



# Analisis Sentimen Berbasis Ontologi di Level Kalimat untuk Mengukur Persepsi Produk

Agus Subhan Akbar<sup>a,\*</sup>, Eko Sedyono<sup>b</sup>, Oky Dwi Nurhayati<sup>c</sup>

<sup>a</sup> Mahasiswa Magister Sistem Informasi Universitas Diponegoro

<sup>b</sup> Magister Sistem Informasi Universitas Kristen Satya Wacana

<sup>c</sup> Jurusan Sistem Komputer, Fakultas Teknik, Universitas Diponegoro

Naskah Diterima : 15 Mei 2015; Diterima Publikasi : 17 Juli 2015

## Abstract

The purpose of this research is to do sentiment analysis on tweets data retrieved using ontology framework and using naïve bayes classifier algorithm for classification process. This study is based on the habits of twitter users who frequently writes opinion, expression, or sentiment on a specific product, especially smartphones. These tweets can be used as a basis for sentiment analysis on a particular product. The method used in this study include the use of ontology framework for tweets retrieval that match the domain of the discussion and the use of naïve bayes classification algorithm for data classification. Classification process carried past the 3 pieces of layer classification to fine tune the final result of classification. Three layers of classification used include buzz/promo classification (classifying tweets into buzz and not-buzz tweets), subjectivity classification (classifying not-buzz tweets into subjective and objective tweets), and sentiment classification (classifying subjective tweets into positive, negative, or neutral tweets). The resulted software can classify tweets with high accuracy. This software was trained and tested with the composition of 25:75, 50:50, 75:25 from sample data and tested 10 times for each composition. Average accuracy of the system reached 84.16%, 86.15%, and 87.54% for each composition. The result showed that by employing this system, product marketing stakeholders can determine the level of user sentiment expressed in the form of tweets. The method used in this study could be developed to improve the accuracy of classification systems.

**Keywords:** Sentence Level Sentiment Analysis; Ontology; Naïve Bayes, Classification

## Abstrak

Penelitian ini bertujuan untuk melakukan analisis sentimen pada data *tweets* yang diambil dengan menggunakan kerangka ontologi dan menggunakan algoritma klasifikasi *naïve bayes* untuk proses klasifikasinya. Penelitian ini didasarkan pada adanya kebiasaan pengguna *twitter* yang sering menulis pendapat, ekspresi, atau sentimen terkait produk tertentu, khususnya *smartphone*. Data *tweets* tersebut bisa digunakan sebagai basis analisis untuk mendapatkan sentimen atas produk tertentu. Metode yang digunakan dalam penelitian ini meliputi penggunaan kerangka ontologi untuk pengambilan data *tweets* yang sesuai dengan domain pembahasan dan penggunaan algoritma klasifikasi *naïve bayes* untuk proses pengklasifikasian data *tweets* sesuai dengan kategori yang ditentukan. Proses klasifikasi yang dilakukan melewati 3 buah lapisan klasifikasi untuk lebih menajamkan hasil akhir klasifikasi. Tiga lapisan klasifikasi yang digunakan meliputi klasifikasi *buzz/promo* (digunakan untuk memisahkan *tweets* promo dan bukan promo), klasifikasi subyektifitas (digunakan untuk memisahkan *tweets* bukan promo menjadi *tweets* subyektif dan obyektif), dan klasifikasi sentimen (digunakan untuk mendapatkan polaritas *tweets* subyektif dalam bentuk sentimen positif, negatif, atau netral). Hasil penelitian ini berupa perangkat lunak yang dapat mengklasifikasi *tweets* dengan tingkat akurasi yang tinggi. Perangkat lunak tersebut dilatih dan diuji dengan komposisi 25:75, 50:50, dan 75:25 dari data dan pengujian dilakukan sebanyak 10 kali untuk setiap komposisi. tingkat akurasi klasifikasi rata-rata dari sistem yang dihasilkan mencapai 84.16%, 86.15%, dan 87.54%. Perangkat lunak ini dapat dikembangkan untuk membantu pihak-pihak terkait pemasaran produk *smartphone* sebagai alat bantu untuk mengetahui tingkat sentimen dari pengguna yang diekspresikan dalam bentuk *tweets*. Metode yang digunakan dalam penelitian ini masih bisa dikembangkan untuk meningkatkan tingkat akurasi klasifikasi sistem.

**Kata kunci:** Analisis Sentimen Level Kalimat; Ontologi; *Naïve Bayes*, Klasifikasi

## 1. Pendahuluan

Informasi teks tertulis bisa dibagi menjadi dua kelompok besar, yaitu fakta dan opini (Liu, 2010). Fakta atau pengetahuan merupakan informasi

obyektif terhadap suatu benda atau keadaan, sedangkan opini merupakan informasi subyektif yang berupa perasaan, penghargaan, ataupun penilaian terhadap benda atau suatu keadaan. Sesuai dengan

\*) Penulis korespondensi: [agussa@live.com](mailto:agussa@live.com)

kondisinya, opini ini bisa bersifat positif ataupun negatif.

Sifat positif atau negatif dari opini ini bisa digunakan untuk mengukur seberapa besar dukungan pembuat opini atas sebuah topik permasalahan. Bagi pembuat topik hal ini bisa digunakan untuk pengambilan langkah-langkah berikutnya terkait dengan topik permasalahan tersebut.

Persepsi produk merupakan gambaran pendapat, opini, atau perasaan konsumen terhadap produk tersebut. Persepsi ini bisa memiliki orientasi positif ataupun negatif. Bagi produsen, persepsi ini menjadi salah satu tolak ukur bagi keberhasilan produk yang dipasarkan tersebut. Sedangkan bagi konsumen, informasi persepsi ini berguna sebagai salah satu faktor untuk mengambil keputusan membeli produk tersebut. Produk yang populer akan mendapatkan opini ataupun komentar yang berjumlah banyak, bisa ratusan atau ribuan, sehingga diperlukan metode yang tepat untuk menghitung orientasi persepsi produk ini (Unnamalai, 2012).

Kemunculan Web 2.0 mengubah interaksi orang dengan internet yang semula sebagai pemakai pasif menjadi pemakai aktif. Salah satu aplikasi yang muncul dalam lingkup Web 2.0 adalah aplikasi *twitter*. Dengan aplikasi ini pengguna internet bisa mengekspresikan pendapatnya secara bebas baik dari sisi isi maupun waktu. Hal ini menjadikan *twitter* berisi data yang besar. Sampai dengan tahun 2013, pengguna aktif *twitter* mencapai 140 juta dengan jumlah *tweets* (teks yang dikirim) rata-rata mencapai 340 juta per-hari (Kontopoulos *et al.*, 2013).

*Tweets* yang berjumlah ratusan juta per-hari tersebut memiliki karakteristik yang khas yaitu adanya batasan panjang per-*tweets* yang berjumlah 140 karakter. Sehingga memunculkan tulisan-tulisan yang tidak mengikuti standar baku tata bahasa, adanya singkatan-singkatan, ataupun simbol-simbol tertentu yang digunakan oleh pengguna untuk mengekspresikan pendapatnya. Hal ini menimbulkan tantangan penelitian untuk mendapatkan metode-metode yang tepat untuk mendapatkan opini dan sentimen dari *tweets* tersebut (Zhang *et al.*, 2011).

Analisis sentimen merupakan bidang riset yang terus berkembang sampai sekarang. Dalam bidang ini, penelitian difokuskan pada studi komputasi atas pendapat, sikap, dan emosi yang dituliskan dalam bentuk teks (Medhat *et al.*, 2014). Sasaran analisis sentimen ini adalah untuk menemukan opini/pendapat yang ada dalam tulisan teks, mendapatkan sentimen yang ada dalam opini yang ditemukan, dan pada ujungnya mendapatkan polaritas/klasifikasi atas sentimen tersebut apakah positif/mendukung atau negatif/tidak mendukung.

Analisis sentimen ini terbagi menjadi tiga jenis, yang terdiri atas: *document level sentiment analysis*, *sentence level sentiment analysis*, dan *aspect-level sentiment analysis* (Medhat *et al.*, 2014). *Document*

*level sentiment analysis* merupakan bidang analisis sentimen dengan masukan dokumen pembahasan tentang satu tema yang dianggap sebagai satu unit masukan. *Sentence level sentiment analysis* disisi lain merupakan bidang analisis sentimen dengan masukan berupa satu kalimat sabagi satu unit masukan. Level yang lebih detail dari analisis sentimen ini adalah *aspect-level sentiment analysis* dengan melakukan klasifikasi sentimen atas aspek tertentu dari entitas yang dibahas, seperti sentimen atas spesifikasi dari produk yang lebih detail dari sekedar sentimen atas produk.

Algoritma yang digunakan untuk melakukan klasifikasi sentimen terbagi menjadi dua bagian besar yaitu algoritma yang berbasis kamus (Taboada *et al.*, 2011; Maks dan Vossen, 2012; Unnamalai, 2012; Kontopoulos *et al.*, 2013) dan berbasis pembelajaran (Ghiassi *et al.*, 2013; He dan Zhou, 2011; Zhou *et al.*, 2014). Penggunaan algoritma klasifikasi sentimen berbasis kamus dengan cara memecah kalimat yang akan dianalisis menjadi sejumlah kata pembentuknya. Kata-kata yang mengandung sentimen dipilih, diproses dengan menggunakan kamus sentimen online seperti *SentiWordNet*, untuk mendapatkan nilai orientasi dari kata-kata tersebut. Berdasarkan nilai orientasi dari kata-kata tersebut, orientasi sentimen atas kalimat itu bisa ditentukan apakah dominan positif, negatif atau netral.

