BAB 1
PENDAHULUAN

1.1 Latar Belakang

Analisa sentimen atau *opinion mining* merupakan proses memahami, mengekstrak dan mengolah data teks tual secara otomatis untuk mendapatkan informasi sentimen yang terkandung dalam suatu kalimat opini. Analisa sentimen dilakukan untuk melihat pendapat atau kecenderungan opini terhadap sebuah masalah atau objek oleh seseorang, apakah cenderung berpandangan atau beropini negatif atau positif. [1]

Media jejaring sosial seperti *Twitter*, *Facebook*, dan *Youtube* merupakan media sosial terpopuler di masyarakat saat ini. Media sosial merupakan sebuah media *online* yang memungkinkan para pengguna yang dapat berinteraksi satu sama lain. Salah satu media sosial adalah *microblog*. *Microblog* adalah suatu bentuk blog (aplikasi *website*) yang memungkinkan penggunanya untuk menulis teks singkat, umumnya kurang dari 200 karakter dan mempublikasikannya, baik untuk dilihat semua orang atau kelompok terbatas yang dipilih oleh pengguna tersebut [2]. Salah satu *microblogging* populer saat ini adalah *Twitter*. Data yang terdapat pada *Twitter* dapat dimanfaatkan untuk memperoleh informasi tertentu, karena pengguna dapat mengungkapkan apa pun melalui *Twitter*, baik itu apa yang dia rasakan, apa yang sedang dialami ataupun kondisi yang terjadi di sekitarnya.

Permasalahan dalam analisa sentimen pada media sosial di Indonesia adalah ditemukan banyak kata tidak baku, singkatan dan penggantian huruf vokal menjadi angka. Sebagai contoh, sebuah kalimat "J4n64n 4d4 du5t4" membuat algoritma analisa sentimen menjadi tidak dapat memberikan hasil yang benar karena akan dianggap sebagai kata yang tidak memiliki makna pada tahap *preprocessing* teks. Untuk hal semacam ini diperlukan sebuah algoritma pencarian padanan kata tidak baku ke dalam kata baku yang sesuai dengan arti kata tersebut (semantik) dan sebuah kamus tidak baku.

Semantic search adalah metode pencarian untuk menampilkan hasil pencarian yang relevan dengan apa yang orang inginkan atau orang cari. *Semantic search* berusaha untuk meningkatkan akurasi pencarian dengan memahami maksud pencarian berdasarkan makna kontekstual sehingga menghasilkan hasil yang lebih relevan.

Dalam pendekatan *semantic search*, elemen istilah (*term)* dideskripsikan sebagai *atomic concept* (kata dengan satu makna), atau *complex concept* (banyak makna), dan *match* (pencocokan) dideskripsikan sebagai *matching semantics* dari *concept* dan bukan kata seperti pada pendekatan sintaksis. Pencarian kata tidak hanya berdasarkan kata yang sama tetapi juga pencarian berdasarkan makna dari kata tersebut. Sebagai contoh, kita melakukan pencarian kata dengan *query* (kalimat pertanyaan) "harga kijang terbaru *matic*" maka akan keluar hasil pencarian harga mobil kijang terbaru meski kita tidak menuliskan kata mobil pada *query* yang kita ketik. Hasil ini merupakan gabungan dan pencocokan kata kijang dan *matic* yang didefinisikan sebagai mobil.

Ide utama dari pendekatan *semantic search* berasal dari pandangan kognitif terhadap dunia, yaitu bahwa arti dari suatu teks (kata) bergantung kepada relasi kontekstual terhadap obyek dalam dunia nyata dari pada relasi linguistik yang terdapat dalam teks atau kamus. Komponen penting dalam model ini adalah keberadaan struktur *concept* atau makna dari kalimat memetakan deskripsi objek informasi dengan *concept* yang terdapat dalam *query* [3]. Salah satu contoh sebuah *query* yang diketikan "Ford films"

maka *semantic search* akan mencari juga pada artikel data yang mengandung kata *movie* karena *films* memiliki makna yang sama dengan *movie*.

Struktur *concept* dapat bersifat umum atau domain spesifik dan dapat dibuat dengan pendekatan manual atau otomatis. Tipe utama dari struktur *concept* yang digunakan dalam pendekatan *semantic search* antara lain struktur *taxonomy* (pengkategorian), thesauri (kamus sinonim atau alat kontrol kosa kata) dan ontologi (pengertian dasar dan relasi antar kata) [4]. *Semantic search* pada intinya merupakan pencarian yang berusaha meningkatkan akurasi dengan menghilangkan ambiguitas melalui pemahaman konteks [5]. *Semantic search* biasa digunakan dalam pembuatan peringkat halaman (*page rank)* dari sebuah *website* pencarian, khususnya Google, yang menggunakan algoritma *Latent Semantic Indexing* (LSI) dalam mengidentifikasi kata kunci (*keyword*) pada halaman *website* dan sinonimnya [6].

