BAB II
TINJAUAN PUSTAKA

2.1. Penelitian Terkait

Penelitian terkait dengan topik analisis sentimen cukup banyak, berikut beberapa penelitian yang tekait dengan analisa sentimen yang menggunakan seleksi fitur dan pemrosesan data awal antara lain penelitian yang dilakukan oleh:

Penelitian Bo Pang dan Lillian Leemengambil judul *Sentiment analysis using subjectivity summarization based on minimum cuts* tahun 2004. Penelitian ini menggunakan dataset review filem dari IMDB yang meliputi 1.000 sentimen positif dan 1.000 sentimen negatif dan pada penelitian ini meneliti adanya *polarity* dalam setiam kalimat yang ada dalam sentimen review dengan menggunakan metode *minimum cut* atau dengan mengambil subjektivitas dan *polarity* data mampu memampatkan data menjadi lebih bersih lebih singkat namun tetap merepresentasikan dokumen yang asli dan juga menyertakan metode klasifikasi naïve bayes.

Penelitian Jonathon Read mengambil judul *Using emoticons to reduce dependency in machine learning techniques for sentiment classification* Tahun 2005, yaitu dengan memanfaatkan *emoticons* yang ada pada yang mengambil dari artikel sebanyak 20.000 artikel berbahasa inggris yang pada dasarnya menunjukkan emosi penulis saat memberikan penulisan pada artikel atau pada review tersebut. Pada penelitian ini juga menggunakan metode klasifikasi naïve bayes.

Penelitian Ravi Parikh dan Matin Movassate mengambil judul *Sentiment analysis of user-generated twitter updates using various classification techniques* Tahun 2009. Pada penelitaian ini peneliti mengambil dataset dari timeline posting di twitter sebanyak 370 sentimen positif dan 370 sentimen negatif serta menerapkan algoritma unigram dan bigram dengan menggunakan metode klasifikasi naïve bayes dan Maximum Entropy serta menunjukkan kinerja naïve bayes lebih unggul dibandingkan Maximum Entropy.

Penelitian Christian Rohrdantz, Ming C Hao dkk mengambil judul *Feature-Based Visual Sentiment Analysis of Text Document Streams* Tahun 2012, pada penelitian ini

menggunakan dataset yang di ambil dari 50.000 web survei dan 16.000 dari RSS *News Feeds* pada penelitian ini menggunakan metode *Linguistic Preprocessing*,*Feature Sentiment Identification*, *Context Identification* untuk memilih fitur dan memetekan secara visual dari data-data yang berada dalam website sehingga dapat diperoleh berbagai pola visual dari data review atau survei meupun berita yang diperoleh dari website secara *realtime*.

2.2. Analisis Sentimen

Analisis sentiment atau disebut juga *Sentiment analysis* atau *opinion mining* merupakan sebuah cabang penelitian di domain *text mining* yang mulai marak pada tahun 2003. *Opinion mining* atau *sentiment analysis* adalah riset komputasional dari opini, sentimen dan emosi yang diekspresikan secara tekstual. Jika diberikan satu set dokumen teks D yang berisi opini (atau sentimen) mengenai suatu objek, maka *opinion mining* bertujuan untuk mengekstrak atribut dan komponen dari objek yang telah dikomentari pada setiap dokumen dan untuk menentukan apakah komentar tersebut positif, atau negatif [10]. Pang *et al*. (2008) menyebutkan bahwa *opinion mining* adalah bagian pekerjaan yang melakukan *review* yang berkaitan dengan perlakuan komputasional opini, sentimen, dan subjektifitas dari teks [11].

Istilah *opinion mining* muncul dalam *paper* Dave *et al*. (2003) yang dipublikasikan dalam *prosiding* konferensi WWW pada tahun 2003. Publikasi tersebut menjelaskan istilah *popularity* dalam komunitas sangat terkait dengan pencarian*web* atau pencarian informasi. Menurut Dave, *et al*. (2003) *opinion mining tool* yang ideal adalah alat yang memproses sekumpulan hasil pencarian untuk item tertentu yang menghasilkan suatu daftar atribut produk (kualitas, fitur dan lainnya) dan melakukan agregasi dari opini-opini tersebut [12].

