BAB II
TINJAUAN PUSTAKA

Pada bab ini menjelaskan tentang tinjauan pustaka dan berhubungan dengan penelitian yang pernah dilakukan sebelumnya.

2.1 Penelitian Terdahulu

Penelitian pertama berjudul "Eksperimen Sistem Klasifikasi Analisa Sentimen *Twitter* Pada Akun Resmi Pemerintah Kota Surabaya Berbasis Pembelajaran Mesin" yang ditulis oleh Nuke Y. A. Faradhillah, Renny P. Kusumawardhani dan Irmasari Hafidz. Penelitian ini bertujuan untuk menganalisis sentimen masyarakat terhadap Pemerintah Kota Surabaya di media sosial *Twitter*. Data yang digunakan didapat dari akun *Twitter* resmi Pemerintah Kota Surabaya @SapawargaSby dan akun @e100ss. Data diambil selama periode 1 September 2015 sampai 13 Oktober 2015. Penelitian ini menggunakan algoritma *Support Vector Machine* (SVM) dan juga *Naïve Bayes.* Hasil pengujian data menunjukan algoritma *Support Vector Machine* (SVM) memiliki tingkat akurasi yang lebih baik daripada algoritma *Naïve Bayes*, yakni sebesar 78,66%. Pada algoritma *Naïve Bayes,* metode praproses penghapusan kata (*stopword removal*) menghasilkan tingkat akurasi yang lebih baik dibandingkan yang tidak menggunakan. Sementara untuk algoritma *Support Vector Machine* (SVM), metode praproses penghapusan kata (*stopword removal*) tidak terlalu berpengaruh pada tingkat akurasi yang dihasilkan [22].

Penelitian kedua berjudul "Perbandingan Metode *Naïve Bayes,* KNN, dan *Decision Tree* Terhadap Analisis Sentimen Transportasi KRL *Commuter Line*" yang ditulis oleh Nova Tri Romadloni, Imam Santoso dan Sularso Budilaksono. Penelitian ini bertujuan untuk melakukan perbandingan antara algoritma *Naïve Bayes*, *K-Nearest Neighbor*, dan *Decision Tree* terhadap analisis sentimen transportasi KRL *Commuter Line*. Data yang digunakan didapat dari media sosial *Twitter.* Penelitian ini menggunakan algoritma *Naïve Bayes, K-Nearest Neighbor*, dan *Decision Tree.* Hasil pengujian data menunjukan algoritma *Naïve Bayes* mendapat tingkat akurasi sebesar 80%,

precision 66,67%, *sensitifity* 100%, dan *specifity* 66,67%. Algoritma *K-Nearest Neighbor* menghasilkan tingkat akurasi sebesar 80%, *precision* 100%, *sensitifity* 50%, dan *specifity* 100%. Pada algoritma *Decision Tree*, didapatkan tingkat akurasi sebesar 100%, *precision* 100%, *sensitifity* 100%, dan *specifity* 100% [23].

Penelitian selanjutnya yaitu penelitian yang dilakukan oleh Sigit Kurniawan, Windu Gata, Dewi Ayu Puspitawati, Nurmalasari, Muhamad Tabrani dan Kadinar Novel dengan judul "Perbandingan Metode Klasifikasi Analisis Sentimen Tokoh Politik Pada Komentar Media Berita Online". Penelitian ini bertujuan untuk melakukan perbandingan metode klasifikasi analisis sentimen tokoh politik pada komentar media berita online. Metode yang dijadikan perbandingan yakni metode *Support Vector Machine* (SVM) dan *Naïve Bayes*. Data berupa komentar masyarakat yang didapat dari situs media berita online Detik, Tribun News, Kompas, Merdeka, Viva, dan Kompasiana. Total data yang diperoleh berjumlah 1480, diambil selama periode waktu 16 Februari 2018 sampai dengan 31 Mei 2018. Proses klasifikasi kedua metode dioptimasikan dengan *Particle Swarm Optimization* (PSO). Hasil akurasi yang didapat dari kedua metode dengan optimasi *Particle Swarm Optimization* (PSO) lebih tinggi daripada tanpa menggunakan optimasi *Particle Swarm Optimization* (PSO). Untuk algoritma *Support Vector Machine* (SVM), tingkat akurasi berada di angka 76,09% tanpa menggunakan optimasi *Particle Swarm Optimization* (PSO). Sementara dengan penggunaan optimasi *Particle Swarm Optimization* (PSO), nilai akurasi mengalami peningkatan menjadi 78,40%. Pada algoritma *Naïve Bayes* nilai akurasi tanpa menggunakan optimasi *Particle Swarm Optimization* (PSO) berada di angka 68,21%, sedangkan dengan opptimasi *Particle Swarm Optimization* (PSO), nilai akurasi mengalami peningkatan menjadi 74,98% [24].

