PEMERIKSA TATA BAHASA: PEMILAHAN FRASE

Hera Suhartanto dan Syandra Sari
Fakultas Ilmu Komputer - Universitas Indonesia

Abstrak

Penyunting kata elektronis telah banyak beredar sejak munculnya perangkat komputer di masyarakat. Keberadaanya sangat membantu pemakai (*user*) dalam merancang dan memproduksi suatu dokumen. Sayangnya kebanyakan penyunting tersebut dirancang khusus untuk bahasa asing sehingga pemakaiannya dalam penyiapan dokumen berbahasa Indonesia kurang membantu. Untuk menutupi kekurangan tersebut beberapa pihak telah dan sedang mengembangkan perangkat lunak pendukung yang bisa disisipkan pada penyunting yang sudah ada. Perangkat pendukung tersebut antara lain adalah Kamus Besar Bahasa Indonesia (KBBI) elektronik [6]. Fasilitas Pemenggalan Kata Indonesia [0], Fasilitas Pemeriksa Ejaan[3], dan Fasilitas Tesaurus [10]. Dalam makalah ini penulis akan menjelaskan prototipe yang tengah dikembangkan sebagai Pemeriksa Tata Bahasa Indonesia. Pemeriksa ini diharapkan mampu memeriksa validitas suatu kalimat bahasa Indonesia sesuai dengan Tata Bahasa Baku Bahasa Indonesia.

1. Pendahuluan

Sejak ditemukannya teknologi komputer pada awal tahun 1940-an, banyak paket program aplikasi sangat membantu para pemakainya yang berkecimpung dalam beberapa bidang yang bervariasi, antara lain perancangan badan pesawat dan mobil, pemrosesan data untuk prakiraan cuaca, penaksiran gerakan cairan di bawah tanah, penaksiran kuantitas unsur pencemar di dalam air tanah, pemodelan sirkuit listrik, pemodelan reaksi kimia, penerapan pada kegiatan perbankan, penerapan pada kegiatan kedokteran, dan aplikasi pada banyak bidang industri, penelitian dan pendidikan. Paket program aplikasi tersebut hanya dipakai oleh kalangan tertentu, dimana paket program yang dibuat untuk aplikasi kedokteran tidaklah relevan untuk aplikasi kedirgantaraan. Berbeda dengan paket tersebut, paket program penyunting naskah (Word Processor) dipakai oleh banyak orang di segala bidang. Untuk penyingkatan penulis memakai istilah penyunting sebagai paket program penyunting naskah.

Ada beberapa penyunting yang dipakai terbatas untuk suatu negara atau kelompok negara yang mempunyai aksara bukan latin, namun paket penyunting beraksara latin lebih mendominasi pasaran pemakai di dunia. Di antara paket penyunting beraksara latin yang telah kita kenal antara lain Wordstar, Wordperfect, MS Word, Chi-Writer, dan Amipro. Karena pembuat paket tersebut adalah orang yang berbahasa asing, kebanyakan berbahasa Inggris, maka kebanyakan paket tersebut hanya cocok untuk naskah yang ditulis dalam bahasa Inggris. Memang beberapa paket penyunting menyediakan fasilitas konversi ke bahasa tertentu misalnya bahasa Jerman, Yunani, atau Rusia, namun fasilitas konversi tersebut hanya terbatas pada tampilan karakter latin ke karakter negara-negara tersebut. Sedangkan fasilitas yang berhubungan langsung dengan bahasa negara tertentu, misalnya pemeriksa ejaan, pemenggalan kata, tesaurus, dan pemeriksa tatabahasa belum didukung oleh paket-paket tersebut.

Memang beberapa waktu yang lalu mulai terdengar pengindonesiaan paket-paket program, misalnya seperti MS DOS versi Indonesia dan Wordstar versi Indonesia. Tetapi proses pengindonesiaan paket tersebut baru pada tahap penerjemahan perintah-perintah dan pesan-pesan paket tersebut ke bahasa Indonesia. Untuk melengkapi perangkat penyunting yang sudah ada, dalam makalah ini penulis akan menjelaskan prototipe Pemeriksa Tata Bahasa Indonesia. Prototipe ini baru diterapkan pada kalimat tunggal. Peneliti di BPPT telah mengadakan penelitian dan pengembangan sistem penerjemah bahasa Indonesia ke bahasa Jepang, penulis memperkirakan bahwa pemeriksaan Tata Bahasanya sudah dilakukan sebelum suatu dokumen diterjemahkan ke bahasa sasaran. Namun pendekatan yang penulis lakukan adalah pendekatan pembuatan suatu bahasa pemrograman komputer.