Dalam penelitian ini dibutuhkan algoritma yang dapat mengenali suatu istilah (*term*) yang berupa sinonim atau *term* yang memiliki banyak makna (polisemi). Algoritma LSI memiliki kemampuan mengenali sinonim dan polisemi. Algoritma LSI melihat *term* berdasarkan nilai semantik bukan berdasarkan sintaksis. Penggunaan teknik matematika yakni *Singular Value Decomposition* (SVD) pada LSI untuk mendekomposisikan matriks *term*-dokumen, dengan mengurangi ruang *term* dan dokumen menjadi dimensi yang lebih kecil menampakkan hubungan yang mendasari *term* dan dokumen dalam semua kombinasi yang memungkinkan dan membuang *noise* yang ada pada ruang vektor [7]. Kekurangan algoritma LSI adalah tidak dapat mengenali salah pengetikan dan pengoreksian kata.

Algoritma Damerau-Levenshtein *distance* (DLD) memiliki kemampuan memperbaiki kata berdasarkan empat macam kesalahan pengetikan yaitu, penyisipan sebuah huruf, penghapusan huruf, penggantian huruf dengan huruf lain, dan penukaran urutan huruf. Kekurangan algoritma DLD adalah hanya berfokus pada pengoreksian kata secara

sintaksis, tidak memperhatikan keterkaitan makna (semantik). Penggabungan algoritma LSI dan DLD diharapkan dapat menghasilkan algoritma baru yang lebih baik.

Agar kata-kata tidak baku yang telah berhasil dikenali oleh algoritma LSI-DLD maupun kata yang tidak dapat dikenali otomatis tetapi dapat dikenali secara manual tidak perlu diproses kembali di masa depan, maka kata yang sudah dikenali disimpan ke dalam kamus bahasa indonesia kata tidak resmi.

1.2 Premis dan Hipotesis

1.2.1 Premis

1. Algoritma LSI dapat digunakan untuk mengenali makna suatu kata berdasarkan relasi semantik, namun tidak dapat mengenali kata yang tidak baku.
2. Algoritma LSI dengan penggunaan *knowledge base* (kamus) memperbaiki skor similaritas antar kalimat.
3. Algoritma DLD dapat mengenali dan memperbaiki kata yang tidak baku akibat penulisan yang tidak benar.

1.2.2 Hipotesis

1. Pembangunan kamus Bahasa Indonesia Kata Tidak Baku dapat dilakukan dengan penggunaan algoritma LSI dan DLD.
2. Penggunaan kamus Bahasa Indonesia Kata Tidak Baku dapat meningkatkan efektivitas proses analisa sentimen terhadap pendapat publik di media sosial.

1.3 Batasan Masalah

Batasan masalah penelitian ini adalah:

1. Data latih merupakan data artikel dari *website* tempo.

2. Data uji merupakan data *tweet* untuk produk Telkomsel. Hasil pengujian disampaikan di ***Lampiran 3*.**

1.4 Tujuan dan Manfaat Penelitian

1.4.1 Tujuan

Tujuan penelitian ini adalah membuat modul *preprocessing* teks yang mengenali kata tidak baku dan membuat relasinya dengan kata baku dalam pembuatan kamus bahasa Indonesia kata tidak baku.

1.4.2 Manfaat penelitian

Manfaat penelitian dalam Pembangunan kamus bahasa Indonesia kata tidak baku dengan Algoritma *Latent Semantic Indexing* dan *Damerau-levenshtein Distance*:

1. Manfaat praktis, membangun kamus bahasa Indonesia Kata Tidak Baku yang dapat digunakan untuk analisa sentimen kalimat-kalimat berbahasa Indonesia dari media sosial.
2. Manfaat teknis menghasilkan modul *preprocessing* teks dengan algoritma baru yang dapat mendeteksi kata tidak baku.

1.5 Kasus

Pada penelitian ini kasus yang dipilih sebagai data adalah kalimat dari *microblog twitter* tentang produk Telkomsel selama bulan mei-juni 2017.

1.6 Sistematika Penulisan

Tesis ini terdiri dari lima bab, dengan isi masing-masing bab sebagai berikut :

1. Bab 1 Pendahuluan. Bab ini berisi latar belakang, identifikasi masalah, tujuan dan manfaat penelitian, pembatasan masalah dan asumsi serta sistematika penulisan.
2. Bab 2 Landasan Teori. Bab ini berisi berbagai konsep dan teori sentimen analisis, semantik, *information retrieval, Natural Language Processing, text mining, stemming, word shrinking, Latent semantic Indexing* serta *Damerau-Levenshtein distance*.
3. Bab 3 Analisis dan Perancangan. Bab ini berisi pendekatan penelitian yang digunakan. Bab ini menguraikan Analisis persoalan, data dan Teknik pengujian penelitian. Teknik yang digunakan dalam penelitian ini adalah *sampling*. Data uji diambil dari kata baku artikel website tempo yang dirubah menjadi kata tidak baku. Perubahan kata baku menjadi kata tidak baku dengan cara penghapusan huruf vokal, pertukaran posisi, penggantian menjadi angka, dan perubahan menjadi singkatan.
4. Bab 4 Implementasi dan pengujian . Bab ini berisi implementasi perancangan kedalam bahasa pemrograman python meliputi lingkungan implementasi, implementasi modul *crawler* dan implementasi analisis semantik. Pada bab ini juga dilakukan pengujian aplikasi.
5. Bab 5 Kesimpulan dan Saran. Berisi kesimpulan berdasarkan hasil pengujian dan saran untuk penelitian selanjutnya