Secara umum, opini dapat diekspresikan atas apa saja, misalnya produk, layanan, individu, organisasi atau suatu kejadian.*Term object* digunakan untuk menunjukan entitas yang telah dikomentari. Menurut Liu suatu *object* memiliki seperangkat komponen dan satu *set* atribut [10]. Suatu *object* O adalah suatu entitas yang dapat berupa suatu produk, topik, orang, kejadian atau organisasi. O direpresentasikan

sebagai hirarki dari komponen, sub komponen dan lain sebagainya. Misalnya pernyataan '*kualitas suara dari Nokia 3630 sangat jelek*', maka *object* disini adalah '*Nokia 3630*' dan atribut yang dikomentari adalah '*kualitas suara'*.

Liu mendefinisikan bahwa suatu kalimat opini merupakan kalimat yang mengekspresikan opini positif atau negatif secara eksplisit atau implisit. Liu juga mengatakan bahwa suatu kalimat opini dapat berupa kalimat subjektif atau kalimat objektif. Opini eksplisit merupakan opini yang secara eksplisit diekspresikan terhadap fitur atau objek dalam suatu kalimat subjektif. Sedangkan opini implisit merupakan opini terhadap fitur atau objek yang tersirat dalam suatu kalimat objektif [10]. Misalnya kalimat '*Kualitas suara dari telepon ini luar biasa*' merupakan opini yang positif dan eksplisit. Sedangkan kalimat '*Earphone ini rusak dalam dua hari'* merupakan opini yang negatif dan implisit. Liu juga mengatakan bahwa meskipun kalimat '*Earphone ini rusak dalam dua hari'* menyampaikan fakta objektif, namun secara implisit kalimat ini mengindikasikan opini negatif terhadap *'earphone'*. Secara umum, kalimat objektif menyiratkan opini postif, negatif ataupun netral [10].

Hingga sekarang, hampir sebagian besar penelitian di bidang *sentiment analysis* hanya ditujukan untuk Bahasa Inggris karena memang *Tools/Resources* untuk bahasa inggris sangat banyak sekali. Namun untuk penelitian kali ini peneliti mencoba meneliti menggunakan sentimen yang menggunakan bahasa Indonesia.

2.3. Pemrosesan Awal Teks

Document Preprocessing yang dilakukan terdiri dari tiga tahapan antara lain:

2.3.1. *Transform Cases*

Dengan fitur *transform cases* kita dapat secara otomatis mengubah semua huruf pada teks menjadi huruf kecil semua atau menjadi huruf kapital semua, pada penelitian ini semua huruf dirubah kedalam huruf kecil karena mayoritas teks berupa tulisan opini yang sebagian besar merupakan huruf kecil semua.

2.3.2. *Filter Tokenize*

Tokenization adalah tugas memisahkan deretan kata di dalam kalimat, paragraf atau halaman menjadi token atau potongan kata tunggal atau *termmed word*. Tahapan ini juga menghilangkan karakter-karakter tertentu seperti tanda baca serta memfilter berdasarkan panjang teks.

2.3.3. *Filter Stop Word (Indonesia)*

Stopword didefinisikan sebagai term yang tidak berhubungan *irrelevant* dengan subyek utama dari database meskipun kata tersebut sering kali hadir di dalam dokumen. Contoh *stopwords* adalah *a, ada, adalah, adanya, adapun, agak, agaknya, agar, akan, akankah* dan masih banyak lagi. dan untuk daftar *Stopword* untuk bahasa indonesia dibuat secara manual dan di integrasikan kedalam *software* Rapidminer. Kata-kata yang terlalu sering muncul dalam dokumen-dokumen bukanlah pembeda yang baik. Bahkan kata-kata yang muncul 80% dalam dokumen-dokumen tidak berguna dalam proses retrieval. Kata-kata ini disebut dengan istilah *stopwords* dan umumnya tidak dijadikan index term. Kandidat umum stopword adalah article, preposisi, dan konjungsi.

Eliminasi *stopwords* bermanfaat dengan adanya pengurangan ukuran strukur index hingga 40%. Karena pengurangan ukuran index, beberapa kata kerja, kata sifat, dan kata keterangan lainnya dapat juga dapat dimasukkan juga ke dalam daftar stopword. Namun eliminasi *stopwords* dapat menyebabkan penurunan nilai recall (jumlah dokumen yang dihasilkan dan relevan/jumlah dokumen relevan).