Karena Tata Bahasa Baku Bahasa Indonesia (TB3I) dijadikan sebagai sumber acuan maka pada bagian 2 akan disajikan secara ringkas Tata Bahasa tersebut. Kemudian pada bagian 3 akan dijelaskan secara ringkas bentuk representasi Tata Bahasa yang bisa diaplikasikan dalam program komputer. Untuk mempermudah pembuatan dan modifikasi prototipe ini akan dipakai perangkat pembantu yakni Lex dan Yacc, kedua perangkat ini akan dijelaskan pada bagian 4 beserta terapan sebagian TB3I. Bagian 5 akan memuat contoh masukan (input) dan keluaran (output) yang

dirancang, kemudian akan dijelaskan rencana pengembangan selanjutnya pada bagian 6.

2. Tata Bahasa Baku Bahasa Indonesia (TB3I)

Karena yang diproses adalah dokumen berbahasa Indonesia maka perlu diterapkan aturan-aturan baku bagaimana suatu kalimat bahasa Indonesia terbentuk pada kerangka utama program prototipe. Kalimat dapat dibagi menurut bentuk dan maknanya (nilai komunikatifnya) [8]. Pada prototipe ini penulis hanya memakai bentuk kalimat karena dari bentuk ini bisa dilihat bagaimana suatu kalimat dibuat. Prototipe baru menerapkan bentuk kalimat tunggal sedangkan kalimat majemuk baru akan dikembangkan di kemudian hari. Bentuk kalimat tunggal yang diterapkan adalah kalimat tunggal berpredikat Frasa Nominal, Frasa Adjektival, Frasa Verbal dan Frasa Lain. Rincian masing-masing kalimat ini bisa dilihat pada daftar acuan [8].

3. Representasi TB3I (*Context Free Grammars*)

Untuk memeriksa kebenaran kumpulan kalimat dalam suatu berkas (*file*) masukan menurut tata bahasa tertentu maka unsur terkecil dari berkas tersebut , yakni karakter, akan dibaca dan dikelompokkan berdasarkan aturan pembentukan unsur kalimat. Penulis mendefinisikan unsur kalimat sebagai tanda baca dan kata. Jadi dari suatu kalimat maka akan dipilah (*parse*) satu persatu suatu kata dan tanda baca. Pengambilan unsur ini dilakukan dari kiri ke kanan . Kumpulan terkecil karakter yang membentuk unsur tersebut disebut **leksim** (*lexim*). Dari kumpulan terkecil tersebut akan dikelompokkan menjadi beberapa subkelompok (jenis) yang mempunyai arti tertentu. Subkelompok ini disebut **token**. Sebagai contoh misalkan kalimat yang akan diproses adalah

Pukul 17:00 kemarin, para pemeriksa memberi Oki istirahat untuk menyantap mie instant dan segelas air putih pesanannya.
[Republika, 13 Januari 1995]

maka kata-kata, misalnya *pemeriksa, memberi, Oki* dan *mie* adalah leksim, sedangkan kata-kata tersebut dapat dikelompokkan berturut-turut sebagai token SUBJEK, VERBA, NAMAORANG dan KATABENDA.

Agar suatu kalimat dapat diperiksa kebenarannya maka diperlukan suatu cara penulisan TB3I dalam bentuk yang mudah untuk dibuatkan programnya. Dalam teori bahasa pemrograman komputer bentuk penulisan ini sering juga disebut Tata Bahasa Bebas-Konteks (*Context-Free Grammar*) (TBBK). TBBK terdiri dari terminal, nonterminal, simbol awal, dan kumpulan produksi. Berikut adalah definisi dari masing-masing unsur TBBK.

Terminal adalah simbol dasar TBBK. Kumpulan dari terminal akan membentuk suatu **untaian** simbol dasar (*string*). Dengan demikian maka pengertian token sama dengan terminal.

Nonterminal adalah suatu perubah (*variables*) sintak yang menotasikan kumpulan atau himpunan untaian.

Dalam suatu TBBK, satu nonterminal berfungsi sebagai **simbol awal.** Dan kumpulan untaian yang merepresentasikan oleh simbol ini adalah bahasa yang dibentuk oleh TBBK.

Produksi pada TBBK menentukan cara dimana terminal dan nonterminal dapat digabungkan untuk membentuk suatu untaian-untaian. Masing-masing produksi terdiri dari nonterminal , diikuti oleh suatu tanda panah (kadang-kadang simbol titik dua :), diikuti oleh untaian atau kumpulan nonterminal dan terminal.