Algoritma klasifikasi sentimen berbasis pembelajaran disisi lain menggunakan proses pembelajaran terhadap data training yang sudah disediakan. Algoritma pembelajaran yang dipakai diantaranya *Naïve Bayes (NB)* (He dan Zhou, 2011), *Support Vector Machine (SVM)* (Ghiassi *et al.*, 2013; He dan Zhou, 2011), *Maximum Entropy (ME)* (He dan Zhou, 2011), atau *Fuzzy-NN* (Zhou *et al.*, 2014). Data pelatihan untuk keperluan pembelajaran disiapkan dan ditandai apakah positif, negatif, atau netral. Berdasarkan hasil pelatihan, masukan kalimat baru yang akan dianalisis dimasukkan ke sistem yang sudah dilatih dan sistem menghasilkan orientasi atas kalimat tersebut apakah positif, negatif, atau netral. Algoritma klasifikasi *Naïve Bayes (NB)* dibuktikan menghasilkan klasifikasi yang lebih baik dibandingkan dengan algoritma *Support Vector Machine (SVM)*, dan *Maximum Entropy (ME)* melalui 10 kali percobaan (He dan Zhou, 2011). *Naïve Bayes* juga digunakan dalam deteksi intrusi untuk mengklasifikasi data menjadi lima kelompok kelas yaitu: Normal, *Probe (gathering information)*, *DoS (denial of service)*, *U2R (user to root)*, *R2L (remote to local)* (Mukherje dan Sharma, 2012).

Pendekatan terbaru dalam bidang analisis sentimen ini adalah penggunaan teknologi semantik web untuk lebih meningkatkan hasil analisis yang didapatkan dari penggunaan pemrosesan bahasa natural dan analisis sentimen (Peñalver-Martinez *et al.*, 2014). Analisis dalam penelitian tersebut dilakukan pada dokumen seperti pada ulasan lengkap

tentang produk dengan kemungkinan aspek-aspek detail dari entitas dengan nilai sentimen yang berbeda untuk setiap aspeknya. Teknik semantik web yang digunakan adalah pemilihan fitur untuk analisis sentimen dengan menggunakan kerangka ontologi, yang pada fase berikutnya menghasilkan data vektor yang digunakan sebagai masukan untuk analisis sentimen.

Kerangka ontologi merupakan pemaparan konsep pengetahuan beserta hubungan antar konsep tersebut sehingga tercipta keterkaitan antar masing-masing konsep. kerangka ini bisa digambarkan dalam sebuah pohon pengetahuan. Tujuan penggambaran ini supaya bisa dimengerti oleh manusia maupun mesin / komputer. Penggunaan kerangka ontologi dalam penelitian ini untuk mempertajam atau membatasi hasil pencarian data *tweets* yang sesuai dengan domain yang akan diteliti.

Penyusunan hasil klasifikasi sentimen dalam rentang waktu tertentu akan lebih bisa menggambarkan perkembangan persepsi orang atas topik tertentu. Sehingga pihak yang berkepentingan bisa mengambil langkah-langkah tertentu atas hasil perkembangan persepsi tersebut. Visualisasi grafik batang ataupun berbentuk lingkaran akan lebih memudahkan bagi pihak yang berkepentingan untuk mendapatkan pengertian atas hasil klasifikasi tersebut (Chen *et al.*, 2014).

Penelitian ini akan menggunakan kerangka ontologi untuk memetakan data *tweets*. Data *tweets* yang didapat, diproses dengan *stanford pos tagger* untuk mendapatkan struktur detail kalimat yang digunakan untuk memisahkan *tweets* subyektif dan obyektif. *Tweets* subyektif (berisi pendapat) diklasifikasi dengan algoritma *naïve bayes* untuk mendapatkan orientasi dari *tweets* tersebut. Hasil klasifikasi akan ditampilkan dalam bentuk angka ataupun grafik sehingga memudahkan bagi pengguna untuk mendapatkan pengertian atas hasil klasifikasi sentimen tersebut.

**2. Kerangka Teori**

**2.1. Analisis Sentimen**

Analisis sentimen merupakan studi komputasi linguistik tentang pendapat, ekspresi, dan perasaan orang terhadap entitas, individu, topik, atau obyek lain yang dituliskan dalam bentuk teks (Liu, 2011). Pendapat ini penting, karena seringkali individu ataupun perusahaan memutuskan sesuatu terlebih dahulu memandang pendapat dari pihak lain.

Opini/pendapat ini bisa direpresentasikan dengan lima variabel yang bisa ditulis dalam persamaan (1) berikut ini:

$$Opinion = (e_i, a_{i,j}, oo_{i,j,k,l}, h_k, t_i) \tag{1}$$

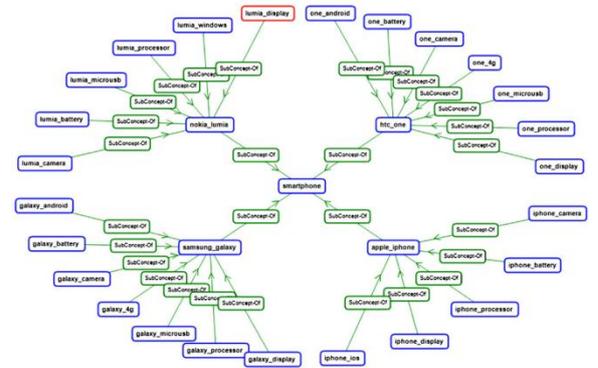
Dimana  $e_i$  adalah entitas atau topik yang dibicarakan,  $a_{i,j}$  merupakan aspek dari  $e_i$ . Sedangkan  $oo_{i,j,k,l}$

merupakan orientasi dari  $a_{i,j}$  terhadap  $e_i$ . Orientasi  $oo$  ini bisa bernilai positif, negatif, ataupun netral. Variabel  $h_k$  merupakan subyek/orang yang menuliskan opini, dan  $t_i$  merupakan waktu saat  $h_k$  menulis opini tersebut.

**2.2. Model Ontologi**

Ontologi merupakan suatu cara untuk memaparkan pengetahuan atas sebuah konsep dan hubungannya dengan konsep lain sehingga tercipta hubungan keterkaitan yang bisa direpresentasikan dalam bentuk pohon pengetahuan. Representasi ini ditujukan untuk bisa dimengerti baik oleh mesin maupun pengguna.

Komponen dalam ontologi terdiri atas: obyek, klas, atribut, dan relasi yang menghubungkan antar obyek. Untuk domain *smartphone* sebagai contoh, konsep atau istilah yang ada didalamnya meliputi merk, sistem operasi, dan spesifikasi teknis dari perangkat. Spesifikasi teknis dari perangkat ini bisa didetailkan meliputi prosesor yang ada didalamnya, kapasitas memori, fasilitas koneksi, baterai, kamera, dan spesifikasi teknis lainnya. Salah satu pemodelan ontologi dengan menggunakan *OntoGen* (<http://ontogen.ijs.si>) untuk *smartphone* seperti yang terlihat padagambar1 berikut ini:



Gambar 1. Visualisasi ontologi domain *smartphone* dengan *OntoGen* (Kontopoulos *et al.*, 2013)

**2.3. Stanford POS Tagger**

Informasi teks tersusun dengan struktur tata bahasa yang sesuai dengan bahasa yang digunakan. Struktur kata dalam tata bahasa meliputi adanya kata benda, kata kerja, kata sifat, dan kata keterangan beserta variasi yang lain. Salah satu yang membedakan bahwa informasi teks tersebut mengandung opini/pendapat dengan yang bersifat obyektif adalah penggunaan kata sifat dan atau kata keterangan. Untuk itu perlu adanya sebuah metode untuk memproses kalimat itu dan membaginya menjadi kata per kata sesuai kelompoknya.

POS Tagger (*Part-of-speech Tagger*) merupakan perangkat lunak pemrosesan bahasa natural yang digunakan untuk mengurai kata-kata penyusun kalimat dan mengelompokkannya ke dalam kelas kata masing-masing (Toutanova *et al.*, 2003). Salah

satu perangkat lunak POS Tagger ini adalah *Stanford Pos Tagger* (<http://nlp.stanford.edu/software/tagger.shtml>). Kelas kata yang didukung oleh perangkat lunak ini seperti dalam tabel 1 berikut:

Tabell. Kelas kata dalam proyek *Penn Treebank* ([https://www.ling.upenn.edu/courses/Fall\\_2003/ling001/penn\\_treebank\\_pos.html](https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html))

Tag	Deskripsi
CC	<i>Coordinating conjunction</i>
CD	<i>Cardinal number</i>
DT	<i>Determiner</i>
EX	<i>Existential there</i>
FW	<i>Foreign word</i>
IN	<i>Preposition or subordinating conjunction</i>
JJ	<i>Adjective</i>
JJR	<i>Adjective, comparative</i>
JJS	<i>Adjective, superlative</i>
LS	<i>List item marker</i>
MD	<i>Modal</i>
NN	<i>Noun, singular or mass</i>
NNS	<i>Noun, plural</i>
NNP	<i>Proper noun, singular</i>
NNPS	<i>Proper noun, plural</i>
PDT	<i>Predeterminer</i>
POS	<i>Possessive ending</i>
PRP	<i>Personal pronoun</i>
PRP\$	<i>Possessive pronoun</i>
RB	<i>Adverb</i>
RBR	<i>Adverb, comparative</i>
RBS	<i>Adverb, superlative</i>
RP	<i>Particle</i>
VB	<i>Verb, base form</i>
SYM	<i>Symbol</i>
TO	<i>to</i>
UH	<i>Interjection</i>
VBD	<i>Verb, past tense</i>
VBG	<i>Verb, gerund or present participle</i>
VBN	<i>Verb, past participle</i>
VBP	<i>Verb, non-3rd person singular present</i>
VBZ	<i>Verb, 3rd person singular present</i>
WDT	<i>Wh-determiner</i>
WP	<i>Wh-pronoun</i>
WP\$	<i>Possessive wh-pronoun</i>
WRB	<i>Wh-adverb</i>

#### 2.4. IPostagger

IPostagger merupakan POS Tagger yang dikembangkan untuk Bahasa Indonesia berbasis *Hidden Markov Model* (HMM) (Wicaksono dan Purwarianti, 2010). IPostagger menggunakan metode tambahan dengan menyusun pohon afiks yang meliputi awalan dan akhiran disertai dengan penggunaan kosa kata yang diambil dari Kamus Besar Bahasa Indonesia (KBBI – Kateglo) untuk membatasi hasil dari pemrosesan pohon afiks.

