori untuk data *training* dengan *word vector* 3784.

Hasil yang didapatkan untuk pengukuran performansi tiap kategori pada data *training* dengan menggunakan *word vector* 3784 menghasilkan rata-rata dari 10 *fold* untuk akurasi total, *recall* total, *precision* total, dan *F-Measure* total sebesar 100%. Selanjutnya akan dilakukan SVM dengan menggunakan kernel *polynomial* pada data *training*.

4.2.2 SVM Menggunakan Kernel Polynomial pada Data Training

Kernel polynomial merupakan pengembangan dari kernel linier dengan menambahkan parameter γ , r dan p. Selama percobaan didapatkan hasil yang paling baik adalah dengan parameter γ = 1 r= 6, dan p = 3. Didapatkan hasil seperti pada Tabel 4.6

Tabel 4.6 Ketepatan Klasifikasi SVM Kernel *Polynomial* pada Data *Training*

		Ketepatan Klasifikasi (%)						
	C	0.001	0.01	**0.1**	1	10	100	1000
Word Vector	1009	93.42	99.28	**100**	100	100	100	100
	1595	85.08	98.64	**100**	100	100	100	100
	2220	82.35	98.84	**100**	100	100	100	100
	2595	81.02	98.93	**100**	100	100	100	100
	3038	79.13	98.95	**100**	100	100	100	100
	3784	79.8	99.08	**100**	100	100	100	100

Tabel 4.6 menunjukkan bahwa pada c=0.1 sampai dengan c=1000 didapatkan hasil ketepatan klasifikasi sebesar 100%. Setelah mengetahui ketepatan klasifikasi pada data *training* dengan menggunakan kernel polynomial dilakukan pengukuran performansi pada *word vector* 3784. Selanjutnya parameter c=0.1 akan digunakan untuk melihat akurasi dari data *testing*.

Hasil pengukuran performansi untuk tiap *fold* pada data *training* dengan menggunakan *word vector* sebesar 3784 menghasilkan rata-rata dari 10 *fold* untuk akurasi total, *recall* total, *precision* total dan *F-Measure* total sebesar 100% Selanjutnya dilakukan pengukuran performansi tiap kategori.

Dari hasil yang didapat diperoleh dari data *training* untuk tiap kategori dengan rata-rata 10 *fold* menghasilkan 100% untuk pengukuran performansi pada *word vector* 3784.

Selanjutnya akan dilakukan ketepatan klasifikasi pada data *testing.*

4.2.3 SVM Menggunakan Kernel Linier pada Data *Testing*

Kelas pada artikel pada data *testing* berita belum diketahui sebelumnya. Kelas pada artikel berita akan diketahui dengan menggunakan model yang terbentuk dari data *training.* Berikut ini akan ditampilkan hasil klasifikasi berita dengan data *testing* menggunakan model yang telah terbentuk pada sebelumnya. Dengan menggunakan *word vector* sebesar 3784 dengan *c*=1 didapatkan hasil sebagai berikut.

Tabel 4.7 Performansi Kernel Linier tiap Fold pada Data *Testing*

Fold	Akurasi Total	*Recall*	*Precision*	*F-Measure*
Fold 1	94%	94%	94.85%	93.95%
Fold 2	94%	94%	94.18%	93.99%
Fold 3	94%	94%	94.36%	93.87%
Fold 4	94%	94%	94.67%	93.96%
Fold 5	94%	94%	94.67%	94.08%
Fold 6	92%	92%	93.03%	92.05%
Fold 7	92%	92%	92.00%	92.00%
Fold 8	90%	90%	90.14%	89.98%
Fold 9	92%	92%	91.86%	91.65%
Fold 10	94%	94%	94.36%	93.87%
Rata-rata	**93%**	**93%**	**93.41%**	**92.94%**

Berdasarkan Tabel 4.7 dapat diketahui bahwa hasil ketepatan klasifikasi dengan menggunakan kernel linier pada *word vector* 3784 didapatkan nilai rata-rata 10 *fold* untuk akurasi total, *recall*, *precision*, dan *F-Measure* sebesar 93%, 93%, 93.41% dan 92.94%. Didapatkan nilai akurasi yang paling tinggi adalah 94%. Diambil *fold* ke 10 untuk melihat performansi tiap kategori dan hasil dari *confusion matrix*.

Tabel 4.8 Performansi Kernel Linier tiap Kategori pada Data *Testing*

	Kategori	*Recall*	*Precision*	*F-Measure*
1	Finance	100%	90.9%	95.23%
2	Hot	100%	90.9%	95.23%
3	News	80%	100%	88.89%
4	Oto	90%	90%	90.00%
5	Sport	100%	100%	100%
	Rata-rata	**94%**	**94.36%**	**93.87%**

Tabel 4.8 menunjukkan bahwa hasil ketepatan klasifikasi dengan menggunakan kernel linier pada *word vector* 3784 didapatkan nilai rata-rata dari 10 *fold* untuk *recall*, *precision*, dan *F-Measure* sebesar 94%, 94.36% dan 93.87%. Kategori yang memiliki nilai akurasi sebesar 100% yaitu kategori *finance, hot* dan *sport.* Untuk kategori *news* dan *oto* memiliki nilai *recall* sebesar 80% dan 90%. Berikut merupakan hasil *confusion matrix* dari *fold* ke 10. *Confusion matrix* untuk tiap *fold* dapat dilihat pada Lampiran 2.

. **Tabel 4.9** *Confusion Matrix* Kernel Linier

Kelas Asli		**Kelas Prediksi**				
		a	b	c	d	e
a	News	**8**	0	1	0	1
b	Finance	0	**10**	0	0	0
c	Hot	0	0	**10**	0	0
d	Sport	0	0	0	**10**	0
e	Oto	0	1	0	0	**9**

Tabel 4.9 dapat diketahui hasil *confusion matrix* dari *fold* ke 10 bahwa kategori *news* jumlah artikel berita yang tepat diklasifikasikan benar pada kategori tersebut sebanyak 8 sedangkan diklasifikasikan kedalam kategori lainnya sebanyak 2 artikel. Untuk kategori *finance,hot* dan *sport* sudah terklasifikasikan dengan benar. Kategori *oto* artikel yang

diklasifikasikan benar kedalam kategori tersebut sebanyak 9 artikel sedangkan 1 artikel terklasifikasikan kedalam kategori yang lain.

4.2.4 SVM Menggunakan Kernel *Polynomial* pada Data *Testing*

Langkah yang akan dilakukan untuk data testing dengan menggunakan kernel *polynomial* sama seperti pada kernel linier. Kelas pada artikel berita akan diketahui dengan menggunakan model yang telah terbentuk dari data *training*.

Tabel 4.10 Performansi Kernel Polynomial tiap Fold pada Data *Testing*

Fold	Akurasi Total	*Recall*	*Precision*	*F-Measure*
Fold 1	94%	94%	94.84%	93.95%
Fold 2	96%	96%	96.36%	95.98%
Fold 3	94%	94%	94.36%	93.87%
Fold 4	94%	94%	94.66%	93.95%
Fold 5	94%	94%	94.66%	94.07%
Fold 6	92%	92%	93.03%	92.04%
Fold 7	92%	92%	92%	92%
Fold 8	90%	90%	90.14%	89.98%
Fold 9	92%	92%	91.86%	91.65%
Fold 10	94%	94%	94.36%	93.87%
Rata-rata	**93.2%**	**93.2%**	**93.63%**	**93.14%**

Berdasarkan Tabel 4.10 dapat diketahui bahwa hasil nilai rat-rata dari 10 *fold* didapatkan akurasi total, *recall*, *precision*, dan *F-Measure* sebesar 93.2%, 93.2%, 93.63% dan 93.14%. Selanjutnya dilihat hasil performansi tiap kategori dan hasil *confusion matrix* pada *fold* ke 2.