Dengan demikian dokumen yang ditulis berdasarkan TB3I, sebagian, dapat dituliskan dalam bentuk TBBK seperti berikut

```
berkas    :    AKHIR
           |   paragraps paragrap AKHIR

paragraps :
           |   paragraps paragrap

paragrap  :    kalimats kalimat TITIK ANPAR

kalimats  :
           |   kalimats kalimat TITIK
```

```
kalimat  :  subjek PredikatFrasaNominal
         |  subjek PredikatFrasaAdjektiva nomina
         |  subjek PredikatFrasaVerba nomina
         |  subjek PredikatFrasaLain

subjek   : nomina | frasaNomina | pronominaPersona | frasaPronomial
```

Pada contoh di atas maka simbol yang merupakan terminal adalah AKHIR, TITIK, dan ANPAR sedangkan simbol-simbol lainnya adalah nonterminal. Pada penerapan yang sebenarnya pendefinisian masing-masing nonterminal harus berakhir pada suatu terminal, misalkan **nomina** bisa didefiniskan sebagai **nomina** : NAMAORANG, dimana NAMAORANG adalah terminal.

Pada contoh TBBK di atas dapat kita lihat bahwa suatu berkas bisa kosong hanya berisi akhir suatu berkas komputer, end-of-file (EOF). Di sini akhir berkas didefiniksan oleh token AKHIR. Suatu berkas terdiri dari satu atau lebih paragrap. Dalam berkas yang akan diproses diasumsikan bahwa setiap paragrap dipisahkan oleh satu spasi kosong. Spasi kosong ini dalam bahasa pemrograman, misalnya C, bisa dikenali dengan adanya dua karakter baris baru (*newline*) '\n'. Dua karakter baris baru ini dapat direpresentasikan oleh token atau terminal ANPAR. Kemudian setiap paragrap terdiri dari satu atau lebih kalimat, dan setiap kalimat diakhiri oleh karakter titik. Dalam penerapan karakter titik dipresentasikan oleh token TITIK. Dan yang terakhir, kalimat didefinisikan oleh unsur-unsur kalimat.

4. Perangkat Pembantu : Lex [5] dan Yacc [4].

Salah satu kemudahan penulisan TBBK dalam bentuk TBBK adalah tersedianya perangkat lunak pembantu (*tools*) yang dapat menerima suatu aturan dalam bentuk TBBK kemudian menghasilkan suatu keluaran berbentuk program dalam bahasa pemrograman tertentu. Perangkat ini bernama Lex dan Yacc dan telah tersedia versi yang bisa berjalan di komputer pribadi memakai DOS atau komputer memakai sistem pengoperasi (*operating system*) UNIX.

Lex berperan membaca sekumpulan karakter yang sesuai dengan aturan , biasanya disebut **ekspresi beraturan** (*regular expression*), lalu

menjalankan *aksi* yang didefinisikan oleh pemakai. Salah satu contoh ekspresi beraturan ditulis dalam bentuk [a-zA-Z]+, ekspresi ini akan mengenal kumpulan karakter yang terdiri dari paling sedikit satu huruf alfabet besar atau kecil. Maka ekspresi itu bisa dipakai untuk mengenali pembacaan suatu kata. Setelah satu kata terbaca, maka kata ini akan diperiksa apakah ada dalam kamus yang dipakai, misalnya KBBI elektronis, jika ada maka jenis (token) kata tunggal itu diberikan ke Yacc.Pemeriksaan keberadaan kata dalam kamus dan pengambilan jenis (KBBI memakai istilah kelas) kata itu dilakukan oleh aksi yang diletakkan di sebelah kanan ekspresi regular. Dalam penerapan aksi itu ditulis seperti berikut

```
[a-zA-Z]+        return token(lihatKamus(yytext));
```

lihatKamus(yytext) berupa suatu bagian program yang memeriksa keberadaan kata ,yang tersimpan pada peubah (variable) yytext, dalam kamus. Jika kata tidak ada dalam kamus maka bagian program itu akan memberikan token ILEGAL kepada Yacc tetapi jika kata ada dalam kamus Yacc akan memperoleh token jenis atau kelompok kata itu. Kemudian Yacc akan memeriksanya apakah token tersebut sama dengan yang ada di tempatnya. Jika tidak sama maka Yacc akan mengeluarkan pesan *Syntax Error*. Jika sama maka Yacc akan meminta token berikutnya dari Lex dan seterusnya sampai akhir berkas ditemukan.

