

# Pemrosesan Bahasa Alami Pada Data Twitter Untuk Penyajian Informasi Jalan dan Lalu Lintas

Dyah Retno Utari<sup>1</sup>, Arief Wibowo<sup>2</sup>  
Fakultas Teknologi Informasi  
Universitas Budi Luhur  
dyah.retnoutari@budiluhur.ac.id.\*

**Abstrak.** Peningkatan pengguna media sosial di Indonesia mendatangkan tantangan dan peluang dalam bidang pemrosesan bahasa alami. Sebagaimana diketahui, di era *big data*, derasnya arus informasi dari media sosial di Indonesia seharusnya dapat dimanfaatkan untuk hal yang positif. Salah satu tantangan yang hadir dalam pemrosesan bahasa alami adalah bagaimana pengolahan sumber data media sosial dapat menjadi proses yang menghasilkan suatu aplikasi pengolah bahasa alami yang bermanfaat. Beberapa komponen dalam pemrosesan bahasa alami adalah penggunaan *scanner*, *parser*, *translator* dan *evaluator*. Penelitian ini menggunakan metode Pemrosesan Bahasa Alami berbasis XML untuk menangkap pesan tentang kebutuhan informasi lalu lintas. Data yang dianalisis merupakan hasil pemrosesan bahasa alami dari media sosial *Twitter*. Hasil penelitian menghasilkan suatu aplikasi pemrosesan bahasa alami yang mampu menyajikan informasi kondisi lalu lintas sebagaimana hasil pengujian berdasarkan data video <http://lewatmana.com>. Hasil pengujian menunjukkan bahwa pemrosesan bahasa alami dari data teks twitter, mampu menghasilkan kebutuhan informasi lalu lintas dengan akurasi mencapai 88,57%.

**Kata Kunci:** Pemrosesan Bahasa Alami, Analisis Media Sosial, Twitter, Ekstraksi Informasi.

## 1 Pendahuluan

Penyebaran Informasi lalu lintas dalam media sosial networking seperti Twitter sangat membantu pengguna yang membutuhkan berita mengenai kondisi jalan dan lalu lintas. Berbagai pengguna Twitter baik perorangan maupun satuan yang mengatur lalu lintas dari pihak kepolisian (Direktorat Lalu lintas) memberikan informasi lalu lintas dan jalan raya.

Dengan banyaknya data tentang kondisi jalan dan informasi tersebut maka diperoleh data dalam jumlah besar yang dapat digali untuk mendapatkan pengetahuan yang lebih bermanfaat melalui teknik Data Mining. Dengan pemrosesan yang tepat maka didapatkan database kondisi jalan dan lalu lintas yang dapat dimanfaatkan untuk membangun aplikasi pemantau kondisi jalan dan lalu lintas.

Namun demikian, data dalam jumlah besar tersebut akan lebih mudah dimanfaatkan apabila diciptakan suatu aplikasi berbasis bahasa alami yang memudahkan pengguna akhir yang ingin menggali informasi dari basisdata yang terkumpul. Untuk itulah diperlukan suatu teknik pengaksesan informasi dari database yang telah diproses menggunakan pemrosesan bahasa alami atau NLP (*Natural Language Processing*), dengan kemudahan dalam penulisan perintah pencarian informasi.

Untuk alasan tersebut maka penelitian ini diperlukan, berisi kegiatan pemrosesan bahasa alami (*natural language processing*) pada database lalu lintas yang sudah diperoleh dari akun-akun twitter yang menginformasikan kondisi jalan dan lalu lintas. Dengan penelitian ini dibangun suatu *prototype* aplikasi pemrosesan bahasa alami untuk mengakses informasi kondisi jalan dan lalu lintas dari database yang sudah dihasilkan.