Kelas kata (*Tag set*) yang digunakan dalam IPostagger ini seperti dalam tabel 2 berikut ini:

Tabel 2. Kelas kata dalam IPostagger (Wicaksono dan Purwarianti, 2010)

Tag	Keterangan	Contoh
OP	<i>Open Parenthesis</i>	{[{
CP	<i>Close Parenthesis</i>	)]}
GM	<i>Slash</i>	/
;	<i>Semicolon</i>	;
:	<i>Colon</i>	:
“	<i>Quotation</i>	“ “
.	<i>Sentence Terminator</i>	.!?
,	<i>Comma</i>	,
-	<i>Dash</i>	-
...	<i>Ellipsis</i>	...
JJ	<i>Adjective</i>	Kaya, Manis
RB	<i>Adverb</i>	Sementara, Nanti
NN	<i>Common Noun</i>	Mobil
NNP	<i>Proper Noun</i>	Bekasi, Indonesia
NNG	<i>Genitive Noun</i>	Bukunya
VBI	<i>Intransitive Verb</i>	Pergi
VBT	<i>Transitive Verb</i>	Memбели
IN	<i>Proposition</i>	Di, Ke, Dari
MD	<i>Modal</i>	Bisa
CC	<i>Coor-Conjunction</i>	Dan, Atau, Tetapi
SC	<i>Subor-Conjunction</i>	Jika, Ketika
DT	<i>Determiner</i>	Para, Ini, Itu
UH	<i>Interjection</i>	Wah, Aduh, Oi
CDO	<i>Ordinal Numerals</i>	Pertama, Kedua
CDC	<i>Collective Numerals</i>	Bertiga
CDP	<i>Primary Numerals</i>	Satu, Dua
CDI	<i>Irregular Numerals</i>	Beberapa
PRP	<i>Personal Pronouns</i>	Saya, Kamu
WP	<i>WH-Pronouns</i>	Apa, Siapa
PRN	<i>Number Pronouns</i>	Kedua-duanya
PRL	<i>Locative Pronouns</i>	Sini, Situ, Sana
NEG	<i>Negation</i>	Buka, Tidak
SYM	<i>Symbols</i>	@#\$\$%^&
RP	<i>Particles</i>	Pun, Kah
FW	<i>Foreign Words</i>	Foreign, Word

#### 2.5. Subjectivity/Opinion Extraction

*Subjectivity Extraction* merupakan proses untuk memilah apakah *tweets* berisi pendapat ataukah tidak. Jadi proses ini lebih untuk memisahkan *tweets* yang berisi pendapat, ekspresi, atau opini dengan *tweets* yang hanya berisi berita (Liu, 2010).

Salah satu cara untuk memilah apakah *tweets* tersebut berisi opini adalah dengan menggunakan POS tagger dan melakukan pengecekan jika terdapat kata sifat atau kata keterangan maka *tweets* tersebut memiliki kemungkinan besar adalah sebuah opini.

#### 2.6. Sentence Level Sentiment Classification (Naïve Bayes Classifier)

Analisis sentimen ditingkat kalimat merupakan subbidang penelitian dari analisis atau klasifikasi sentimen. Level kalimat berarti yang dianalisis adalah satu kalimat dengan memperhatikan kata-kata penyusun kalimat tersebut.

*Naïve Bayes Classifier* merupakan algoritma klasifikasi text sederhana yang dikembangkan berdasarkan teorema *Bayes* dengan asumsi atas variabel-variabel yang terlibat saling bebas satu dengan yang lainnya. Sehingga pada saat digunakan dalam analisis kalimat, setiap kata dalam kalimat tersebut dianggap saling bebas terhadap kata yang lain. Teknik klasifikasi ini sering digunakan untuk mendeteksi email sampah (*spam email*), klasifikasi teks, dan pengelompokan dokumen.

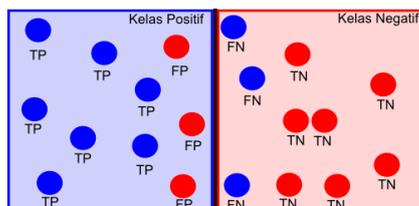
Formulasi yang digunakan dalam algoritma klasifikasi ini ditujukan untuk mendapatkan nilai maksimal dari "*maximum a posteriori*" (MAP) sesuai dengan persamaan(2) dan (3) berikut ini (Zhang, 2004; Manning *et al.*, 2008):

$$C_{map} = \arg \max_{c \in C} (P(c|d)) = \arg \max_{c \in C} (P(c) \prod_{1 \leq k \leq n_d} P(t_k|c)) \quad (2)$$

$$P(t|c) = \frac{T_{ct}+1}{\sum_{t' \in V} (T_{ct'}+1)} = \frac{T_{ct}+1}{\sum_{t' \in V} (T_{ct'}+B')} \quad (3)$$

Dimana  $t_k$  merupakan token/kata yang ada dalam dokumen/kalimat, C merupakan klas pengelompokan (positif, negatif, netral),  $P(c|d)$  merupakan probabilitas bersyarat klas c dari dokumen/kalimat d,  $P(c)$  merupakan probabilitas awal/sebelumnya dari klas c,  $P(t_k|c)$  merupakan probabilitas bersyarat dari token  $t_k$  dalam kelas c, dan B' merepresentasikan jumlah *term*/token yang ada dalam pustaka yang digunakan.

Perhitungan performa merupakan formulasi untuk perhitungan performa yang digunakan. Formula yang digunakan meliputi *precision*, *recall*, *accuracy*, dan *F-Measure* (Haddi *et al.*, 2013). Masing-masing bisa dihitung dari parameter yang berasal dari keberhasilan sistem dalam mengenali fitur yang ada yang meliputi *True Positif (TP)*, *False Positif (FP)*, *True Negatif (TN)*, dan *False Negatif (FN)*. Parameter-parameter ini bisa lebih jelas digambarkan dalam gambar 2 berikut ini:



Gambar 2. Ilustrasi Parameter

Berdasarkan gambar 2 di atas, TP merupakan item positif yang berhasil diklasifikasi sebagai kelas positif, FP merupakan item negatif yang diklasifikasi sebagai kelas positif, TN merupakan item negatif yang diklasifikasi dalam kelas negatif, sedangkan FN merupakan item positif yang diklasifikasi sebagai kelas negatif.

Perhitungan performa untuk klasifikasi biner (2 kelas/kategori) yang diperlihatkan pada gambar 2 di atas bisa dinyatakan seperti pada persamaan (4), (5), (6), dan (7) berikut ini:

$$precision = \frac{TP}{TP+FP} \quad (4)$$

$$recall = \frac{TP}{TP+FN} \quad (5)$$

$$accuracy = \frac{TP+TN}{TP+FP+TN+FN} \quad (6)$$

$$F - Measure = \frac{2 * precision * recall}{precision + recall} \quad (7)$$

Berdasarkan persamaan (4), *precision* merupakan perbandingan atas jumlah klasifikasi yang benar dibandingkan dengan total item yang berhasil diklasifikasi di kelas positif. *Recall*, yang dinyatakan dalam persamaan (5), merupakan perbandingan atas jumlah klasifikasi yang benar dengan jumlah item yang seharusnya berada dalam kelas tersebut. *Accuracy*, yang dituliskan dalam persamaan (6), merupakan perbandingan jumlah ketepatan klasifikasi atas semua item yang diklasifikasi. Nilai *accuracy* ini digunakan untuk perhitungan ketepatan klasifikasi untuk seluruh kelas yang ada dalam klasifikasi.

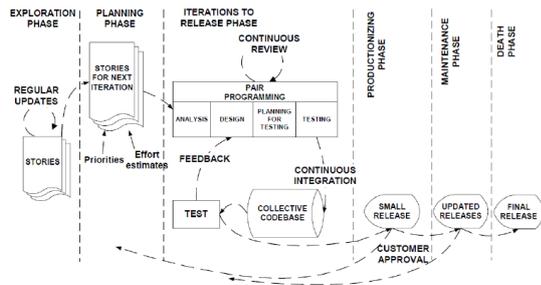
*F-Measure* yang dinyatakan dalam persamaan (7) merupakan perhitungan rata-rata antara *precision* dan *recall*. *F-Measure* merupakan salah satu metode pengambilan rata-rata yang juga disebut dengan *harmonic mean*. Perhitungan ini merupakan nilai rata-rata dari variabel dengan rentang nilai yang sama ( $0 \leq x \leq 1$ ) tetapi dari cara penilaian yang berbeda (*precision* dan *recall*).