Tabel 4.11 Performansi Kernel *Polynomial* tiap Kategori pada Data *Testing*

Kategori	***Recall***	***Precision***	***F-Measure***
Finance	90%	100%	94.74%
Hot	100%	90.91%	95.24%
News	90%	100%	94.74%
Oto	100%	90.91%	95.24%
Sport	100%	100%	100%
Rata-rata	**96%**	**96.36%**	**95.99%**

Tabel 4.11 menunjukkan bahwa hasil ketepatan klasifikasi dengan menggunakan kernel *polynomial* pada *word vector* 3784 didapatkan nilai rata-rata dari 10 *fold recall*, *precision*, dan *F-Measure* sebesar 96%, 96.36% dan 95.99%. Kategori *hot, oto* dan *sport* memiliki nilai *recall* 100% . sedangkan kategori *finance,news* dan *sport* memiliki nilai *precision* sebesar 100%. Kategori *news* dan *finance* memiliki nilai *recall* sebesar 90%. Berikut merupakan *confusion matrix* dari *fold* ke 2 yang akan ditampilkan pada Tabel 4.12

Tabel 4.12 *Confusion Matrix* Kernel Polynomial

Kelas Asli		**Kelas Prediksi**				
		a	b	c	d	e
a	News	**9**	0	1	0	0
b	Finance	0	**9**	0	0	1
c	Hot	0	0	**10**	0	0
d	Sport	0	0	0	**10**	0
e	Oto	0	0	0	0	**10**

Tabel 4.12 menunjukkan bahwa kategori *hot,oto* dan *sport* tepat diklasifikasi kedalam kategori tersebut sedangkan kategori *news* dan *finance* dari 10 artikel terdapat pengklasifikasian ke dalam kategori lainnya sebanyak 1 artikel.

Tabel 4.13 Pengukuran Performansi SVM

	Akurasi Total	***Recall***	***Precision***	***F-Measure***
Linier	93%	93%	93.41%	92.94%
Polynomial	93.2%	93.2%	93.63%	93.14%

Tabel 4.13 merupakan hasil dari rata-rata tiap *fold* untuk tiap nilai dari akurasi total, *recall, precision* dan *F-Measure.* Dapat dilihat bahwa pada kernel tersebut memiliki nilai yang sama baiknya akan tetapi nilai akurasi kernel polynomial lebih tinggi dari pada linier. Untuk dibandingkan dengan KNN maka digunakan SVM dengan menggunakan kernel polynomial.

4.2.5 Model SVM *Multiclass* Menggunakan Kernel Polynomial Data Testing

Berdasarkan hasil pembahasan ketepatan klasifikasi kernel polynomial lebih baik dari pada kernel linier. Sehingga apabila menggunakan persamaan kernel polynomial $K(\mathbf{x_i}, \mathbf{x}) = (\gamma \mathbf{x_i^T x} + r)^p$.

Dimana $K(x_i, x) = (\gamma \phi(x_i)^T \phi(x) + r)^p$. Pada kasus SVM ini terdapat 5 kelas dengan tiap kelas memiliki jumlah *support vector* yang sebanyak 31. Selanjutnya adalah membentuk 10 persamaan biner SVM dengan metode *one against one*. Berikut merupakan beberapa persamaan SVM biner.

SVM Biner kategori 1 vs 2

$$f^{12}(x) = \sum_{i=1}^{62} \alpha_i y_i (1 \times \phi(\mathbf{x_i})^T \phi(\mathbf{x}) + 6)^3 - 0{,}06142692$$

SVM Biner kategori 1 vs 3

$$f^{13}(x) = \sum_{i=1}^{62} \alpha_i y_i (1 \times \phi(\mathbf{x_i})^T \phi(\mathbf{x}) + 6)^3 + 0.13187868$$

SVM Biner kategori 1 vs 4

$$f^{14}(x)=\sum_{i=1}^{62}\alpha_i y_i(1\times\phi(\mathbf{x_i})^T\phi(\mathbf{x})+6)^3-0.23493447$$

SVM Biner kategori 1 vs 5

$$f^{15}(x)=\sum_{i=1}^{62}\alpha_i y_i(1\times\phi(\mathbf{x_i})^T\phi(\mathbf{x})+6)^3-0.03214075$$

SVM Biner kategori 2 vs 3

$$f^{23}(x)=\sum_{i=1}^{62}\alpha_i y_i(1\times\phi(\mathbf{x_i})^T\phi(\mathbf{x})+6)^3+0.19821238$$

SVM Biner kategori 2 vs 4

$$f^{24}(x)=\sum_{i=1}^{62}\alpha_i y_i(1\times\phi(\mathbf{x_i})^T\phi(\mathbf{x})+6)^3+0.06061854$$

SVM Biner kategori 2 vs 5

$$f^{25}(x)=\sum_{i=1}^{62}\alpha_i y_i(1\times\phi(\mathbf{x_i})^T\phi(\mathbf{x})+6)^3-0.35047705$$

SVM Biner kategori 3 vs 4

$$f^{34}(x)=\sum_{i=1}^{62}\alpha_i y_i(1\times\phi(\mathbf{x_i})^T\phi(\mathbf{x})+6)^3-0.35047705$$

SVM Biner kategori 3 vs 5

$$f^{35}(x)=\sum_{i=1}^{62}\alpha_i y_i(1\times\phi(\mathbf{x_i})^T\phi(\mathbf{x})+6)^3-0.15152921$$

SVM Biner kategori 4 vs 5

$$f^{45}(x)=\sum_{i=1}^{62}\alpha_i y_i(1\times\phi(\mathbf{x_i})^T\phi(\mathbf{x})+6)^3+0.21545623$$

Untuk persamaan ke 1 jika didapatkan nilai $f(x)\geq 0$ maka artikel data tersebut akan dikategorikan kedalam kategori 1, namun jika $f(x)\leq 0$ maka artikel tersebut dikategorikan kedalam kategori 2.

4.3 K-*Nearest Neighbor* dalam *Text Mining*

KNN merupakan salah satu pendekatan yang sederhana untuk diimplementasikan dan merupakan metode lama yang digunakan dalam pengklasifikasian. K-NN memiliki tingkat efisiensi yang tinggi dan dalam beberapa kasus memberikan tingkat akurasi yang tinggi dalam hal pengklasifikasian. Prinsip kerja K-*Nearest Neighbor* (K-NN) adalah melakukan klasifikasi berdasarkan kedekatan lokasi (jarak) suatu data dengan data yang lain. Dekat atau jauhnya lokasi (jarak) bisa dihitung melalui salah satu dari besaran jarak yang telah ditentukan yakni jarak *Euclidean*

4.3.1 *K-NN* pada Data Training

KNN untuk data *training* kelas pada artikel berita telah diketahui dengan tujuan menggunakan data *training* untuk menghasilkan ketepatan klasifikasi yang akan digunakan untuk *data testing* .