Penelitian yang terakhir berjudul "Analisis Sentimen Masyarakat Terhadap *E-Commerce* Pada Media Sosial Menggunakan Metode *Naïve Bayes* Classifier (NBC) Dengan Seleksi Fitur *Information Gain* (IG)" yang dilakukan oleh Abdan Syakuro. Penelitian ini bertujuan untuk mengetahui seberapa baik

performa metode *Naïve Bayes* dengan seleksi fitur *Information Gain* (IG) dalam klasifikasi sentimen analisis. Data yang digunakan didapat dari media sosial *Twitter* dengan kata kunci yang berkaitan dengan beberapa *e-commerce* yang ada di Indonesia, diantaranya Lazada, Bukalapak, dan Tokopedia. Total data yang diperoleh berjumlah 3000 cuitan yang diambil selama 3 bulan dari bulan Januari sampai Maret 2017. Hasil akurasi yang didapat dari metode *Naïve Bayes* dengan seleksi fitur *Information Gain* (IG) lebih baik daripada metode *Naïve Bayes* tanpa menggunakan *Information Gain* (IG), yaitu mencapai 88,8% [25]

Tabel 2.1 Penelitian Terdahulu

No.	Judul Paper, Penulis dan Tahun Penelitian	Masalah	Metode	Hasil
1	Eksperimen Sistem Klasifikasi Analisa Sentimen Twitter Pada Akun Resmi Pemerintah Kota Surabaya Berbasis Pembelajaran Mesin. Nuke Y. A. Faradhillah, Renny P. Kusumawardani, Irmasari Hafidz, 2016 [22].	Metode praproses penghapusan kata (*stopword removal*) tidak berpengaruh pada hasil akurasi *algoritma Support Vector Machine* (SVM).	*Naïve Bayes* dan *Support Vector Machine* (SVM).	Hasil pengujian data menunjukan algoritma *Support Vector Machine* (SVM) memiliki tingkat akurasi yang lebih baik daripada algoritma *Naïve Bayes*, yakni sebesar 78,66%. Pada algoritma *Naïve Bayes*, metode praproses penghapusan kata (*stopword removal*) menghasilkan tingkat akurasi yang lebih baik dibandingkan yang tidak menggunakan. Sementara untuk algoritma *Support Vector Machine* (SVM), metode praproses penghapusan kata (*stopword removal*) tidak terlalu berpengaruh pada tingkat akurasi yang dihasilkan.