Perhitungan *accuracy* yang digunakan untuk mencari tingkat keakuratan model untuk klasifikasi lebih dari dua kelas (*n-class*) bisa dinyatakan dalam persamaan (8) berikut ini:

$$accuracy = \frac{\sum_{i=1}^n TP_i}{\sum_{i=1}^n (TP_i + FP_i)} \quad (8)$$

## 2.7. Siklus Pengembangan Perangkat Lunak

Salah satu metodologi pengembangan perangkat lunak yang ada adalah metodologi *Extreme Programming (XP)*. Siklus pengembangan perangkat lunak yang mengadopsi metodologi ini melewati 5 fase yang terdiri atas: Eksplorasi, Perencanaan, dan iterasi tahapan release, produksi, perawatan, dan penyelesaian. Tahapan-tahapan dalam metodologi *XP* ini bisa digambarkan secara lebih detail dalam gambar 3 berikut ini (Abrahamsson *et al.*, 2002):



Gambar 3. Siklus tahapan dalam metodologi XP (Abrahamsson *et al.*, 2002)

Fase eksplorasi merupakan tahapan bagi pengguna dalam proses pengembangan perangkat lunak untuk merumuskan dan menuliskan kebutuhan-kebutuhan yang paling mendasar dari sistem yang disebut dengan *User Story*. Sedangkan dari sisi pengembang akan mencari sejumlah alternatif arsitektur sistem yang bisa digunakan. Untuk penelitian ini, fase eksplorasi merupakan fase bagi peneliti untuk mempelajari tentang kebutuhan sistem dan mencoba sejumlah alternatif perangkat yang dibutuhkan dalam pengembangan sistem.

Fase perencanaan yang merupakan kelanjutan dari fase sebelumnya digunakan untuk memilah dari sejumlah kebutuhan sistem yang ada. Kebutuhan yang paling utama dan paling menggambarkan sistem dipilih untuk kemudian dimulai pembuatannya. Penjadwalan lebih detail untuk penyelesaian kebutuhan-kebutuhan tersebut juga dibuat pada tahapan ini.

Fase berikutnya adalah iterasi pengembangan sistem. Dalam fase ini ada empat tahapan yang akan diulang beberapa kali untuk menyelesaikan pengembangan sistem. Iterasi langkah yang ada meliputi analisis, desain, perencanaan uji coba, dan uji coba. Jika ada yang kurang dalam satu siklus maka akan ada iterasi berikutnya untuk menyelesaikan saran-saran pengembangan yang ada.

Fase berikutnya adalah tahapan melakukan rilis pertama dari perangkat lunak yang telah dihasilkan. Selama rilis tersebut tentu akan ada kritik dan saran untuk memperbaiki software. Pengembangan ini bisa saja diulang beberapa kali sesuai dengan kritik dan saran yang ada. Tahapan ini sudah masuk ke dalam fase perawatan.

Fase terakhir adalah rilis final. Rilis final dilakukan jika kebutuhan sistem sudah dipenuhi dan sudah berfungsi sesuai dengan kebutuhan. Rilis Final ini juga bisa merupakan akhir dari fase pengembangan perangkat lunak untuk versi tersebut. Fase pengembangan berikutnya bisa dilakukan dengan memulai dari awal.

### 3. Metodologi

#### 3.1. Bahan Penelitian

Penelitian analisis sentimen berbasis ontologi di level kalimat ini akan membutuhkan sejumlah bahan

dan alat penelitian. Bahan penelitian yang dibutuhkan berupa sejumlah data yang digunakan untuk menguji kerangka sistem yang dibangun. Sedangkan alat penelitian meliputi perangkat yang dibutuhkan selama penelitian meliputi perangkat lunak dan perangkat keras.

Bahan-bahan yang akan digunakan dalam penelitian ini adalah data yang diambil dari *twitter* tentang produk yang akan digunakan sebagai bahan untuk masukan sistem klasifikasi yang dikembangkan. Karakteristik data *twitter* atau biasa disebut dengan *tweets* ini meliputi:

- Tweets* memiliki panjang maksimal 140 karakter.
- Tweets* kadang ditambahi dengan *hashtag* yang biasanya untuk lebih fokus pada pembahasan topik tertentu, misalnya: #smartphone
- Tweets* bisa berupa perulangan atau mendukung atas *tweet* yang lain, atau istilahnya *retweet*, biasanya dideteksi dengan adanya simbol RT. *Retweet* ini ada 2 jenis yaitu *Original Retweet* dan *Retweet With Quote*.
- Dalam *Tweets* kadang bisa berisi *link* atau *url*.
- Saat *Tweets* dikirimkan, Informasi waktu yang meliputi jam, tanggal, bulan, dan tahun ikut dikirimkan secara otomatis.

Produk yang akan digunakan dalam penelitian ini terdiri atas data *tweets* yang terkait dengan produk *smartphone*. Data *tweets* terkait *smartphone* ini akan diambil dari *twitter* dengan menggunakan *Twitter API* (API yang disediakan oleh *Twitter*) dengan memanfaatkan pustaka *Twitter4j*.

#### 3.2. Alat Penelitian

Alat-alat yang digunakan dalam penelitian ini meliputi perangkat keras, perangkat lunak sistem, dan perangkat lunak bantu lainnya. Berikut detail spesifikasi peralatan yang digunakan:

- Perangkat Keras komputer dengan spesifikasi prosesor 2.0 Ghz atau yang lebih tinggi, Memory 4 Gb atau yang lebih tinggi, Harddisk 120Gb atau yang lebih tinggi, Kartu Jaringan untuk koneksi internet, dan perangkat pelengkap lainnya.
- Perangkat Lunak Sistem yang digunakan meliputi:
  - Sistem Operasi *Windows 7* atau yang lebih tinggi
  - Java Development Kit* Versi 6 atau lebih tinggi
  - IDE Pengembangan *NetBeans 8*
  - Twitter API SDK, Twitter4J 4.0.2*
  - MySQL Database Server*
  - Apache Http Server*
  - Komponen Pemrograman Grafik *JFreeChart 1.0.19*.

#### 3.3. Prosedur Penelitian

Penelitian ini akan diawali dengan studi pustaka, identifikasi permasalahan, penyusunan kerangka sistem berbasis metode yang dipilih,

pengumpulan data, perancangan sistem, dan ujicoba sistem. Pengembangan perangkat lunak dalam penelitian ini mengikuti konsep *Extreme Programming (XP)*.

### 3.3.1. Identifikasi Permasalahan

*Tweets* yang dikirim ke *twitter* berjumlah sangat besar setiap harinya. Rata-rata per-hari mencapai 340 juta *tweets* (Kontopoulos *et al.*, 2013). Isi dari *tweets* tersebut bervariasi dari semua bidang. Dari yang berisi tentang opini sebuah produk, opini keseharian pengguna, debat antar pengguna, informasi teknologi terbaru, dan promo-promo (*marketing buzz*) yang demikian besarnya. Data *tweets* yang berjumlah besar ini bisa dimanfaatkan untuk mendapatkan penilaian atau sentimen dari pengguna terhadap produk tertentu seperti *smartphone*.

Keterbatasan panjang karakter dalam *tweets* yang hanya berjumlah maksimal 140 karakter terkadang menjadikan para pengguna menuliskan *tweets* mereka dengan kata-kata yang disingkat, menggunakan bentuk lain dari kata tersebut, ataupun dengan memberikan link / url ke halaman web lain.

Beberapa perusahaan periklanan di internet menggunakan *twitter* untuk mendistribusikan konten iklan yang dipesan oleh pemasang iklan dan menerima pendaftaran dari para pengguna *twitter* untuk ikut terlibat mendistribusikan konten iklan tersebut dengan imbalan finansial. Konten-konten ini sering disebut dengan *marketing buzz* atau *tweets* promo. Untuk klasifikasi sentimen, *tweets* seperti ini dikecualikan.

Contoh-contoh *tweets* terkait seperti berikut ini:

1. *The iPhone 6 battery life is amazing*
2. *Useful informations for iPhone 6 users.*  
<http://t.co/M5PjdhUU56>
3. *TODAY DEALS #933 >>*  
<http://t.co/BNpfSME8FMonoprice> 2200mAh  
*External Backup Battery Charger for iPhone 3GS 4...* <http://t.co/hAjLLU0h2x>
4. *Phones Offers >>* <http://t.co/LPK9kSAf35> #034  
*New Original Battery EB-BG900BBEGWW For Samsung Galaxy S5 i9600 G9...*  
<http://t.co/e3k9vb2Z1a>
5. *#BestPrice - Genuine Battery For Samsung Galaxy Not...* [http://t.co/fZnReykes#Deals\\_UK](http://t.co/fZnReykes#Deals_UK) /  
<https://t.co/jj6D7nVuDj> <http://t.co/HhIgXAePIZ>

Dari contoh-contoh tersebut di atas, menjadi sebuah tantangan untuk membangun sebuah sistem klasifikasi sentimen berbasis ontologi atas opini yang disampaikan oleh pengguna *twitter* terkait tentang produk *smartphone* dalam penelitian ini. *Tweets* yang terkait dengan *smartphone* tetapi berisi *marketing buzz* juga perlu dikecualikan dari sistem.