Tabel 4.14 Ketepatan Klasifikasi KNN pada Data *Training*

	Akurasi Total	***Recall***	***Precision***	***F-Measure***
2-NN	83.97%	83.97%	90.27%	87.00%
3-NN	75.60%	75.60%	87.48%	81.80%
5-NN	68.86%	68.86%	85.97%	76.43%

Tabel 4.14 dapat diketahui bahwa tingkat akurasi yang tertinggi dengan menggunakan k=2. Didapatkan hasil nilai akurasi,*recall, precision* dan *F-Measure* masing-masing sebesar 83.97%, 83.97%, 90.27% dan 87%. Semakin besar k yang digunakan akan menghasilkan nilai akurasi semakin kecil.

Berikut merupakan hasil pengukuran performansi tiap kategori yang ditampilkan pada Tabel 4.19

Tabel 4.15 Performansi KNN tiap Kategori pada Data *Training*

Kategori	***Recall***	***Precision***	***F-Measure***
Finance	83.33%	97.40%	89.83%
Hot	82.22%	92.50%	87.06%

Tabel 4.15 (Lanjutan)

Kategori	*Recall*	*Precision*	*F-Measure*
News	83.33%	94.93%	88.75%
Oto	96.66%	65.90%	78.37%
Sport	90%	98.78%	94.18%
Rata-rata	**87.11%**	**89.91%**	**87.64%**

Tabel 4.15 menunjukkan bahwa hasil ketepatan klasifikasi tiap kategori didapatkan nilai dari *recall*, *precision* , dan *F-Measure* sebesar 87.11%, 89.91% dan 87.64%. Kategori *oto* memiliki nilai *recall* yang paling tinggi yaitu 96.66% . sedangkan kategori *hot* memiliki nilai *recall* yang rendah yaitu sebesar 82.22%. Berikut merupakan hasil dari pengukuran performansi tiap *fold* yang ditampilkan pada Tabel 4.20

Tabel 4.16 Performansi KNN tiap Fold pada Data *Training*

Fold	**Akurasi Total**	***Recall***	***Precision***	***F-Measure***
Fold 1	82.67%	82.67%	90.19%	84.00%
Fold 2	83.78%	83.78%	90.65%	85.07%
Fold 3	82.00%	82.00%	89.71%	83.50%
Fold 4	84.67%	84.67%	90.63%	85.68%
Fold 5	86.67%	86.67%	91.20%	87.48%
Fold 6	82.00%	82.00%	89.83%	83.40%
Fold 7	82.44%	82.44%	90.15%	83.94%
Fold 8	84.89%	84.89%	90.78%	85.95%
Fold 9	87.11%	87.11%	89.91%	87.64%
Fold 10	83.56%	83.56%	89.67%	84.67%
Rata-rata	**83.98**%	**83.98**%	**90.27**%	**85.13**%

Tabel 4.16 dengan menggunakan data *training* pada *word vector* 3784 dengan k=2 didapatkan hasil nilai rata-rata dari 10 *fold* untuk akurasi total, *recall* total, *precision* total,dan *F-Measure* total yaitu sebesar 83.98%, 83.98%, 90.27% dan 85.13%.

4.3.2 *K-NN* pada Data Testing

Langkah yang digunakan untuk data *testing* sama dengan langkah yang digunakan pada SVM data *testing* yaitu kelas pada artikel berita belum diketahui dan dengan menggunakan model KNN yang terbentuk pada *training* dan akan memprediksi artikel tersebut masuk ke dalam kategori berita tersebut. Berikut merupakan hasil dari pengukuran performansi menggunakan KNN.

Tabel 4.17 Performansi KNN tiap Fold pada Data *Testing*

Fold	**Akurasi Total**	***Recall***	***Precision***	***F-Measure***
Fold 1	54%	54%	75.90%	51.16%
Fold 2	64%	64%	83.25%	65.18%
Fold 3	56%	56%	86.25%	56.53%
Fold 4	68%	68%	87.69%	70.12%
Fold 5	58%	58%	74.61%	54.99%
Fold 6	60%	60%	83.56%	62.08%
Fold 7	52%	52%	72.92%	51.64%
Fold 8	64%	64%	81.98%	65.05%
Fold 9	66%	66%	78.89%	64.49%
Fold 10	58%	58%	86.45%	60.22%
Rata-rata	**60%**	**60%**	**81.15%**	**60.15%**

Tabel 4.17 dengan menggunakan *word vector* 3784 dengan k=2 didapatkan hasil nilai rata-rata dari 10 fold untu akurasi total, *recall, precision,* dan *F-Measure* yaitu sebesar 60%, 60%, 81.15% dan 60.15%. *Fold* ke 8 a kan digunakan untuk melihat performansi tiap kategori dan *confusion matrix*. Berikut merupakan hasil dari performansi tiap kategori yang ditampilkan pada Tabel 4.18

Tabel 4.18 Performansi KNN tiap Kategori pada Data *Testing*

Kategori	***Recall***	***Precision***	***F-Measure***
Finance	70%	100%	82.35%
Hot	100%	38.46%	55.55%
News	40%	100%	57.14%
Oto	50%	100%	66.67%
Sport	80%	100%	88.89%
Rata-rata	**68%**	**87.69%**	**70.12%**

Tabel 4.18 dapat diketahui bahwa dengan *word vector* 3784 didapatkan hasil dari nilai *recall, precision,* dan *F-Measure* sebesar 6 8%, 87.69% dan 70.12%. Kategori *hot* memiliki nilai *recall* sebesar 100% sedangkan kategori *news* memiliki nilai *recall* yang paling rendah yaitu sebesar 40%. Pada kategori *hot* memiliki nilai *precision* paling rendah yaitu 38.46%. Berikut merupakan hasil *confusion matrix* dari *fold* ke 8 pada Tabel 4.19

Tabel 4.19 *Confusion Matrix* dengan Metode KNN

Kelas Asli		Kelas Prediksi				
		a	b	c	d	e
a	News	**4**	0	6	0	0
b	Finance	0	**7**	3	0	0
c	Hot	0	0	**10**	0	0
d	Sport	0	0	2	**8**	0
e	Oto	0	0	5	0	**5**

Tabel 4.19 menunjukkan bahwa kategori *hot* dari 10 artikel semuanya diklasifikasikan dengan benar. Kategori *news* merupakan kategori yang paling sedikit diklasifikasikan dengan tepat. Terdapat 4 artikel yg d iklasifikasikan dengan benar sisanya diprediksi masuk kedalam kategori *hot* sebanyak 6 artikel . Kategori *finance* dari 10 artikel terdapat 7 artikel yang diklasifikasikan kedalam kategori tersebut sebanyak 7 artikel sedangkan 3 artikel masuk diklasifikasikan

kedalam kategori lain. Kategori *sport* terdapat 8 artikel yang diklasifikasikan dengan benar sedangkan 2 artikel masuk kedalam kategori lain.

4.4 Perbandingan Antara SVM dan KNN

Setelah didapatkan hasil ketepatan klasifikasi pada kedua metode maka langkah selanjutnya adalah membandingkan. Berikut merupakan perbandingan antara kedua metode berdasarkan akurasi total, *precision, recall,* dan *F-Measure*.

Tabel 4.20 Perbandingan Hasil Ketepatan Klasifikasi Antara SVM dan KNN

Metode	Akurasi Total	Recall	Precision	F-Measure
SVM	93.2%	93.2%	93.63%	93.14%
KNN	60%	60%	81.15%	68.90%

Tabel 4.20 dapat dilihat bahwa dari hasil pengukuran performansi yang dilihat dari akurasi total, *precision, recall,* dan *F-Measure* SVM kernel *polynomial* lebih baik dari pada KNN. Hasil dari KNN memberikan tingkat akurasi lebih kecil dibandingkan dengan metode SVM.