2	Perbadingan Metode Naïve Bayes, KNN, dan Decision Tree Terhadap Analisis Sentimen Transportasi KRL Commuter Line. Nova Tri Romadloni, Imam Santoso, Sularso Budilaksono, 2019 [23].	Nilai akurasi yang kurang baik saat nilai k kecil.	*Naïve Bayes*, *Decision Tree*, dan *K-Nearest Neighbor.*	Hasil pengujian data menunjukan algoritma *Naïve Bayes* mendapat tingkat akurasi sebesar 80%, *precision* 66,67%, *sensitifity* 100%, dan *specifity* 66,67%. Algoritma *K-Nearest Neighbor* menghasilkan tingkat akurasi sebesar 80%, *precision* 100%, *sensitifity* 50%, dan *specifity* 100%. Pada algoritma *Decision Tree,* didapatkan tingkat akurasi sebesar 100%, *precision* 100%, *sensitifity* 100%, dan *specifity* 100%.
3	Perbandingan Metode Klasifikasi Analisis Sentimen Tokoh Politik Pada Komentar Media Berita Online. Sigit Kurniawan, Windu Gata, Dewi Ayu Puspitawati, Nurmalasari, Muhamad Tabrani, Kadinar Novel, 2019 [24].	Baik algoritma Support Vector Machine maupun *Naïve Bayes* tidak mencapai akurasi terbaiknya tanpa optimasi dari *Particle Swarm Optimization.*	*Support Vector Machine* (SVM), *Naïve Bayes.*	Hasil akurasi yang didapat dari kedua metode dengan optimasi *Particle Swarm Optimization* (PSO) lebih tinggi daripada tanpa menggunakan optimasi *Particle Swarm Optimization* (PSO).

4	Analisis Sentimen Masyarakat Terhadap E-Commerce Pada Media Sosial Menggunakan Metode Naïve Bayes Classifier (NBC) Dengan Seleksi Fitur Information Gain (IG). Abdan Syakuro, 2017 [25].	Banyaknya bias yang mengurangi tingkat akurasi dari algoritma *Naïve Bayes.*	*Naïve Bayes Classifier* (NBC).	Hasil akurasi yang didapat dari metode *Naïve Bayes* dengan seleksi fitur *Information Gain* (IG) lebih baik daripada metode *Naïve Bayes* tanpa menggunakan *Information Gain* (IG), yaitu mencapai 88,8%.

2.2 Dasar Teori

Dasar teori menjelaskan tentang teori yang dibutuhkan untuk mendukung dalam penelitian ini.

2.2.1 COVID-19

COVID-19 merupakan penyakit menular yang disebabkan oleh Coronavirus jenis terbaru. Coronavirus merupakan suatu kelompok virus yang dapat menyerang baik hewan maupun manusia. Biasanya Coronavirus menyebabkan terjadinya infeksi saluran pernafasan pada manusia, mulai batuk, pilek hingga *Middle East Syndrome* (MERS), serta *Severe Acute Respiratory Syndrome* (SARS). Penyakit COVID-19 pertama kali ditemukan pada akhir tahun 2019 tepatnya pada bulan Desember di Wuhan, Tiongkok. Penularan penyakit ini umumnya ditandai dengan demam, batuk kering, dan tubuh yang mudah terasa lelah. Gejala lain yang mungkin dialami oleh orang yang terjangkit atau tertular penyakit ini meliputi rasa nyeri dan sakit pada anggota tubuh tertentu, hidung tersumbat, sakit kepala, konjungtivis, sakit tenggorokan, diare, kehilangan indera penciuman atau perasa, ruam pada kulit.

Penyebaran COVID-19 terbilang cepat. Penyakit ini dapat menyebar dan juga menular lewat cairan dari hidung ataupun mulut yang keluar saat seseorang yang terinfeksi berbicara, batuk atau bersin. Cairan ini juga dapat menempel pada benda dan permukaan lainnya di sekitar kita seperti meja, gagang pintu, pegangan tangan, bahkan uang. Penularan dapat terjadi apabila seseorang yang baru saja menyentuh benda-benda yang terkena cairan dari orang yang sudah terinfeksi COVID-19 kemudian menyentuh mata, hidung, atau mulut. Hal terburuk yang dapat disebabkan oleh penyakit COVID-19 yaitu kematian. Hingga sekarang belum ditemukan vaksin untuk penyakit ini [26].

2.2.2 *Natural Language Processing* (NLP)

Natural Language Processing (NLP) atau yang juga dikenal dengan Pemrosesan Bahasa Alami merupakan salah satu cabang ilmu dari *artificial*

intelligence atau kecerdasan buatan yang memberi pelajaran terhadap komputer agar dapat memahami, menafsirkan serta memanipulasi bahasa alami manusia [27]. Pada NLP dilakukan pembuatan model komputasi bahasa yang memungkinkan adanya interaksi manusia dan komputer dengan perantara bahasa alami yang digunakan oleh manusia [28]. NLP bertujuan memecahkan masalah untuk memahami bahasa manusia dengan berbagai aturan gramatika dan aturan semantiknya serta mengubah bahasa tersebut menjadi representasi formal yang dapat dipahami dan diproses oleh komputer [29].