### 3.3.2. Karakteristik Tweets

*Tweets*, pada awalnya, merupakan bentuk ekspresi atau pendapat yang dikirim oleh pengguna ke media sosial *twitter*. Isi *tweets* terkait dengan

bidang apa saja yang menjadi perhatian pengguna. Akan tetapi perkembangan terkini, *tweets* juga digunakan sebagai media penyebar informasi, media iklan (*marketing buzz*), selain tetap sebagai media ekspresi oleh pengguna.

Terkait dengan klasifikasi opini / sentimen dalam penelitian ini, *Tweets* yang digunakan sebagai dasar bisa terdiri atas:

1. *Tweets* asli, yang merupakan asli tulisan dari pengguna, dan
2. *Retweets* yang merupakan dukungan dari pengguna yang lain atas *tweets* asli dengan membagikan *tweets* ini ke pengikutnya dan juga bisa memberikan tambahan komentar terhadap *tweets* tersebut sebelum dibagi. *Retweets* ini sesuai dengan dokumentasi dari *twitter.com* bisa berupa *original retweets* dan *retweets with quote*. *Original retweets* merupakan dukungan terhadap *tweets* asli tanpa menambah komentar lebih lanjut, sedangkan *retweets with quote* bisa berisi tambahan komentar atas *tweets* asli.

Penelitian ini menggunakan pustaka *Twitter4J* yang dapat membedakan masing-masing *tweets* tersebut. Salah satu fungsi dalam pustaka *Twitter4J* adalah fungsi *isRetweet()*. Fungsi ini akan mengembalikan nilai *false* untuk *tweets* asli yang berhasil di *download*. Untuk *original retweets*, fungsi ini akan mengembalikan nilai *true*. Sedangkan untuk *retweets with quote*, fungsi ini akan mengembalikan nilai *false*.

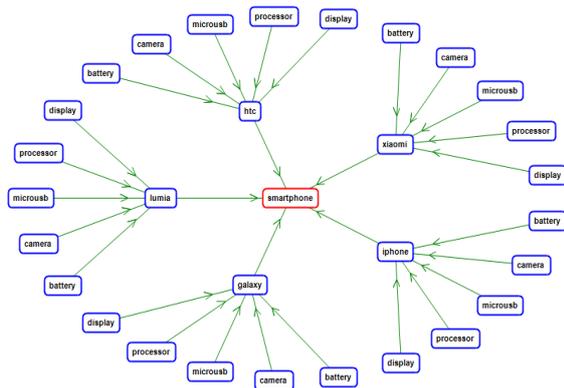
Dengan demikian, untuk *original retweets* akan dihitung sebagai dukungan terhadap *tweets* asli. Jika *tweets* asli berorientasi positif, maka *original retweets* dihitung sama dengan *tweets* aslinya, begitu juga sebaliknya. Di penelitian yang lain *original retweets* ini tidak dimasukkan dalam analisis (Zhang *et al.*, 2011). Sedangkan untuk *tweets* asli dan *retweets with quote* tetap akan dimasukkan ke dalam model klasifikasi sentimen.

### 3.3.3. Penggunaan Ontologi

Desain ontologi sebagai basis pengetahuan memegang peranan penting untuk mempertajam dan mempersempit hasil pencarian yang dilakukan dengan menggunakan pustaka *Twitter4J*. Sehingga hanya *tweets* dari domain yang ditentukan saja yang diberikan oleh *twitter.com* pada saat proses pengambilan / *downloadtweets*.

Desain ontologi yang digunakan dalam penelitian ini dikembangkan dari desain ontologi yang ada di gambar 1 dengan penambahan obyek. Domain yang digunakan adalah *smartphone* dengan obyek meliputi *iphone*, *galaxy*, *htc*, *lumia*, dan *xiaomi*. Untuk masing-masing obyek tersebut ditentukan sejumlah sub konsep/atribut yang meliputi *battery*, *camera*, *microusb*, *processor*, dan *display*. Penyusunan kerangka ontologi dalam penelitian ini dibuat dengan menggunakan perangkat lunak *ontogen* versi 2.0.0.

Hasil rancangan kerangka ontologi yang dibuat ditampilkan dalam gambar 4 berikut ini:



Gambar 4. Visualisasi ontologi domain smartphone untuk penelitian ini dengan Ontogen 2.0.0

Berdasarkan ontologi yang dibangun di atas, dilakukan pencarian tweets di *twitter.com* dengan menggunakan *twitter4j*. Untuk setiap kombinasi obyek dan atribut yang ada disusun sebagai keyword pencarian. Sebagai contoh untuk obyek *iphone* dan atribut *battery* maka keyword yang digunakan adalah “*iphone battery*” (dengan tanpa tanda petik) (Kontopoulos *et al.*, 2013).

3.3.4. Perhitungan Algoritma Naïve Bayes

Algoritma klasifikasi *naïve bayes* digunakan untuk melakukan klasifikasi terhadap data tweets. Sebelum digunakan untuk melakukan klasifikasi, algoritma ini perlu dilatih terlebih dahulu dengan sejumlah data latih. Contoh pelatihan dan penggunaan algoritma ini seperti pada penjelasan berikut ini. Contoh data tweets beserta anotasi manualnya seperti pada tabel 3 berikut ini:

Tabel 3. Contoh data tweets dan anotasinya

Tweets	klas
The iPhone 6 battery life is amazing	positif
This iPhone battery is amazing	positif
The Galaxy s5 battery life is amazing	positif
The camera on the iPhone 6 is literally amazing	positif
iPhone battery is so shit	negatif
Galaxy s5 camera is shit	negatif
iPhone 4s camera quality shit	negatif
iPhone battery life is ridiculously shit	negatif

Berdasarkan data di atas, *prior probability*, P(C), untuk masing-masing kelas adalah  $P(\text{positif}) = 4/8 = 0.5$  dan  $P(\text{negatif})=4/8=0.5$ . Data statistik yang didapat dari data di atas seperti tertuang dalam tabel 4 berikut ini:

Tabel 4. Data statistik dari contoh data

Data	Jumlah
Jumlah kata	49
Jumlah kata tanpa duplikat	18
Jumlah kata dalam kelas positif	28
Jumlah kata dalam kelas negatif	21
Jumlah kata unik dalam kelas positif	10
Jumlah kata unik dalam kelas negatif	17

Perhitungan peluang bersyarat untuk masing-masing kata dihitung dengan menggunakan persamaan (3). Hasil perhitungan peluang bersyarat ini ditampilkan pada tabel 5 berikut ini:

Tabel 5. Perhitungan peluang bersyarat

No	Kata (x)	P(x positif)	P(x negatif)
1	4s	0.0263	0.0526
2	6	0.0789	0.0263
3	amazing	0.1316	0.0263
4	battery	0.1053	0.0789
5	camera	0.0526	0.0789
6	Galaxy	0.0526	0.0526
7	iPhone	0.1053	0.1053
8	is	0.1316	0.1053
9	life	0.0789	0.0526
10	literally	0.0526	0.0263
11	on	0.0526	0.0263
12	quality	0.0263	0.0526
13	ridiculously	0.0263	0.0526
14	s5	0.0526	0.0526
15	shit	0.0263	0.1316
16	so	0.0263	0.0526
17	The	0.1316	0.0263
18	This	0.0526	0.0263

Berdasarkan data *prior probability* masing-masing kelas dan peluang bersyarat masing-masing kata, maka perhitungan klasifikasi bisa dilakukan. Perhitungan klasifikasi untuk kalimat: *iPhone camera is amazing* bisa dilakukan seperti pada tabel 6 berikut ini:

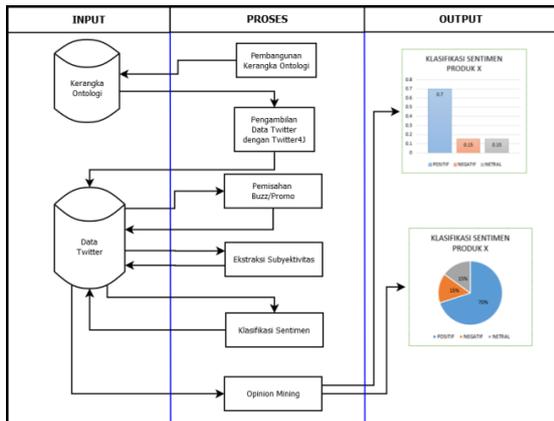
Tabel 6. Perhitungan klasifikasi data

Item Perhitungan	C=positif	C=negatif
Prior probability P(C)	0.5	0.5
P(x=iphone C)	0.1053	0.1053
P(x=camera C)	0.0526	0.0789
P(x=is C)	0.1316	0.1053
P(x=amazing C)	0.1316	0.0263
Product Probability	0.000048	0.000012

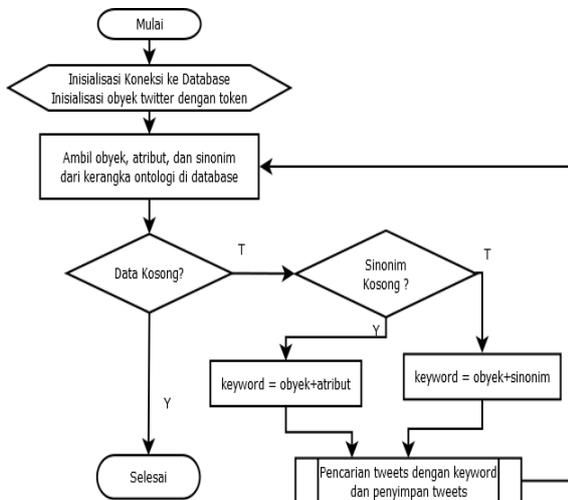
Dengan hasil perhitungan di Tabel 6 dan persamaan (2) maka keputusan klasifikasi *naïve bayes* atas kalimat tersebut adalah positif. Keputusan ini diambil karena nilai total *product probability* untuk kelas positif lebih tinggi dibandingkan dengan kelas negatif.