(halaman ini sengaja dikosongkan)

Lampiran 1. Judul dan Kategori Artikel

No	Kategori	Judul
1	News	Masinton Mengaku Sudah Damai dengan Dita, MKD: Laporan Belum Dicabut
2	News	Jessica, Hasil Tes Psikiatri, dan Gugatan Praperadilan
⋮	⋮	⋮
100	News	Demo Bawa Ambulans ke Istana, Ratusan Dokter Menyoal BPJS
101	Finance	Ayo Curi Ilmu Bos Sido Muncul, Salah Satu Orang Terkaya RI
103	Finance	Rawan Dipakai Teroris, Eropa Mau Hapus Uang Pecahan 500 Euro
⋮	⋮	⋮
200	Finance	Ini Alasan Pemerintah Tak Lanjutkan Proyek Jembatan Selat Sunda
201	Hot	'Darah' Jazz Joey Alexander Mengalir dari Orangtua
202	Hot	Puisi J.R.R Tolkien yang Hilang 43 Tahun Lalu Ditemukan
⋮	⋮	⋮
300	Hot	Korban Ngaku Menyesal Jadi Penggemar Saipul Jamil
301	Sport	Jelang Tes di Phillip Island, Lorenzo Antusias dan Penasaran
302	Sport	SBY Siap Sambut Kehadiran Atlet Karate Internasional
303	Sport	Misi Tim Thomas Indonesia Kalahkan Taiwan dan Jadi Juara Grup
⋮	⋮	⋮
400	Sport	Liga Champions adalah Mimpi, Bukan Obsesi bagi Juventus dan Allegri
401	Oto	Ini Strategi Wuling Hadapi Persaingan di Indonesia
402	Oto	Dikabarkan Meluncur 2016, AHM: CBR250RR dalam Pengembangan
⋮	⋮	⋮
500	Oto	Yamaha Xabre Sudah Ada di Semua Daerah Indonesia

Lampiran 2. *Confusion Matrix* Kernel Linier tiap fold

			Fits						Fits		
	a	b	c	d	e		a	b	c	d	e
a	10	0	0	0	0	a	10	0	0	0	0
b	0	10	0	0	0	b	0	10	0	0	0
c	1	0	9	0	0	c	0	1	9	0	0
d	0	0	0	10	0	d	0	0	2	8	0
e	0	2	0	0	8	e	0	1	0	0	9
			Fits						Fits		
	a	b	c	d	e		a	b	c	d	e
a	9	0	0	1	0	a	8	0	1	1	0
b	0	9	1	0	0	b	0	10	0	0	0
c	0	1	9	0	0	c	1	0	9	0	0
d	0	0	0	10	0	d	1	0	0	9	0
e	0	0	0	0	10	e	0	0	0	0	10
			Fits						Fits		
	a	b	c	d	e		a	b	c	d	e
a	10	0	0	0	0	a	9	0	0	1	0
b	0	10	0	0	0	b	0	10	0	0	0
c	1	0	8	1	0	c	2	0	8	0	0
d	0	1	0	9	0	d	0	0	0	9	1
e	0	0	0	0	10	e	0	0	1	0	9
			Fits						Fits		
	a	b	c	d	e		a	b	c	d	e
a	8	0	1	1	0	a	10	0	0	0	0
b	0	10	0	0	0	b	0	10	0	0	0
c	0	0	10	0	0	c	0	1	7	1	1
d	0	0	1	9	0	d	0	0	0	10	0
e	0	0	0	0	10	e	0	0	1	0	9

Lampiran 2. (Lanjutan)

	Fits						Fits				
	a	b	c	d	e		a	b	c	d	e
a	9	0	1	0	0	a	10	0	0	0	0
b	0	10	0	0	0	b	0	10	0	0	0
c	0	0	10	0	0	c	0	1	8	1	0
d	1	0	0	9	0	d	1	0	0	9	0
e	0	0	1	0	9	e	0	0	0	0	10

a : Finance
b : Hot
c : News
d : Oto
e : Sport

Lampiran 3. *Confusion Matrix* Kernel Polynomial tiap fold

	Fits						Fits				
	a	b	c	d	e		a	b	c	d	e
a	10	0	0	0	0	a	10	0	0	0	0
b	0	10	0	0	0	b	0	10	0	0	0
c	1	0	9	0	0	c	0	1	9	0	0
d	0	0	0	10	0	d	0	0	2	8	0
e	0	2	0	0	8	e	0	1	0	0	9
	Fits						Fits				
	a	b	c	d	e		a	b	c	d	e
a	9	0	0	1	0	a	8	0	1	1	0
b	0	10	0	0	0	b	0	10	0	0	0
c	0	1	9	0	0	c	1	0	9	0	0
d	0	0	0	10	0	d	1	0	0	9	0
e	0	0	0	0	10	e	0	0	0	0	10
	Fits						Fits				
	a	b	c	d	e		a	b	c	d	e
a	10	0	0	0	0	a	9	1	0	0	0
b	0	10	0	0	0	b	0	10	0	0	0
c	1	0	8	1	0	c	2	0	8	0	0
d	0	1	0	9	0	d	0	0	0	9	1
e	0	0	0	0	10	e	0	0	1	0	9
	Fits						Fits				
	a	b	c	d	e		a	b	c	d	e
a	8	0	1	1	0	a	10	0	0	0	0
b	0	10	0	0	0	b	0	10	0	0	0
c	0	0	10	0	0	c	0	1	7	1	1
d	0	0	1	9	0	d	0	0	0	10	0
e	0	0	0	0	10	e	0	0	1	0	9

Lampiran 3. (Lanjutan)

	Fits						Fits				
	a	b	c	d	e		a	b	c	d	e
a	9	0	1	0	0	a	10	0	0	0	0
b	0	10	0	0	0	b	0	10	0	0	0
c	0	0	10	0	0	c	0	1	8	1	0
d	1	0	0	9	0	d	1	0	0	9	0
e	0	0	1	0	9	e	0	0	0	0	10

a : Finance
b : Hot
c : News
d : Oto
e : Sport

Lampiran 4. Akurasi Data Testing Kernel Linier

	Akurasi	*Recall*	*Precision*
0.01	93	93	93.41263
0.1	93	93	93.41263
1	93	93	93.41263
10	93	93	93.41263
100	93	93	93.41263
1000	93	93	93.41263

Lampiran 5. Akurasi Data Testing Kernel Polynomial

C	Gamma = 1	*r* = 6	*p* = 3
	Akurasi	*Recall*	*Precision*
0.1	93.2	93.2	93.63081
1	92.8	92.8	93.3026
10	92.8	92.8	93.3026
100	92.8	92.8	93.3026
1000	92.8	92.8	93.3026

r = 6		*c* = 0.1	
Gamma	*p* = 2	*p* = 3	*p* = 4
1	93.2	93.2	93.2
2	93.2	93.2	93.2
3	93.2	93.2	93.2

Lampiran 6. Prediksi Artikel Berita metode SVM dengan Kernel Linier dan Polynomial