Pada praktek dan penerapannya, NLP ini memiliki tantangan, yang diantaranya adalah sebagai berikut [29]:

1. *Part-of-speech Tagging*

   Part-of-speech tagging atau penandaan kelas kata ini sulit dilakukan karena pengelasan kata sangat bergantung pada konteks penggunaan kata tersebut.

2. *Text Segmentation*

   Text segmentation atau segmentasi teks sulit dilakukan terhadap bahasa tulis yang tidak memiliki pembatas kata yang spesifik, seperti misalnya pada bahasa Thailand, Mandarin dan Jepang. Segmentasi teks juga sulit dilakukan terhadap bahasan lisan yang seringkali membaurkan bunyi antarkata.

3. *Word Sense Disambiguation*

   Word sense disambiguation atau makna kata yang ambigu. Seringkali ditemukan suatu kata yang memiliki lebih dari satu makna baik dalam bentuk homonim atau polisemi. Untuk dapat membedakan makna kata tersebut harus terlebih dahulu dilihat dari konteks penggunaannya.

4. *Syntatic Ambiguity*

   Syntatic ambiguity atau ambiguitas sintaksis. Suatu bahasa memiliki berbagai macam struktur kalilmat yang berbeda. Untuk

memilih struktur kalimat yang tepat dibutuhkan penggabungan informasi semantik dan juga kontekstual.

5. *Imperfect or Irregular Input*

   Imperfect or irregular input atau masukan yang tidak sempurna atau tidak biasa. Kesalahan penulisan baik secara pengetikan atau gramatikal juga mempersulit proses NLP.

6. *Speech Act*

   Speech act atau penuturan. Terkadang struktur kalimat tidak dengan tepat menggambarkan tujuan dan maksud dari penulis, gaya bahasa dan konteks kalimat menentukan maksud yang dinginkan.

2.2.3 *Sentiment Analysis*

Sentiment analysis merupakan merupakan kajian tentang cara menyelesaikan dan memecahkan masalah dari berdasarkan opini masyarakat, sikap serta emosi suatu entitas, dimana entitas tersebut dapat mewakili individu [30]. *Sentiment analysis* atau yang juga disebut *opinion mining* merupakan proses memahami, mengekstrak serta mengolah data tekstual secara otomatis guna mendapatkan informasi yang terkandung dalam suatu kalimat opini. Dilakukannya analisis sentimen ini bertujuan untuk melihat pendapat atau kecenderungan opini terhadap suatu masalah ataupun objek oleh seseorang, apa memiliki kecenderungan positif, negatif, atau netral [31].

Sentiment analysis merupakan bagian dari NLP yang membangun sistem untuk mengenali serta mengekstraksi opini dalam bentuk teks. Di internet banyak terdapat informasi berbentuk teks yang tidak terstruktur, seperti misalnya pada blog, forum, media sosial, situs yang berisi *review*, dan lain-lain. Informasi yang tidak terstruktur ini kemudian diubah menjadi data yang terstruktur dengan *sentiment analysis.* Terdapat tiga jenis *sentiment analysis* yaitu *Fine-Grained Sentiment Analysis*, *Intent Sentiment Analysis* dan *Aspect-Based Sentiment Analysis.* Perbedaan ketiganya adalah

pada *Fine-Grained Sentiment Analysis* akan mengelompokkan respon menjadi beberapa kategori seperti positif, negatif dan netral. *Intent Sentiment Analysis* akan mengidentifikasi sebuah pesan atau teks termasuk ke dalam golongan pesan atau teks keluhan, saran, pendapat, pertanyaan, atau pujian terhadap sesuatu, misalnya terhadap produk atau layanan. Sementara pada *Aspect-Based Sentiment Analysis* dapat lebih fokus pada elemen-elemen yang lebih spesifik dari suatu teks [32].