2.4. Naïve Bayes

Teorema Bayes adalah sebuah pendekatan untuk sebuah ketidaktentuan yang diukur dengan probabilitas. Teorema bayes dikemukakan oleh Thomas Bayes. Thomas Bayes hidup pada abad 18 yang merupakan orang yang sangat terkenal dalam bidang probabilitas. Pengklasifikasi Bayesian juga berguna dalam pembenaran teoritis untuk pengklasifikasi lain yang tidak secara eksplisit menggunakan teorema Bayes[13].
Berikut *teorema bayes* :

$$P(X|H) = \frac{P(H|X)P(H)}{P(X)} \quad (1)$$

Naive bayes adalah penyederhanaan dari *teorema bayes*. Berikut rumus *naive bayes* :

$$P(X|H) = P(H|X)P(H) \quad (2)$$

Keterangan :

X	: data dengan class yang belum diketahui
H	: hipotesis data x merupakan suatu class spesifik
P(H|X)	: probabilitas hipotesis H berdasarkan kondisi X (posteriori probability)
P(H)	: probabilitas hipotesis H (prior probability)
P(X|H)	: probabilitas X berdasar kondisi pada hipotesis H
P(X)	: probabilitas dari X

Naïve Bayesian Classifier menyederhanakan hal ini dengan asumsi bahwa fitur-fitur yang terdapat didalamnya saling tidak tergantung atau independen, setiap kata independen satu sama lain.

Naïve Bayesian Classifier menyederhanakan hal ini dengan asumsi bahwa fitur-fitur yang terdapat didalamnya saling tidak tergantung atau independen, setiap kata independen satu sama lain.

Beberapa keuntungan naïve bayes [7], diantaranya:

a. Kuat terhadap pengisolasi gangguan pada data
b. Jika terjadi kasus missing value ketika proses komputasi sedang berlangsung, maka objek tersebut akan diabaikan
c. Dapat digunakan untuk data yang tidak relevan

2.5. Pembobotan Atribut (Attribute Weighting)

Untuk memilih atribut atau pembobotan atribut (*attribute weighting*) yang tepat terhadap analisis sentimen maka dapat dilakukan dengan beberapa metode pembobotan atribut diantaranya:

2.5.1. Weight by Correlation

Operator ini memberikan skema pembobotan berdasarkan korelasi. Dan operator ini menghitung korelasi setiap atribut dengan atribut label dan mengembalikan nilai absolut atau kuadrat sebagai berat [14].

Perhitungan bobot dapat dilakukan dengan menggunakan rumus sebagai berikut:

$$w(a) = \begin{cases} |Corr(a)| & (|Corr(a)| \geq t) \\ 0 & (|Corr(a)| \geq t) \end{cases} \quad (3)$$

Dimana:

Corr(a) = koefisien korelasi antara *feature* "a" dan sentimen
t = nilai *treshhold*

Jarak terkecil akan menandakan bahwa hubungan semakin dekat, sedangkan jarak dapat di hitung dengan rumus

$$d(u, v) = \sum_i w(a_i) f(u_i, v_i) \quad (4)$$
$$f(u_i, v_i) = (u_i - v_i)^2$$

Dimana

u = data training
v = data testing
a_i = *feature* i

2.5.2. Weight by Chi Squared Statistic

Operator ini menghitung relevansi dari fitur oleh komputasi untuk setiap atribut contoh masukan menetapkan nilai statistik *chi-squared* sehubungan dengan atribut class [14].

Perhitungan dapat dilakukan dengan rumus sebagai berikut:

$$x^2(t, c) = \frac{N \times (A \times D - B \times C)^2}{(A+B) \times (C+D) \times (A+C) \times (B+D)} \quad (5)$$

A = the number of document in category *c* and containing *t*
B = the number of document not in category *c* and containing *t*
C = the number of document in category *c* and not containing *t*
D = the number of document not in category *c* and not containing *t*
N = the total number of document

2.5.3. Weight by **SVM**

Operator ini menggunakan koefisien dari *hyperplance* dihitung oleh **SVM** sebagai bobot fitur[14].

Bobot fitur ini dapat di hitung dengan menggunakan rumus:

$$\tau(r_1, r_2) = \frac{P-Q}{P+Q} = 1 - \frac{2Q}{P+Q} \quad (6)$$

Dimana

r_i = bobot yang diperoleh dengan menerapkan metode terhadap dataset.
P = jumlah elemen yang sama di bagian atas *triangular* matriks dari r_1 dan r_2.
Q = jumlah elemen yang berbeda di bagian atas *triangular* matriks dari r_1 dan r_2.