Judul	Aktual	Linier	Polynom
Pedangdut Hesty 'Klepek-klepek' Ditangkap Polisi	News	Hot	News
Ahok: Pasukan Oranye itu Manajernya Lurah, Apa pun Dikerjain!	News	News	News
Luhut: SP3 Bukan untuk Perlemah, Penggunaannya Diputuskan Pimpinan KPK	News	News	News
Plastik Kulit Udang Ramah Lingkungan Buatan Siswi SMA	News	News	News
Sidang Praperadilan, Polda Tangkis Perlawanan Tim Jessica dengan Mudah	News	News	News
Belasan Motor yang Dimodifikasi hingga Bentuknya 'Aneh' Dimusnahkan Polisi	News	Oto	Hot
Bantu Kaum Miskin, Ridwan Kamil Segera Luncurkan Family Help Family	News	News	News
Penahanan Daeng Aziz Soal Pencurian Listrik Diputuskan Usai Polisi Gelar Perkara	News	News	News
Diduga Korupsi, Mantan Kekasih Presiden Bolivia Ditahan	News	News	News
Brigadir Petrus yang Mutilasi 2 Anaknya Tetap Diproses, Keputusan Diserahkan ke Hakim	News	News	News
Ayo Curi Ilmu Bos Sido Muncul, Salah Satu Orang Terkaya RI	Finance	Finance	Finance
Kesepakatan Arab Cs Tak Kuat, Harga Minyak Jatuh Hampir 4%	Finance	Finance	Finance
Harga Rumah di RI Melambung Tinggi, Ini Respons Pemerintah	Finance	Finance	Finance
Ada Kereta Cepat, Waktu Tempuh JKT-BDG Setara Naik Pesawat	Finance	Finance	Oto
BI Rate Turun, Ekonomi RI Diproyeksi Tumbuh 5,4%	Finance	Finance	Finance
Dirut BEI: OJK Tak Akan Keluarkan Aturan Pembatasan Margin Bank	Finance	Finance	Finance
Kupon Obligasi Indonesia Eximbank Dipatok 8,5-9,6%	Finance	Finance	Finance
Darmin Harap Bunga Kredit Bisa Turun Jadi 9% di Akhir 2016	Finance	Finance	Finance

Lampiran 6. (Lanjutan)

Pantau Ketat Wajib Pajak, Ditjen Pajak Siapkan 'Peta Khusus'	Finance	Finance	Finance
Tol Sumatera Akan Dorong Pertumbuhan Ekonomi Daerah Sampai 15%	Finance	Finance	Finance
Puisi J.R.R Tolkien yang Hilang 43 Tahun Lalu Ditemukan	Hot	Hot	Hot
Ditangkap di Hotel, Hesty Klepek-klepek Sudah Cuti Nyanyi Sejak Awal 2016	Hot	Hot	Hot
Lihat Transformasi Wajah CL '2NE1' Selama 17 Tahun dalam 60 Detik!	Hot	Hot	Hot
Sahrul Gunawan dan Istri Ungkap Alasan Bercerai	Hot	Hot	Hot
Fans Ngeluh Kepanasan Menunggu Nomor Antrean Masuk Konser EXO	Hot	Hot	Hot
Sofia Vergara Ungkap Kebahagiaan Pasca Nikah di Karpet Merah Oscar	Hot	Hot	Hot
Menantikan Penampilan Kocak Onew 'SHINee' di 'Descendants of the Sun'	Hot	Hot	Hot
Kehidupan Pribadi JMono di 'Berwarna' Neurotic	Hot	Hot	Hot
Bens Leo: Ireng Maulana Pahlawan Musisi Jazz	Hot	Hot	Hot
Korban Ngaku Menyesal Jadi Penggemar Saipul Jamil	Hot	Hot	Hot
Di Phillip Island, Rossi Punya Tamu Istimewa untuk Beri Saran	Sport	Sport	Sport
Antara Ayah dan Schumacher, Idola Rio Haryanto	Sport	Sport	Sport
Ananda Mikola: Rio Layak Tampil di F1	Sport	Sport	Sport
Raikkonen: Mercedes Tak Akan Dominan Lagi	Sport	Sport	Sport
Selesai Bersama Liverpool, Garuda Diajak Sponsori Rio di F1	Sport	Sport	Sport
Lewis Hamilton Sebut MotoGP Keren Sekali	Sport	Sport	Sport
Marquez Masih Kesulitan di Tikungan	Sport	Sport	Sport
Tim Putri PGN Popsivo Menang Mudah atas Bekasi BVN	Sport	Sport	Sport
Finis Disebut Jadi Target Awal Rio pada Balapan Perdana	Sport	Sport	Sport
Target Juara untuk Hendra/Ahsan	Sport	Sport	Sport
Toyota Indonesia Luncurkan Rush Berkapasitas 7 Penumpang	Oto	Oto	Oto

Lampiran 6. (Lanjutan)

Diperkenalkan Medio 2015, Pemesanan VW Polo 1.2 TSI Terus Mengalir	Oto	Oto	Oto
Kaderisasi Klub Ertiga Lewat Berkemah	Oto	Oto	Oto
Produsen Ban China Segera Gulirkan Ban di Cikampek	Oto	Finance	Oto
Mitsubishi Gelontorkan 10 Truk Baru Tahun Ini	Oto	Oto	Oto
Selera Konsumen Rolls-Royce Indonesia Lebih Tinggi dari Singapura	Oto	Oto	Oto
Wah, Orang Kaya Singapura Pamer Ferrari, Lamborghini, Pagani di Depan Rumah	Oto	Oto	Oto
Berada di Naungan Polri, FKPM Motor Besar Indonesia Janji Tak Arogan	Oto	Oto	Oto
Pengguna Moge Ingin Bentuk Pandangan Positif di Mata Masyarakat	Oto	Oto	Oto
Begini Cara Mudah Mengemudikan Ferrari California T	Oto	Oto	Oto

Lampiran 7. *Confusion Matrix* Metode KNN

	Predict						Predict					
	a	b	c	d	e		a	b	c	d	e
a	2	7	0	1	0	a	4	6	0	0	0
b	0	9	0	1	0	b	0	10	0	0	0
c	1	8	1	0	0	c	0	6	4	0	0
d	0	1	0	9	0	d	0	5	0	5	0
e	0	4	0	0	6	e	0	2	0	1	7
	Predict						Predict					
	a	b	c	d	e		a	b	c	d	e
a	4	5	0	1	0	a	3	7	0	0	0
b	0	10	0	0	0	b	0	10	0	0	0
c	0	6	4	0	0	c	0	5	5	0	0
d	0	3	0	7	0	d	0	9	0	1	0
e	0	2	0	1	7	e	0	1	0	2	7
	Predict						Predict					
	a	b	c	d	e		a	b	c	d	e
a	2	8	0	0	0	a	7	3	0	0	0
b	0	10	0	0	0	b	0	10	0	0	0
c	0	5	5	0	0	c	0	6	3	1	0
d	0	7	0	3	0	d	0	5	0	5	0
e	0	2	0	0	8	e	0	2	0	1	7
	Predict						Predict					
	a	b	c	d	e		a	b	c	d	e
a	7	3	0	0	0	a	9	0	0	1	0
b	0	10	0	0	0	b	0	9	0	1	0
c	0	6	4	0	0	c	1	3	4	2	0
d	0	5	0	5	0	d	0	2	0	8	0
e	0	2	0	0	8	e	0	1	0	6	3

Lampiran 7. (Lanjutan)

	Predict						Predict				
	a	b	c	d	e		a	b	c	d	e
a	2	8	0	0	0	a	7	3	0	0	0
b	0	9	1	0	0	b	0	10	0	0	0
c	0	9	1	0	0	c	0	7	3	0	0
d	0	0	0	10	0	d	0	6	0	4	0
e	0	2	0	1	7	e	0	5	0	0	5