2.2.4 *Twitter*

Twitter merupakan sebuah situs media sosial yang mulai dikembangkan pada tahun 2006. Situs ini pertama kali ditemukan oleh Jack Dorsey dan Evan Williams. Twitter merupakan *social networking* dimana memungkinkan penggunanya dapat saling berkomunikasi satu sama lain melalui fitur yang bernama *tweet.* Dengan fitur *tweet* pengguna dapat membuat tulisan atau teks sebanyak 280 karakter [33]. Tidak hanya *tweet*, saat ini *Twitter* memiliki banyak fitur lainnya seperti *direct message* yang memungkinkan pengguna berkomunikasi satu sama lain dengan lebih privat, *story* yang memungkinkan pengguna dapat merekam momen baik itu foto atau video secara langsung atau *realtime*, *live* yang memungkinkan pengguna melakukan siaran langsung (melalui aplikasi pihak ketiga *Periscope*), *voice note* memungkinkan pengguna untuk merekam suara, dan masih banyak fitur lainnya.

Twitter menyediakan API (*Application Programming Interface*). *Twitter* API diperuntukkan bagi pengembang. Dengan *Twitter* API memungkinkan pengguna dapat membaca, menulis dan mengambil data dari *Twitter.* Penggunaan *Twitter* API ini juga memungkinkan pengembang untuk mengambil informasi atau data pengguna di *Twitter* atau suatu subjek di lokasi tertentu [34].

2.2.5 Klasifikasi

Klasifikasi merupakan kategori *supervised learning*. Metode klasifikasi mempelajari data yang ada dan memprediksi data-data lainnya. Ada beberapa algoritma yang sering digunakan dalam klasifikasi misalnya *Naïve Bayes, K-Nearest Neighbor, Support Vector Machine* dan lain-lain.

a. *Naïve Bayes*

Naïve Bayes merupakan sebuah metode klasifikasi yang berdasar pada teorema *Bayes*. Metode ini memprediksi data di masa yang akan mendatang berdasarkan data sebelumnya atau data yang sudah ada. Ciri utama dari metode *Naïve Bayes* ini adalah asumsi yang kuat akan independensi dari masing-masing kondisi. Berikut merupakan formulasi dari *Naïve Bayes* :

$$posterior\ probability = \frac{likelihood * class\ prior\ probability}{evidence} \quad (2.1)$$

Notasi umum *posterior probability* dapat dituliskan juga sebagai berikut :

$$P(C|X) = \frac{P(X|C)P(c)}{P(x)} \quad (2.2)$$

Keterangan :

x = Data dengan *class* yang belum diketahui

c = Hipotesis data merupakan suatu *spesific class*

P(C|X) = *Posterior probability*

P(c) = *Prior probability*

P(x|c) = Probabilitas berdasarkan kondisi hipotesis

P(c) = Probabilitas c

Nilai *evidence* selalu sama untuk tiap kelas pada suatu sampel. Nilai posterior nantinya akan dibandingkan dengan nilai posterior

kelas lain untuk menentukan ke kelas mana suatu sampel akan diklasifikasikan[35].

Berikut merupakan contoh perhitungan manual dari *Naïve Bayes.* Akan digunakan lima data yang terdiri dari dua data yang berlabel positif, dua data berlabel negatif, dan satu data yang belum memiliki label baik positif ataupun negatif. Data positif akan dilabeli dengan '1', sementara data negatif akan dilabeli dengan '0'. Perhitungan dilakukan untuk mengetahui apakah data yang belum memiliki label tersebut masuk ke dalam kategori data berlabel positif atau data berlabel negatif.

Tabel 2.2 Sampel Data yang Akan Diklasifikasi

Dokumen ke-	Isi Dokumen	Label
1	Covid ajar aku sabar	1
2	Banyak hikmah balik covid	1
3	Sejak covid aku tambah bodoh	0
4	Males banget pakai masker	0
5	Covid bikin tambah bodoh miskin males	?