3.3.5. Rancangan Sistem

Kerangka sistem informasi yang diajukan seperti pada Gambar 5 dan Deskripsi detail alur sistem pada Gambar 6 berikut ini:



Gambar 5. Kerangka sistem informasi klasifikasi sentimen berbasis ontologi



Gambar 6. Deskripsi alur sistem

A. Pembangunan Kerangka Ontologi

Kerangka ontologi dengan domain *smartphone* yang telah ditentukan sebelumnya dalam gambar 4 dibangun dengan menggunakan database *MySQL*. Setiap Obyek disimpan dalam tabel *corpus\_object* begitu juga dengan atribut untuk setiap obyek disimpan dalam tabel *corpus\_attribute*. Struktur yang digunakan seperti pada Tabel 7 dan Tabel 8.

Tabel 7. Struktur tabel *corpus\_object*

Kolom	Tipe	Keterangan
Id	TINYINT	Id setiap obyek
object	VARCHAR(50)	Nama obyek, unik

Tabel 8. Struktur tabel *corpus\_attribute*

Kolom	Tipe	Keterangan
id	TINYINT	Id setiap atribut
attribute	VARCHAR(50)	Nama atribut, unik

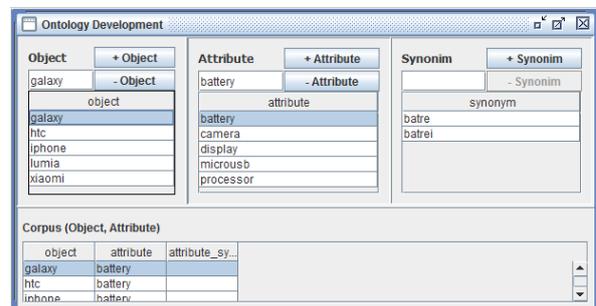
Setiap atribut ada kemungkinan mempunyai sinonim, untuk itu perlu ditambahkan daftar sinonim yang diperlukan untuk memperluas atribut ini. Struktur tabel sinonim dibuat seperti pada Tabel 9.

Tabel 9. Struktur tabel *corpus\_attribute\_synonym*

Nama Kolom	Tipe	Keterangan
id	TINYINT	Id setiap sinonim
at_id	TINYINT	Id atribut
attribute	VARCHAR(50)	Nama atribut, unik

*Keyword* yang digunakan untuk pencarian di *twitter* dengan menggunakan pustaka *Twitter4J* dibentuk dengan menggabungkan obyek dan atributnya atau obyek dengan sinonim dari atribut.

Antar muka pembangunan kerangka ontologi ini dibuat dengan menggunakan bahasa pemrograman *java* dengan perangkat pengembang *netbeans 8.0*. Gambar 7 berikut adalah tampilan antar muka pembangunan kerangka ontologi tersebut:



Gambar 7. Tampilan antar muka pembangunan kerangka ontologi

B. Pengambilan Data Twitter dengan *Twitter4J*

Pengambilan *tweets* di *Twitter* dengan menggunakan pustaka *Twitter4J* dilakukan dengan menggunakan kerangka ontologi yang telah dibentuk pada langkah sebelumnya. Sesuai dengan ketentuan *Twitter*, setiap aplikasi yang akan mengambil data *tweets* harus mengikuti aturan yang sudah ditentukan. Diantara aturan tersebut antara lain :

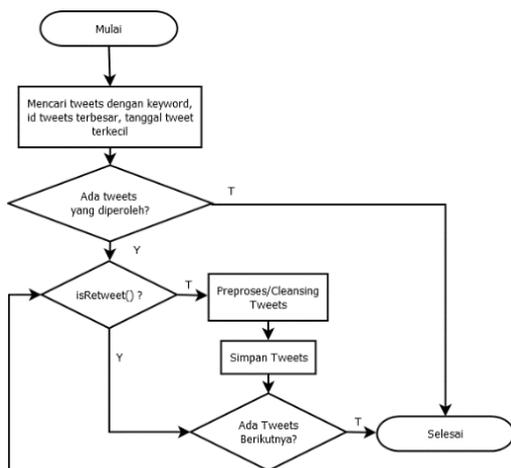
- Setiap aplikasi yang akan menggunakan *Twitter API* harus mendapatkan token otentikasi *OAuth* terlebih dahulu (<https://dev.twitter.com/oauth/overview>).
- Setiap penggunaan fungsi pencarian sesuai dengan standar *API* terbaru versi 1.1, dibatasi dengan maksimal 180 kali pencarian setiap 15 menit (<https://dev.twitter.com/rest/reference/get/search/tweets>).
- Sisa fasilitas fungsi pencarian yang belum digunakan bisa dicek dengan menggunakan api *application/rate\_limit\_status* ([https://dev.twitter.com/rest/reference/get/application/rate\\_limit\\_status](https://dev.twitter.com/rest/reference/get/application/rate_limit_status)).

Dengan menggunakan pustaka *Twitter4j* proses pengambilan data *tweets* bisa lebih mudah, dengan memasukkan data-data yang diperlukan seperti token akses *OAuth*, *keyword* pencarian, beserta data batasan pencarian seperti mulai limit id, limit tanggal, dan jumlah maksimal *tweets* per-pencarian

Berdasarkan kerangka ontologi yang telah disusun, untuk setiap obyek, atribut dan sinonimnya, jika ada sinonim maka diproses terlebih dahulu proses *download*nya dengan kata kunci pencarian “obyek sinonim”. Sedangkan jika sinonimnya kosong maka kata kunci pencarian yang digunakan dalam bentuk “obyek atribut”. Untuk setiap *tweets* yang berhasil di *download* dilakukan proses pembersihan terhadap *tweets* yang meliputi:

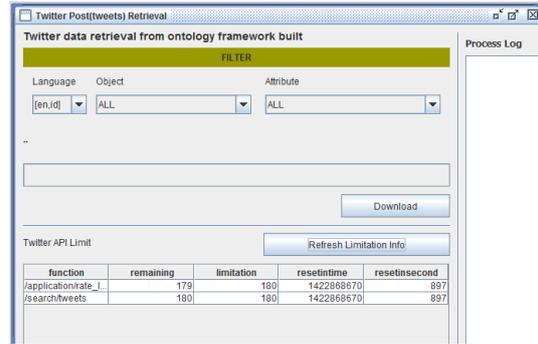
- a. Menghapus karakter non-latin
- b. Menghapus *url* yang ada
- c. Menghapus tanda *hashtag* “#” tanpa menghapus konten yang ada dalam *hashtag* tersebut.
- d. Untuk *tweets* yang merupakan *retweets* asli tidak disimpan dengan pertimbangan tidak ada tambahan informasi kecuali berupa dukungan.
- e. *Tweets* yang merupakan hasil pencarian lebih dari satu kunci pencarian (kombinasi obyek dan atribut yang berbeda) dikesampingkan dari pemrosesan lanjutan.

Diagram alir untuk proses pencarian *tweets* dan penyimpanannya bisa digambarkan dalam Gambar 8 berikut ini:



Gambar 8. Diagram alir untuk pencarian dan penyimpanan *tweets*

Tampilan antar muka blok sistem untuk pengambilan data *tweets* dengan kemungkinan mengambil per-obyek per-atribut atau semua dibuat dengan menggunakan bahasa pemrograman *java* dan memanfaatkan teknik *threading* sehingga pada saat proses komunikasi dengan situs *twitter.com* bisa berjalan di *background* tidak satu proses dengan *gui* dari aplikasi. Tampilan antar muka tersebut seperti pada Gambar 9 berikut ini:



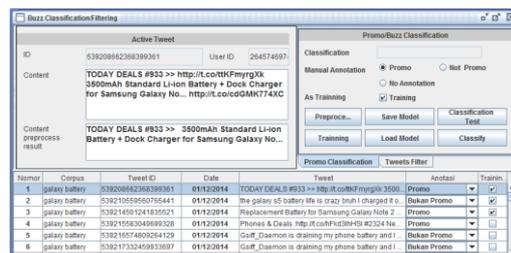
Gambar 9. Tampilan antar muka pengambilan data *twitter*

*Tweets* yang sudah terkoleksi akan diklasifikasi dengan tujuan akhir untuk mendapatkan sentimen atas produk *smartphone* sesuai dengan atribut yang ada. Proses klasifikasi dilakukan melewati tiga lapisan secara berurutan. Keluaran klasifikasi lapisan pertama akan menjadi masukan bagi lapisan klasifikasi berikutnya. Keluaran lapisan klasifikasi terakhir akan menjadi masukan untuk pelaporan rekapitulasi sentimen (*Opinion Mining*).