Lampiran 8. Prediksi Artikel Berita metode KNN

Judul	Aktual	Prediksi
Pedangdut Hesty 'Klepek-klepek' Ditangkap Polisi	News	Hot
Ahok: Pasukan Oranye itu Manajernya Lurah, Apa pun Dikerjain!	News	Hot
Luhut: SP3 Bukan untuk Perlemah, Penggunaannya Diputuskan Pimpinan KPK	News	News
Plastik Kulit Udang Ramah Lingkungan Buatan Siswi SMA	News	Oto
Sidang Praperadilan, Polda Tangkis Perlawanan Tim Jessica dengan Mudah	News	Hot
Belasan Motor yang Dimodifikasi hingga Bentuknya 'Aneh' Dimusnahkan Polisi	News	Finance
Bantu Kaum Miskin, Ridwan Kamil Segera Luncurkan Family Help Family	News	News
Penahanan Daeng Aziz Soal Pencurian Listrik Diputuskan Usai Polisi Gelar Perkara	News	News
Diduga Korupsi, Mantan Kekasih Presiden Bolivia Ditahan	News	News
Brigadir Petrus yang Mutilasi 2 Anaknya Tetap Diproses, Keputusan Diserahkan ke Hakim	News	News
Ayo Curi Ilmu Bos Sido Muncul, Salah Satu Orang Terkaya RI	Finance	Finance
Kesepakatan Arab Cs Tak Kuat, Harga Minyak Jatuh Hampir 4%	Finance	Finance
Harga Rumah di RI Melambung Tinggi, Ini Respons Pemerintah	Finance	Hot
Ada Kereta Cepat, Waktu Tempuh JKT-BDG Setara Naik Pesawat	Finance	Hot
BI Rate Turun, Ekonomi RI Diproyeksi Tumbuh 5,4%	Finance	Finance

Lampiran 8. (Lanjutan)

Dirut BEI: OJK Tak Akan Keluarkan Aturan Pembatasan Margin Bank	Finance	Hot
Kupon Obligasi Indonesia Eximbank Dipatok 8,5-9,6%	Finance	Finance
Darmin Harap Bunga Kredit Bisa Turun Jadi 9% di Akhir 2016	Finance	Finance
Pantau Ketat Wajib Pajak, Ditjen Pajak Siapkan 'Peta Khusus'	Finance	Finance
Tol Sumatera Akan Dorong Pertumbuhan Ekonomi Daerah Sampai 15%	Finance	Finance
Puisi J.R.R Tolkien yang Hilang 43 Tahun Lalu Ditemukan	Hot	Hot
Ditangkap di Hotel, Hesty Klepek-klepek Sudah Cuti Nyanyi Sejak Awal 2016	Hot	Hot
Lihat Transformasi Wajah CL '2NE1' Selama 17 Tahun dalam 60 Detik!	Hot	News
Sahrul Gunawan dan Istri Ungkap Alasan Bercerai	Hot	News
Fans Ngeluh Kepanasan Menunggu Nomor Antrean Masuk Konser EXO	Hot	Hot
Sofia Vergara Ungkap Kebahagiaan Pasca Nikah di Karpet Merah Oscar	Hot	News
Menantikan Penampilan Kocak Onew 'SHINee' di 'Descendants of the Sun'	Hot	News
Kehidupan Pribadi JMono di 'Berwarna' Neurotic	Hot	Hot
Bens Leo: Ireng Maulana Pahlawan Musisi Jazz	Hot	News
Korban Ngaku Menyesal Jadi Penggemar Saipul Jamil	Hot	Hot
Di Phillip Island, Rossi Punya Tamu Istimewa untuk Beri Saran	Sport	Sport
Antara Ayah dan Schumacher, Idola Rio Haryanto	Sport	Sport

Lampiran 8. (Lanjutan)

Ananda Mikola: Rio Layak Tampil di F1	Sport	Sport
Raikkonen: Mercedes Tak Akan Dominan Lagi	Sport	Sport
Selesai Bersama Liverpool, Garuda Diajak Sponsori Rio di F1	Sport	Sport
Lewis Hamilton Sebut MotoGP Keren Sekali	Sport	News
Marquez Masih Kesulitan di Tikungan	Sport	News
Tim Putri PGN Popsivo Menang Mudah atas Bekasi BVN	Sport	Hot
Finis Disebut Jadi Target Awal Rio pada Balapan Perdana	Sport	Hot
Target Juara untuk Hendra/Ahsan	Sport	Sport
Toyota Indonesia Luncurkan Rush Berkapasitas 7 Penumpang	Oto	Oto
Diperkenalkan Medio 2015, Pemesanan VW Polo 1.2 TSI Terus Mengalir	Oto	News
Kaderisasi Klub Ertiga Lewat Berkemah	Oto	Oto
Produsen Ban China Segera Gulirkan Ban di Cikampek	Oto	News
Mitsubishi Gelontorkan 10 Truk Baru Tahun Ini	Oto	News
Selera Konsumen Rolls-Royce Indonesia Lebih Tinggi dari Singapura	Oto	Oto
Wah, Orang Kaya Singapura Pamer Ferrari, Lamborghini, Pagani di Depan Rumah	Oto	News
Berada di Naungan Polri, FKPM Motor Besar Indonesia Janji Tak Arogan	Oto	Oto
Pengguna Moge Ingin Bentuk Pandangan Positif di Mata Masyarakat	Oto	Oto
Begini Cara Mudah Mengemudikan Ferrari California T	Oto	News

Lampiran 9. *Syntax K-Fold Cross Validation*

```
k = 10
k1 = floor(n1/k)
k2 = floor(n2/k)
k3 = floor(n3/k)
k4 = floor(n4/k)
k5 = floor(n5/k)

AkurasiTrain = data.frame(matrix(ncol = 4, nrow = (k+1)))
AkurasiTest = data.frame(matrix(ncol = 4, nrow = (k+1)))
for (i in 1:k)
{
   if (i==k)
  {
   sam1 = Y1[((i-1)*k1+1):n1]
    sam2 = Y2[((i-1)*k2+1):n2]
    sam3 = Y3[((i-1)*k3+1):n3]
    sam4 = Y4[((i-1)*k4+1):n4]
    sam5 = Y5[((i-1)*k5+1):n5]
  }else
  {
    sam1 = Y1[((i-1)*k1+1):(i*k1)]
    sam2 = Y2[((i-1)*k2+1):(i*k2)]
    sam3 = Y3[((i-1)*k3+1):(i*k3)]
    sam4 = Y4[((i-1)*k4+1):(i*k4)]
    sam5 = Y5[((i-1)*k5+1):(i*k5)]
  }
  DataTrain = Data[-c(sam1, sam2, sam3, sam4, sam5),]
  DataTest = Data[c(sam1, sam2, sam3, sam4, sam5),]
```

Lampiran 10. *Syntax Stopword* dan *Case Folding*

a. *Function Stopword* dan *Case Folding*

```
SnCF=function(data,stoplist)
{
 artc=as.character(data)

 #Stopwords
  stplst=as.character(stoplist)
  stpwrd=removeWords(artc,stplst)
  write.csv(stpwrd,"D://Stopword.csv")

 #CaseFolding
  artc=as.character(stpwrd)
  artc=removePunctuation(artc)
  artc=removeNumbers(artc)
  artc=tolower(artc)
  write.csv(artc,"D://CaseFolding.csv")
}
```

b. *Running Stopword* dan *Case Folding*

```
library(NLP)
library(tm)
setwd("D://RunData")
data=scan("D://RunData/Results.csv",what="character(0)",se
p="\n",encoding="UTF-8")
stoplist=scan("D://RunData/Stoplist.csv",what="character(0)"
,sep="\n",encoding="UTF-8")
source("D://RunProgram//StopwordsCaseFolding.txt")
SnCF(data,stoplist)
```