- Perhitungan *prior* untuk data dengan label positif :

$$\boldsymbol{P(1) = \frac{2}{4} = 0,5}$$

- Perhitungan *prior* untuk data dengan label negatif :

$$\boldsymbol{P(0) = \frac{2}{4} = 0,5}$$

- Perhitungan probabilitas *term* pada dokumen ke-5 terhadap dokumen yang lainnya :

$$\boldsymbol{P(covid|1) = \frac{2+1}{7+14} = \frac{3}{21} = 0,14}$$

$$\boldsymbol{P(bikin|1) = \frac{0+1}{7+14} = \frac{1}{21} = 0,04}$$

$$\boldsymbol{P(tambah|1) = \frac{0+1}{7+14} = \frac{1}{21} = 0,04}$$

$$\boldsymbol{P(bodoh1) = \frac{0+1}{7+14} = \frac{1}{21} = 0,04}$$

$$\boldsymbol{P(miskin|1) = \frac{0+1}{7+14} = \frac{1}{21} = 0,04}$$

$$\boldsymbol{P(males|1) = \frac{0+1}{7+14} = \frac{1}{21} = 0,04}$$

$$\boldsymbol{P(covid|0) = \frac{1+1}{7+14} = \frac{2}{21} = 0,09}$$

$$\boldsymbol{P(bikin|0) = \frac{0+1}{7+14} = \frac{1}{21} = 0,04}$$

$$\boldsymbol{P(tambah|0) = \frac{1+1}{7+14} = \frac{2}{21} = 0,09}$$

$$\boldsymbol{P(bodoh0) = \frac{1+1}{7+14} = \frac{2}{21} = 0,09}$$

$$\boldsymbol{P(miskin|0) = \frac{0+1}{7+14} = \frac{1}{21} = 0,04}$$

$$\boldsymbol{P(males|0) = \frac{1+1}{7+14} = \frac{2}{21} = 0,09}$$

- Perhitungan dengan persamaan *Naïve Bayes* untuk menentukan kategori atau kelas dari dokumen ke-5.

$$\boldsymbol{P(1|d5) = 0,5 \times 0,14 \times 0,04 \times 0,04 \times 0,04 \times 0,04 \times 0,04}$$
$$\boldsymbol{= 0,000000007168}$$

$$\boldsymbol{P(0|d5) = 0,5 \times 0,09 \times 0,04 \times 0,09 \times 0,09 \times 0,04 \times 0,09 = 0,000000052488}$$

Dari perhitungan di atas, dapat disimpulkan bahwa dokumen ke-5 masuk ke dalam kategori atau kelas dokumen negatif.

b. *K-Nearest Neighbor (KNN)*

K-nearest Neighbor (KNN) merupakan sebuah algoritma yang digunakan untuk melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang memiliki jarak terdekat dengan objek tersebut. Nilai K terbaik tergantung pada data, secara umum nilai K yang tinggi akan mengurangi *noise* pada klasifikasi akan tetapi hal ini membuat batasan antar setiap klasifikasi menjadi kabur [23]. Untuk menghitung tingkat kemiripan tetangga antar 2 objek dapat menggunakan persamaan [36]:

$$Sim(d_i . q_i) = \frac{\sum_{j=1}^{t}(q_{ij} . d_{ij})}{\sqrt{\sum_{j=1}^{t}(q_{ij})^2 . \sum_{j=1}^{t}(d_{ij})^2}} \quad (2.3)$$

Keterangan :

q_{ij} = Bobot istilah *j* pada dokumen *i* = tf_{ij} . idf_j

d_{ij} = Bobot istilah *j* pada dokumen *i* = tf_{ij} . idf_j

2.2.6 *Text Preprocessing*

Tahap ini merupakan tahapan pertama dalam pemrosesan teks. Tahap ini juga merupakan salah satu tahapan yang paling penting dalam *text mining*. *Text* Preprocessing ini merupakan tahap dimana sistem melakukan seleksi data yang kemudian akan diproses pada setiap dokumen. Ada beberapa tahap dalam *text processing* ini diantaranya:

a. *Case Folding*

Pada tahap *case folding*, akan dilakukan pengubahan semua huruf dalam dokumen menjadi huruf kecil. Dan hanya karakter huruf ‘a’ sampai ‘z’ saja yang disisakan. Karakter lainnya akan dihilangkan serta dianggap delimiter.