### C. Pemisahan Buzz/Promo

Lapisan pertama klasifikasi ini digunakan untuk memisahkan *tweets* yang berupa *buzz* periklanan dari data *tweets* yang ada. *Tweetsbuzz* / iklan biasanya dilakukan oleh pengguna yang biasa disebut *buzzer* dengan ikut menyebarkan iklan dari perusahaan periklanan dan mendapatkan imbalan finansial. Biasanya *tweets buzz* seperti ini dilakukan berulang-ulang dan mengandung kata-kata seperti “*Deals*”, “*Best Deals*”, “*Today Deals*” dan beberapa kata lain.

Untuk menangani pemisahan *tweetsbuzz* dan *non-buzz* digunakan algoritma klasifikasi *naïve bayes*. Algoritma klasifikasi *Naïve Bayes* dilatih dengan sejumlah *tweets* yang dianotasi secara manual sebagai *tweets buzz* dan yang bukan *tweets buzz*. Hasil pelatihan merupakan model klasifikasi yang digunakan untuk melakukan pemisahan data *tweets* yang ada. Keluaran dari lapisan klasifikasi ini adalah *tweets* yang sudah dipisahkan antara *tweets buzz* dan *tweets non-buzz*. *Tweets non-buzz* akan diteruskan ke lapisan berikutnya sebagai masukan. Contoh klasifikasi yang dilakukan di layer ini seperti yang terlihat pada gambar 10 berikut ini:



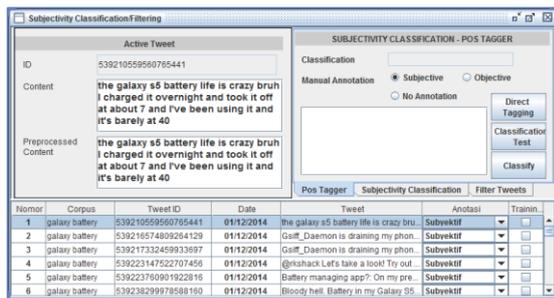
Gambar 10. Lapisan klasifikasi *Tweets Buzz* dan *Non-Buzz*

**D. Ekstraksi Subyektivitas**

Lapisan klasifikasi ini bertujuan untuk memisahkan tweets yang subyektif dan tweets yang obyektif dari masukan tweets lapisan sebelumnya. Tweets subyektif adalah tweets yang berisi opini, persepsi, atau pendapat pengirim tweets. Sedangkan tweets obyektif diluar itu, bisa berupa berita.

Pemisahan tweets ini dilakukan dengan menggunakan algoritma klasifikasi naive bayes yang dilatih dengan data tweets yang sudah dilabeli secara manual sebagai tweets subyektif dan tweets obyektif.

Keluaran lapisan ini adalah tweets yang sudah dikelompokkan sebagai tweets subyektif dan tweets obyektif. Tweets subyektif akan diteruskan sebagai masukan untuk lapisan klasifikasi. Contoh tampilan antar muka lapisan klasifikasi ini seperti pada gambar 11 berikut ini:



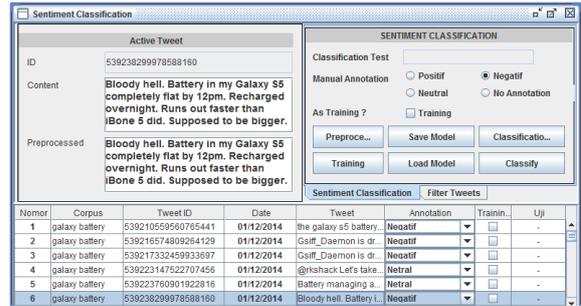
Gambar 11. Lapisan klasifikasi subyektivitas

**E. Klasifikasi Sentimen**

Klasifikasi sentimen merupakan lapisan terakhir untuk klasifikasi tweets untuk mendapatkan persepsi umum tentang produk yang dituju. Masukan dari lapisan ini berasal dari lapisan sebelumnya yaitu tweets yang dikategorikan sebagai tweets subyektif.

Untuk keperluan analisis tweets subyektif yang didapat dianotasi manual sebagai tweets yang positif, negatif, dan netral. Dari data tersebut sebagian digunakan sebagai data pelatihan dan selebihnya digunakan sebagai data uji.

Model yang dibentuk untuk mengklasifikasikan tweets ini menggunakan algoritma klasifikasi naive bayes dengan multi klas sebagai keluaran. Kelas yang dituju adalah positif, negatif, dan netral. Hasil klasifikasi ini disimpan di database sebagai bahan untuk pelaporan dan analisis penambangan data / Opinion Mining. Contoh tampilan antar muka yang dirancang untuk klasifikasi sentimen ini seperti pada gambar 12 seperti di bawah ini:

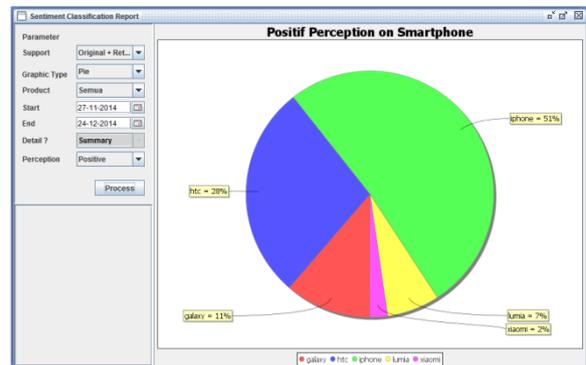


Gambar 12. Lapisan klasifikasi sentimen

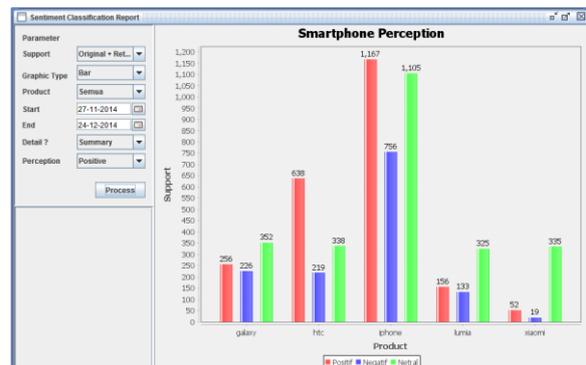
**F. Opinion Mining**

Lapisan OM (Opinion Mining) ini merupakan lapisan akhir dari sistem ini. Tujuan lapisan ini adalah memanfaatkan hasil proses sebelumnya untuk ditampilkan dalam format grafikal ataupun tabular supaya bisa lebih dimengerti.

Contoh tampilan laporan dalam lapisan ini seperti ditunjukkan pada Gambar 13 dan 14 berikut ini:



Gambar 13. Laporan sentimen dalam bentuk grafik pie



Gambar 14. Laporan sentimen dalam bentuk grafik batang

**3.3.6. Pelatihan dan Pengujian Sistem**

Tiga buah lapisan klasifikasi yang dibuat, dilatih dengan menggunakan data tweets yang diperoleh dari hasil pengambilan data dari situs twitter. Untuk keperluan pelatihan ini, data tweets dianotasi secara manual. Data tweets yang digunakan untuk pelatihan ini seperti yang disajikan dalam Tabel 10, 11, dan 12 berikut ini:

Tabel 10. Anotasi manual klasifikasi promo/n-promo

Tipe Anotasi	Jumlah Data
Promo	4.350
N-Promo	4.350

Tabel 11. Anotasi manual subyektivitas

Tipe Anotasi	Jumlah Data
Subyektif	3.068
Obyektif	1.282

Tabel 12. Anotasi manual sentimen

Tipe Anotasi	Jumlah Data
Positif	1.169
Negatif	953
Netral	946

Setiap lapisan dilatih dan diuji dengan data seperti yang ada di Tabel 10, 11, dan 12 dengan komposisi perbandingan untuk data latih dan data uji 25%:75%, 50%:50%, dan 75%:25%. Untuk masing-masing komposisi dilakukan pengukuran sebanyak 10 kali dengan pengambilan data acak untuk masing-masing data latih dan data uji.

#### 4. Hasil dan Pembahasan

##### 4.1. Penggunaan Kerangka Ontologi

Dari 14.437 *tweets* yang diperoleh dari *twitter* diteliti dengan menggunakan sejumlah perintah *query* untuk mengetahui jumlah *tweets* yang diluar domain *smartphone*. Dari hasil pengecekan, ditemukan ada 13 *tweets* yang diluar domain *smartphone*. *Tweets* tersebut mengandung keyword yang benar (*galaxy camera, galaxy screen*) tetapi mengarah ke domain film (*The Guardian of The Galaxy*).

Kejadian ini memungkinkan sekali terjadi karena *tweets* adalah tulisan bebas dari pengguna. Untuk meminimalkan hasil pencarian ini bisa dengan menggunakan *keyword* yang lebih spesifik.

##### 4.2. Pengujian lapisan klasifikasi

Tiga buah lapisan klasifikasi di uji dengan metode seperti yang dijelaskan di bagian 3.3.6. Hasil rata-rata pengujian untuk masing-masing lapisan seperti yang disajikan dalam tabel 13, 14, dan 15 di bawah ini:

Tabel 13. Rata-rata pengujian klasifikasi Promo

Kelas	Precision	Recall	F-Measure	Accuracy
25_ori	0.9625	0.9557	0.9590	0.9591
25_buzz	0.9560	0.9626	0.9593	
50_ori	0.9693	0.9642	0.9667	0.9667
50_buzz	0.9644	0.9692	0.9668	
75_ori	0.9769	0.9688	0.9728	0.9728
75_buzz	0.9690	0.9769	0.9729	

Berdasarkan hasil pengujian yang disajikan di tabel 13, model yang dikembangkan di subsistem ini memiliki tingkat akurasi yang cukup tinggi. Tingkat akurasi yang dihasilkan dengan data pelatihan 25% menghasilkan nilai 0.9591 yang berarti sangat mendekati nilai 1. Begitu juga untuk data pelatihan 50% dan 75% dengan tingkat akurasi 0.9667 dan 0.9728, masing-masing memiliki tingkat akurasi yang cenderung naik jika data pelatihannya ditambah.