Lampiran 11. *Syntax Stemming* Sastrawi yang telah Dimodifikasi

```
<?php
require_once __DIR__ . '/vendor/autoload.php';

include 'PHPExcel/IOFactory.php';
$inputFileName = 'artikel.xls';
$stemmerFactory = new \Sastrawi\Stemmer\StemmerFactory();
$stemmer  = $stemmerFactory->createStemmer();
//  Read your Excel workbook
try {
    $inputFileType =
PHPExcel_IOFactory::identify($inputFileName);
    $objReader =
PHPExcel_IOFactory::createReader($inputFileType);
    // $objReader->setInputEncoding('ISO-8859-1');
    $objPHPExcel = $objReader->load($inputFileName);
} catch(Exception $e) {
    die('Error loading file
"'.pathinfo($inputFileName,PATHINFO_BASENAME).'": '.$e-
>getMessage());
}

//  Get worksheet dimensions
$sheet = $objPHPExcel->getSheet(0);
$highestRow = $sheet->getHighestRow();
$highestColumn = $sheet->getHighestColumn();
$sentence = array();
$output = array();
```

Lampiran 11.(Lanjutan)

```
for($i=1; $i<=$highestRow+1; $i++)
{
        $sentence[$i] = $objPHPExcel->getActiveSheet()-
>getCell('A'.$i)->getValue();
        $output[$i]   = $stemmer->stem($sentence[$i]);
        // echo $output[$i];
        // echo "<br/>";
}

//start while loop to get data
for($i=1;$i<=$highestRow+1;$i++)
{
   $objPHPExcel->getActiveSheet()->setCellValue('A'.$i,
$output[$i]);
}
// Redirect output to a client's web browser (Excel5)
header('Content-Type: application/vnd.ms-excel');
header('Content-Disposition:
attachment;filename="results.csv"');
header('Cache-Control: max-age=0');
$objWriter =
PHPExcel_IOFactory::createWriter($objPHPExcel, 'CSV');
$objWriter->save('php://output');
```

Lampiran 12. *Syntax K-Nearest Neighbor*

```
function (train, test, cl, k = 1, l = 0, prob = FALSE, use.all = TRUE)
{
    train <- as.matrix(train)
    if (is.null(dim(test)))
        dim(test) <- c(1, length(test))
    test <- as.matrix(test)
    if (any(is.na(train)) || any(is.na(test)) || any(is.na(cl)))
        stop("no missing values are allowed")
    p <- ncol(train)
    ntr <- nrow(train)
    if (length(cl) != ntr)
        stop("'train' and 'class' have different lengths")
    if (ntr < k) {
        warning(gettextf("k = %d exceeds number %d of patterns",
            k, ntr), domain = NA)
        k <- ntr
    }
    if (k < 1)
        stop(gettextf("k = %d must be at least 1", k), domain = NA)
    nte <- nrow(test)
    if (ncol(test) != p)
        stop("dims of 'test' and 'train' differ")
    clf <- as.factor(cl)
    nc <- max(unclass(clf))
    Z <- .C(VR_knn, as.integer(k), as.integer(l), as.integer(ntr),
        as.integer(nte), as.integer(p), as.double(train),
as.integer(unclass(clf)),
        as.double(test), res = integer(nte), pr = double(nte),
        integer(nc + 1), as.integer(nc), as.integer(FALSE),
as.integer(use.all))
    res <- factor(Z$res, levels = seq_along(levels(clf)), labels =
levels(clf))
```

Lampiran 12. (Lanjutan)

```
    res <- factor(Z$res, levels = seq_along(levels(clf)), labels =
levels(clf))
    if (prob)
        attr(res, "prob") <- Z$pr
    res
}
<bytecode: 0x07f36028>
<environment: namespace:class>
```

Lampiran 13. *Pre Processing Text*

a) Stemming

Anggota f-pdip dpr masinton pasaribu sebut sudah ada damai dengan
jelang limpah berkas ke jaksa jessica mala wongso sangka kasus mati wayan mirna
wali kota jakarta utara rustam effendi sebut daeng aziz bukan tokoh masyarakat
meski sudah siap pilih kendara dinas masing masing all new toyota camry
majelis agama islam katolik budha dan khonghucu komentar kait aktivitas
⋮
Usai diluncurkan akhir Januari lalu, motor sport teranyar Yamaha, Xabre

b) Stopword

1	"anggota f-pdip dpr masinton pasaribu damai "
2	"jelang limpah berkas jaksa jessica mala wongso sangka mati wayan mirna"
3	"wali kota jakarta utara rustam effendi daeng aziz tokoh masyarakat"
4	" pilih kendara dinas all new toyota camry"
5	"majelis agama islam katolik budha khonghucu komentar kait aktivitas"
⋮	⋮
500	"Usai diluncurkan Januari , motor sport teranyar Yamaha, Xabre"

c) Casefolding

1	anggota fpdip dpr masinton pasaribu damai
2	jelang limpah berkas jaksa jessica mala wongso sangka mati wayan mirna
3	wali kota jakarta utara rustam effendi daeng aziz tokoh masyarakat
4	pilih kendara dinas all new toyota camry
5	majelis agama islam katolik budha khonghucu komentar kait aktivitas
⋮	⋮
500	usai diluncurkan januari motor sport teranyar yamaha xabre

Lampiran 13. (Lanjutan)

d) Tokenizing

a	abm	acara	aceh	adil	adu	⋯	zootopia
0	0	2	0	0	0	…	0
0	0	0	0	1	0	…	0
0	0	0	0	0	0	…	0
0	0	0	0	0	0	…	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
0	0	0	4	0	0	…	0

DAFTAR LAMPIRAN

Halaman

Lampiran 1. Judul dan Kategori Artikel 49
Lampiran 2. *Confusion Matrix* Kernel Linier tiap Fold.....50
Lampiran 3. *Confusion Matrix* Kernel Polynomial tiap Fold.....52
Lampiran 4. Akurasi Data Testing Kernel Linier.....54
Lampiran 5. Akurasi Data Testing Kernel Polynomial 55
Lampiran 6. Prediksi Artikel Berita metode SVM dengan Kernel Linier dan Polynomial 56
Lampiran 7. *Confusion Matrix* metode KNN 59
Lampiran 8. Prediksi Artikel Berita metode KNN.....61
Lampiran 9. *Syntax K-Fold Cross Validation*.....64
Lampiran 10. *Syntax Stopword* dan *Case Folding*.....65
Lampiran 11. Syntax *Stemming* Sastrawi yang Telah Dimodifikasi.....66
Lampiran12. Syntax *K-Nearest Neighbor* 67
Lampiran13. *Pre Processing Text*.....70

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan analisis dan pembahasan yang telah dilakukan dapat diambil kesimpulan dari penelitian ini.