b. ***Tokenizing***

Tahap ini akan menguraikan deskripsi yang semula berupa kalimat menjadi kata dan menghilangkan delimiter seperti misalnya tanda titik, koma, spasi, serta karakter angka yang terdapat dalam kata tersebut [37].

c. *Filtering* (*Stopword Removal*)

Stopword merupakan kosakata yang tidak bermakna atau dengan kata lain bukan merupakan kata yang unik dari suatu dokumen. Seperti misalnya “di”, “karena”, “dan”, “oleh” dan lain sebagainya. Sebelum proses *filtering* atau *stopword removal* dilakukan, dibuat terlebih dahulu daftar *stopword* atau *stoplist.* Kata-kata yang masuk dalam *stoplist* akan dihapus dari deskripsi sehingga yang tersisa hanyalah kata-kata yang mencirikan isi dari suatu dokumen.

d. *Stemming*

Stemming merupakan proses pemetaan serta penguraian berbagai bentuk dari suatu kata menjadi bentuk kata dasar. Tujuan dari proses *stemming* ini adalah menghilangkan segala bentuk imbuhan baik berupa prefiks, sufiks dan konfiks yang ada dalam setiap kata [38].

2.2.7 *Term Frequency Inverse Document Frequency* (TF-IDF)

TF-IDF merupakan teknik pengambilan informasi yang membebani *term frequency* serta dokumen inversnya. Setiap kata dan istilah memiliki nilai bobot TF dan IDF tersendiri. Nilai bobot tersebut dinamakan TF-IDF. TF-IDF digunakan untuk mengukur *keywords* yang terdapat dalam setiap

dokumen dan menghitung seberapa sering kemunculannya di dalam dokumen [39].

$$IDF = \log\left(\frac{D}{df}\right) \quad (2.4)$$

$$W = tf.IDF \quad (2.5)$$

Keterangan :

d = Dokumen ke-d

W = Nilai TF-IDF

D = Total dokumen

df = Jumlah dokumen yang mengandung kata yang dimaksud/dicari

tf = Banyak kata dicari dalam dokumen

2.2.8 *K-Fold Cross Validation*

K-Fold Cross Validation merupakan salah satu metode yang digunakan untuk evaluasi model prediksi. Metode ini membagi data ke dalam *k* subset dan melakukan pengulangan sebanyak *k* kali guna pembelajaran serta pengujian. Metode ini menggunakan satu subset untuk dijadikan sebagai data uji dan subset lainnya digunakan sebagai data pembelajaran pada setiap pengulangan yang dilakukan [40].

2.2.9 *Python*

Python adalah bahasa pemrograman tingkat tinggi yang dapat digunakan untuk memprogram berbagai macam aplikasi. Pertama kali dirilis pada tahun 1991 oleh Guido van Rossum. Bahasa pemrograman ini memiliki sistem pustaka yang luas, *Python* menyediakan berbagai modul yang siap dipakai untuk berbagai macam keperluan pemrograman [41].

2.2.10 *Scikit-learn*

Scikit-learn merupakan sebuah modul untuk bahasa pemrograman *Python*. *Scikit-learn* mampu memudahkan dalam dalam melakukan

pemrosesan data ataupun *training* data untuk kebutuhan *machine learning* [42]. Teknologi yang mendasari dibangun atau dibuatnya modul *Scikit-learn* ini yaitu *Numpy*, *Scipy*, dan *Cython*. Modul *Scikit-learn* dapat digunakan untuk menangani kasus-kasus seperti *Classification*, *Clustering*, *Regression, Dimensionality Reduction, Model Selection, Preprocessing* [43][44][45][46].