Program keluaran dari Lex berbahasa pemrograman C biasanya bernama lex.yy.c sedangkan program keluaran Yacc bernama y.tab.c, dan masing-masing token didefinisikan pada berkas y.tab.h. Dengan mengasumsikan bahwa pemrograman dilakukan di bawah sistem pengoperasi UNIX, maka ketiga berkas keluaran itu bisa dikompilasi *cc lex.yy.c y.tab.c y.tab.h -ll* dan dihasilkan berkas a.out. Berkas terakhir ini bisa memroses dokumen berbahasa Indonesia dengan perintah *a.out < namaBerkas*.

5. Masukan dan Keluaran hasil proses

Berkas yang berisi kalimat berikut, tanpa kata yang tercetak miring, diproses.

1 Ketegangan justru terlihat pada petugas reserse yang *berupaya* keras
2 mengorek keterangan dari pemuda berpostur tegap ini. Dua reserse

3 dari *Divisi Bunuh-Culik* terlihat mondar-mandir keluar masuk ruang
4 pemeriksaan, dengan wajah tegang, sembari membawa berkas hasil
5 pemeriksaan.
6
7 Pemeriksaan terhadap Oki *berlangsung* marathon sejak pertama kali
8 dia tertangkap, Sabtu lalu.

dan keluaran akan mengeluarkan pesan

baris 1, syntax error dekat kata "keras".
baris 3, syntax error dekat kata "terlihat".
baris 7, syntax error dekat kata "marathon".

Kesalahan pertama terjadi karena pada baris pertama pada saat yytex berisi untaian "keras", Yacc sedang mengharapkan kehadiran suatu verba. Namun yang ditemui adalah adverbia "keras". Kesalahan kedua terjadi karena Yacc mengharapkan kehadiran suatu kata berjenis keterangan tempat namun yang diterima adalah verba "terlihat". Dan kesalahan ketiga terjadi karena Yacc sedang mengharapkan kehadiran suatu verba namun yang diterima adalah adverbia.

6. Penutup

Dalam waktu dekat, prototip ini akan dikembangkan sehingga bisa memroses berkas yang memuat kalimat majemuk. Namun penambahan terminal dan nonterminal pada TBBKnya akan membutuhkan ruang memori yang lebih besar sehingga patut dicarikan penulisan TBBK yang lebih padat tetapi bisa merepresentasikan seluruh TB3I

Sistem pemakaian kamus akan dikembangkan mengingat ketidakmungkinan pengisian seluruh isi KBBI elektronis ke dalam ruang memori apa lagi jika pemakai hanya menggunakan komputer pribadi (PC). Prototipe akan dimodifikasi sehingga bisa berjalan di atas PC tanpa memerlukan persyaratan perangkat keras tambahan.

Direncanakan juga akan diberikan fasilitas pengisian kata yang tak ada secara interaktif. Pemakai bisa memasukkan kata baru ke dalam kamus dan mendefinisikan kelompoknya. Fasilitas ini memungkinkan

terbentuknya kamus lokal yang memuat kata-kata yang tidak ada dalam KBBI elektronis.

Pengembangan yang sangat menarik adalah penambahan fasilitas untuk melihat gaya penulisan suatu berkas. Paling tidak gaya ini bisa dilihat dari banyaknya jenis kalimat yang dipakai oleh si penulis. Begitu juga dengan kalimat yang terlalu banyak katanya sehingga menyulitkan si pembaca. Kasus terakhir ini bisa dideteksi dengan menghitung jumlah kata dalam suatu kalimat dan menampilkannya ke layar komputer jika ternyata jumlah itu terlalu besar.

Daftar Acuan

[0] Atik Wintarti, *Fasilitas Pemenggalan Kata Indonesia*, Skripsi S2, Fasilkom-UI, Depok, 1995

[1] Benny Nugroho, *Pembangkitan Kata Turunan dari Kata Dasar Bahasa Indonesia*, Tugas Akhir S2, Fasilkom-UI, 1994.

[2] BPP Teknologi, *Indonesian Master Dictionary Specification*, Version 2.0, Jakarta, 1992.

[3] Bobby Nazief , *Fasilitas Pemeriksa Ejaan dan Analisis Statistik Kata*, proyek intern, Pusilkom-UI, 1994.

[4] Johnson, S., *Yacc: Yet Another Compiler-Compiler*, Bell Laboratories, New Jersey, 1978.

[5] Lesk, M. E., and Schmidt, E., *Lex - A Lexical Analyzer Generator*, Bell Laboratories, New Jersey, 1978.

[6] Mirna Adriani, Bobby Nazief dan Fanny Santosa, *Pengembangan Kamus Elektronis Bahasa Indonesia*, proyek intern, Pusilkom-UI, 1993.

[7] Syandra Sari, *Prototipe Pemeriksa Tata Bahasa Indonesia*, Skripsi S1, Fasilkom-UI, Depok, 1995.