1. Metode *Support Vector Machine* dengan menggunakan kernel linier dan polynomial didapat hasil kernel polynomial lebih baik dari kernel linier pada *word vector* 3784. Untuk dibandingkan dengan hasil KNN digunakan kernel polynomial dengan hasil yang didapatkan pada data *testing* untuk masing-masing pengukuran performa dari nilai rata-rata 10 *fold* didapatkan akurasi total, *recall*, *precision*, dan *F-Measure* sebesar 93.2%, 93.2%, 93.63% dan 93.14%.
2. Metode *K-Nearest Neighbor* dengan menggunakan k=2 pada data *testing* dengan *word vector* sebesar 3784 didapatkan hasil tiap performa dari nilai rata-rata 10 *fold* didapatkan akurasi total, *recall, precision,* dan *F-Measure* adalah 60%, 60%, 81.15%, 68.90%.
3. Perbandingan antara kedua metode SVM dan K-NN didapatkan hasil SVM kernel *polynomial* lebih baik dibandingkan dengan K-NN.

5.2 Saran

Saran untuk penelitian selanjutnya adalah agar didapatkan performansi lebih baik maka menggunakan kernel yang sesuai dengan jenis data. Untuk prediksi kelas pada *multiclass* SVM hanya menggunakan metode *one against one* dimana terdapat metode lainnya seperti *one against all* pada kasus *multiclass*.

DAFTAR PUSTAKA

Ariadi, D. & Fithriasari, K. (2015). Klasifikasi Berita Indonesia Menggunakan Metode Naïve Bayessian Classification dan Support Vector Machine dengan Confix Stripping Stemmer. *Jurnal Sains dan Seni ITS, 4*(2), 2337-3520.

Arifin, T. (2015). Implementasi Metode K-Nearest Neighbor untuk Klasifikasi Citra Pap Smear Menggunakan Tekstur Nukleus. *Jurnal Informatika Vol II.*

Bengio, Y. (2004). No Unbiased Estimator of the Variance of K-Fold Cross-Validation. *Journal of Machine Learning Research 5 (2004) 1089–1105*.

Buana, P. W. , & Putra, I. K.G.D.(2012). Combination of K-Nearest Neighbor and K-Means based on Term Re-weighting for Classify Indonesian News. *International Journal of Computer Applications* 11(50).0975-8887.

Cristianini, N., & Shawe-Taylor , J. (2000). An Introduction to Support Vector Machine. Cambridge: Cambridge University Press.

Dasarathy, B. V.(1990). "Nearest Neighbours (NN) Norms,NN Pattern Classification Techniques". IEEE Computer Society Press,

Darujati, C., & Gumelar, A. B. (2012). Pemanfaatan Teknik Supervised untuk Klasifikasi Teks Bahasa Indonesia. Sistem Informasi, Fakultas Ilmu Komputer, Universitas Narotama Surabaya.

Dragut, E., Fang, F., Sistla, P., Yu, C., & Meng, W. (2009). Stop Word and Related Problems in Web Interface Integration. VLDB Endowment.

Dunham, Margareth H.(2003). Data Mining Introductory and Advanced Topics New Jersey. Prentic Hall.

G.G. Enas and S. C. Choi.(1986). "Choice of the smoothing parameter and efficiency of k-nearest Neighbours

classification," Computers & Mathematics with Applications, vol. 12, no. 2, pp. 235-244.

Guduru, N. (2006). *Text Mining With Support Vector Machines And Non-Negative Matrix Factorization Algorithms.* University Of Rhode Island.

Gupta, V., Lehal, G.S. (2009). "A Survey of Text Mining Techniques and Application". *Journal of Emerging Technologies in Web Intelligence*. Vol. 1, pp. 60-75.

Hamzah, A. (2012). Klasifikasi Teks dengan Naïve Bayes Classifier (NBC) untuk Pengelompokan Teks Berita dan Abstract Akademis. In Prosiding Seminar Nasional Apikasi Sains & Teknologi (SNAST) Periode III, p. B269-B277. Yogyakarta.

Han, Jiawei dan Kamber, Micheline. (2006). *Data Mining : Concept and Techniques Second Edition*, Morgan Kauffman Publishers.

Hearst, M. A. (1997). Text Data Mining: Issues, Techniques, and The Relationship to Information Access. In Presentation notes for UW/MS workshop on data mining (pp. 112-117).

Hotho, A., Nurnberger, A., & Paass, G. (2005). *A Brief Survey of Text Mining.* Kassel: University of Kassel.

Hsu, C.-W., Chang, C.-C., & Lin, C.-J. (2010). *A Practical Guide to Support Vector Classification.* Taiwan: Department of Computer Science National Taiwan University.

Joachims, T. (1998). Text Categorization with Support Vector Machines: Learning with Many Relevant Features.

Kementrian Komunikasi dan Informatika. (2014). *Pengguna Internet Di Indonesia Capai 82 Juta*. Diakses pada 20 Januari 2016, dari http://kominfo.go.id/publikasi/content/detail/3980/kemkominfo-pengguna-internet-di-indonesia-capai-82 juta/0/berita_satker

Nugroho, A. S. dkk. (2003). *Support Vector Machine : Teori dan Aplikasinya dalam Bioinformatika*. IlmuKomputer.Com. Indonesia.

Prasetyo, E. (2012). Data Mining Konsep dan Aplikasi Menggunakan MATLAB. Yogyakarta: ANDI Yogyakarta.

Prilianti, K. R., & Wijaya, H. (2014). Aplikasi Text Mining untuk Automasi Penentuan Tren Topik Skripsi dengan Metode K-Means Clustering. Jurnal Cybermatika, 2(1).

Sembiring, K. (2007). Penerapan Teknik Support Vector Machine untuk Pendeteksian Intrusi pada Jaringan,Institut Teknologi Bandung, Bandung

Schneider,J. (1997). *Cross Validation*. Diakses pada 29 Juni 2016,https://www.cs.cmu.edu/~schneide/tut5/node42.html

Tala, F. Z. (2003). A Study of Stemming Effects on Information Retrieval in Bahasa Indonesia. Institute for Logic, Language and Computation Universeit Van Amsterdam.

Weiss, S. M. (2010). Text mining: Predictive Methods for Analyzing Unstructured Information. New York: Springer.

Y. Hamamoto, S. Uchimura, and S. Tomita.(1997) "A Bootstrap Technique for Nearest Neighbours Classifier Design," IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, vol. 19, no. 1, pp. 73-79.

Y. Yang and X. Liu.(1999) "A re-examination of text categorization methods," in Proceedings of SIGIR-99, 22nd ACM International Conference on Research and Development in Information Retrieval, Berkeley, pp. 42-49.

BIODATA PENULIS

Penulis, yang bernama lengkap Siti Nur Asiyah merupakan anak kedua dari dua bersaudara. . Penulis lahir di Lamongan, pada tanggal 21 Februari tahun 1994. Penulis menempuh masa studinya di SMA Trimurti Surabaya , D3 Statistika ITS dan melanjutkan ke Lintas Jalur Statistika ITS dengan NRP 1314105016. Selama menjadi mahasiswa, penulis juga aktif dalam organisasi menjadi staff Pengembangan Sumber Daya Mahasiswa HIMASTA-ITS 12/13 dan Wakil Ketua HIMASTA-ITS 13/14. Selain itu penulis juga aktif dalam mengikuti program kreatif mahasiswa dalam bidang penelitian. Penulis yang memiliki motto hidup *"Talk More Do More"* dan merupakan seseorang yang tidak pernah berhenti mencari kesibukan dan ingin mengabdikan dirinya untuk bermanfaat bagi Masyarakat. Segala saran dan kritik membangun selalu diharapkan oleh penulis melalui email sitinurasiyah@live.com.

"Jangan membuang waktu sedetik pun dalam penyesalan, karena menyesali kesalahan di masa lalu sama saja dengan menularkan kembali kesalahan itu pada diri sendiri"