Untuk nilai *precision* dan *recall* memiliki kecenderungan yang sama dengan menghitung *harmonic mean (F-Measure)* dari kedua variabel tersebut. Untuk masing-masing kelas setiap komposisi uji (25%, 50%, 75%), nilai *precision* untuk kelas *tweetsori* memiliki kecenderungan lebih tinggi dari kelas *tweets buzz* walaupun dengan selisih yang kecil (berbeda di digit ketiga desimal).

Dengan nilai-nilai yang dihasilkan seperti yang tersaji dalam tabel 13, model untuk subsistem klasifikasi promo dan bukan promo memberikan tingkat keakuratan yang tinggi.

Tabel 14. Rata-rata pengujian lapisan subyektivitas

Kelas	Precision	Recall	F-Measure	Accuracy
25_subj	0.9312	0.8943	0.9123	0.8788
25_obj	0.7693	0.8418	0.8037	
50_subj	0.9343	0.9203	0.9273	0.8982
50_obj	0.8161	0.8451	0.8303	
75_subj	0.9385	0.9370	0.9377	0.9122
75_obj	0.8503	0.8530	0.8514	

Berdasarkan hasil uji yang tertera di tabel 14 di atas, model yang dikembangkan untuk subsistem ini memiliki tingkat akurasi yang tinggi seperti yang ada di kolom *accuracy*. Untuk kolom *precision* dan *recall* juga mendukung tingkat akurasi ini. Untuk masing-masing kelompok komposisi, kelas subyektif memiliki nilai *precision* yang lebih tinggi dibandingkan dengan kelas obyektif. Begitu juga untuk nilai *harmonic mean (F-Measure)* dari masing-masing kelas di setiap komposisi ini.

Tingkat akurasi yang ada di subsistem ini memiliki nilai lebih rendah dibandingkan dengan tingkat akurasi yang ada di subsistem sebelumnya. Penyebabnya adalah tingkat kerumitan untuk memisahkan antara *tweets* yang subyektif dan *tweets*

yang obyektif lebih tinggi dibandingkan pemisahan pada subsistem sebelumnya.

Tabel 15. Rata-rata pengujian lapisan sentimen

Kelas	Precision	Recall	F-Measure	Accuracy
25_pos	0.8134	0.6854	0.7435	
25_neg	0.5666	0.8565	0.6815	0.6868
25_neu	0.7681	0.5175	0.6169	
50_pos	0.8368	0.7126	0.7695	
50_neg	0.5954	0.8962	0.7153	0.7197
50_neu	0.8194	0.5503	0.6577	
75_pos	0.8565	0.7328	0.7896	
75_neg	0.6142	0.9151	0.7346	0.7411
75_neu	0.8454	0.5759	0.6839	

Berdasarkan hasil uji yang disajikan di tabel 15, tingkat akurasi model klasifikasi cukup tinggi dengan maksimal nilai 0.7411. Dari tabel tersebut model ini cukup bisa mengenali *tweets* dengan sentimen positif dan netral dibandingkan dengan *tweets* yang memiliki sentimen negatif. Hal ini dibuktikan dengan nilai *precision* yang lebih tinggi untuk *tweets* positif dan netral dibandingkan dengan *tweets* negatif.

Tingkat akurasi model di subsistem ini lebih rendah dibandingkan dengan dua model sebelumnya. Hal ini karena tingkat kerumitan pemisahan untuk setiap kategori lebih tinggi dibandingkan dengan dua subsistem sebelumnya dan jumlah kelas dalam model ini lebih banyak dari dua model sebelumnya. Jumlah kelas yang lebih banyak ini juga akan menambah kerumitan pemisahan di model ini.

Rata-rata pengujian dari semua lapisan klasifikasi ini disajikan dalam tabel 16 berikut ini:

Tabel 16. Rata-rata pengujian keseluruhan sistem

Kompo- sisi	Sub- sistem Promo	Sub- sistem Subyek- tivitas	Sub- sistem Senti- men	Rata- rata	%
25:75	0.9591	0.8788	0.6868	0.8416	84.16%
50:50	0.9667	0.8982	0.7197	0.8615	86.15%
75:25	0.9728	0.9122	0.7411	0.8754	87.54%

## 5. Kesimpulan

Penggunaan kerangka ontologi untuk pencarian data di *twitter* terbukti mampu mempertajam hasilnya. Dari 14.437 *tweets* yang diteliti ditemukan 13 *tweets* yang tidak terkait dengan domain pembahasan. data ini tentu tidak terlalu signifikan dari total hasil pencarian.

Penggunaan algoritma klasifikasi *naïve bayes* terbukti mampu untuk mengklasifikasi *tweets* dengan tingkat akurasi yang tinggi. Hal ini dibuktikan dengan hasil rata-rata uji di masing-masing lapisan klasifikasi yang disajikan di tabel 13, 14, dan 15. Juga untuk rata-rata keseluruhan sistem yang

disajikan di tabel 16. Hasil akurasi yang bisa dicapai berada pada kisaran 84.16% sampai dengan 87.54%.

Hal-hal yang perlu diperhatikan dalam analisis ini, seperti: penggunaan kata kunci yang spesifik supaya bisa mendapatkan hasil pencarian yang sesuai dengan domain pembahasan, anotasi manual untuk pelatihan dan pengujian sistem, dan pra-pemrosesan data *tweets* (*cleaning*) juga ikut mempengaruhi hasil klasifikasi.

## Daftar Pustaka

- Abrahamsson, P., Salo, O., Ronkainen, J. dan Warsta, J., 2002. Agile software development methods: Review and analysis, VTT Publication.
- Chen, L., Wang, F., Qi, L., dan Liang, F., 2014. Experiment on sentiment embedded comparison interface, *Knowledge-Based Systems*, 64, 44-58.
- Ghiassi, M., Skinner, J., dan Zimbra, D., 2013. Twitter brand sentiment analysis: A hybrid system using n-gram analysis and dynamic artificial neural network, *Expert Systems with Applications: An International Journal*, 40(16), 6266-6282.
- Haddi, E., Liu, X., dan Shi, Y., 2013. The role of text pre-processing in sentiment analysis, *Procedia Computer Science*, 17, 26-32.
- He, Y., dan Zhou, D., 2011. Self-training from labeled features for sentiment analysis, *Information Processing & Management*, 47 (4), 606-616.
- Kontopoulos, E., Berberidis, C., Dergiades, T., dan Bassiliades, N., 2013. Ontology-based sentiment analysis of twitter posts, *Expert systems with applications*, 40(10), 4065-4074.
- Liu, B., 2011. *Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data, Second Edition*, Springer.
- Liu, B., 2010. *Sentimen Analysis and Subjectivity In: Indurkha, N., Damerau, F. J. (Eds.), HANDBOOK OF NATURAL LANGUAGE PROCESSING, Second Edition*. Boca Raton, FL, 627-666.
- Maks, I., dan Vossen, P., 2012. A lexicon model for deep sentiment analysis and opinion mining applications, *Decision Support Systems*, 53(4), 680-688.
- Manning, C.D., Raghavan, P., dan Schütze, H., 2008. *Introduction to information retrieval*, Cambridge: Cambridge university press.
- Medhat, W., Hassan, A., dan Korashy, H., 2014. Sentiment analysis algorithms and applications: A survey, *Ain Shams Engineering Journal*, 5, 1093-1113
- Mukherjee, S., dan Sharma, N., 2012. Intrusion detection using naive Bayes classifier with feature reduction, *Procedia Technology*, 4, 119-128.
- Peñalver-Martinez, I., Garcia-Sanchez, F., Valencia-Garcia, R., Rodríguez-García, M. Á., Moreno, V.,

- Fraga, A., dan Sánchez-Cervantes, J. L., 2014. Feature-based opinion mining through ontologies, *Expert Systems with Applications*, 41(13), 5995-6008.
- Taboada, M., Brooke, J., Tofiloski, M., Voll, K., dan Stede, M., 2011. Lexicon-based methods for sentiment analysis, *Computational linguistics*, 37(2), 267-307.
- Toutanova, K., Klein, D., Manning, C. D., dan Singer, Y., 2003. Feature-rich part-of-speech tagging with a cyclic dependency network, *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, vol 1, 173-180.
- Unnamalai, K., 2012. Sentiment analysis of products using web, *Procedia Engineering*, 38, 2257-2262.
- Wicaksono, A. F., dan Purwarianti, A., 2010. HMM based part-of-speech tagger for Bahasa Indonesia, *In Fourth International MALINDO Workshop*, Jakarta.
- Zhang, H., 2004. The optimality of naive Bayes, *Proceedings of the Seventeenth Florida Artificial Intelligence Research Society Conference*, Florida, 562-567.
- Zhang, L., Ghosh, R., Dekhil, M., Hsu, M., dan Liu, B., 2011. Combining lexicon-based and learning-based methods for Twitter sentiment analysis, *HP Laboratories, Technical Report HPL-2011*, 89.
- Zhou, S., Chen, Q., dan Wang, X., 2014. Fuzzy deep belief networks for semi-supervised sentiment classification, *Neurocomputing*, 131, 312-322.