Telah dilakukan berbagai penelitian tentang sistem pengolah bahasa alami, terutama yang menggunakan bahasa Indonesia sebagai input bahasa alami dan menggunakan XML dalam format penyimpanan basis data. Beberapa penelitian yang pernah dilakukan antara lain:

a. Ema Utami dan Sri Hartati melakukan penelitian pemrosesan Bahasa Alami tentang alih teks bahasa Inggris ke bahasa Indonesia (translasi) menggunakan pendekatan metode rule based. Aplikasi yang dibangun mampu menganalisa pada tingkatan sintaksis dan semantik secara lebih mendalam sehingga sistem penerjemah yang dibuat bukan hanya kamus yang menerjemahkan kata per kata. Setiap kata bahasa Inggris yang diinput akan diproses melalui analisis leksikal (*scanner*), analisis sintaks (*parser*), analisis semantik (*translator*) dan analisis

pragmatik (evaluator) sebelum mendapatkan hasil akhir berupa teks dalam bahasa Indonesia. Fokus penelitian yang dilakukan adalah teknik pendekatan rule based dalam melakukan translasi bahasa Inggris ke dalam bahasa Indonesia [1].

b. Pada penelitian Eri Zuliarso dan Sri Hartati, penelitian tentang bahasa alami dilakukan dengan obyek pada database dalam format XML. Pemrosesan bahasa alami dilakukan pada basis data yang berisi record tentang pustaka/literatur seperti jurnal, buku dan bahan kepustakaan lainnya. Input teks berupa kalimat bahasa Indonesia untuk dijadikan bahasa Query untuk mengakses database kepustakaan. Teknik pemrosesan bahasa alami menggunakan tahapan umum seperti *scanner*, *parser*, *translator* dan evaluator. Namun demikian, setelah proses *scanner* dan *parser*, didapatkan daftar token yang diubah menjadi sintaks bahasa Query dan selanjutnya diproses ke evaluator [2].

c. Suprihadi melakukan penelitian tentang bahasa alami dalam peningkatan kualitas citra digital. Proses pengolahan bahasa alami dimulai dari proses pengguna memasukkan kalimat perintah dalam bahasa Indonesia, kemudian diproses oleh pengolah bahasa alami sehingga menghasilkan citra yang diinginkan pemakai. Pengolah bahasa alami juga dapat menghasilkan pesan kesalahan jika kalimat perintah yang dimasukkan tidak dapat diterima atau tidak sesuai dengan aturan produksi yang ditentukan [3].

d. Andayani dalam penelitiannya melakukan pemrosesan bahasa alami untuk menerjemahkan kalimat bahasa Indonesia menjadi Query dalam pencarian Basis Data akademik. Pemrosesan bahasa alami menggunakan elemen *Scanner*, *Parser*, *Translator* dan *Evaluator* [4].

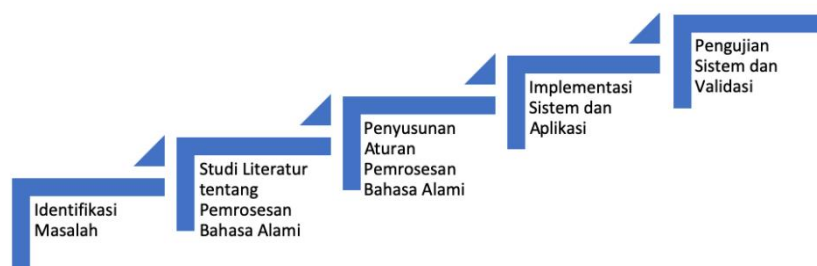
e. Hanif Al Fatta dalam penelitiannya menggunakan Natural Language Processing (NLP) dengan teknik *State Machine Parser*. Pendekatan bahasa alami yang dilakukan bertujuan untuk mengolah bahasa alami dalam konteks pemeriksaan struktur kalimat dalam bahasa Inggris. Proses dimulai dengan pemeriksaan kata atau token yang dimasukkan dan memastikan bahwa antar token yang dihasilkan memiliki (Grammar) struktur/tata kalimat bahasa Inggris yang baik. Penelitian dilakukan pada struktur Grammar sederhana yang diujikan dengan teknik State Machine [5].

f. Nanang Prasetyantara telah melakukan penelitian tentang pengolahan bahasa alami dari twitter untuk membangun basis pengetahuan tentang reaksi obat. Hasil penelitian menunjukkan bahwa metode C4.5 dapat digunakan untuk mengklasifikasi tentang reaksi obat. Hasil perhitungan akhir *entropy* dan *gain* serta pohon keputusan masih ada nilai atribut yang belum memasuki kelas [6].

Dari studi terdahulu diketahui bahwa pemrosesan bahasa alami telah banyak dilakukan berbasis teks namun demikian untuk pemrosesan bahasa alami dengan twitter untuk penyajian informasi lalu lintas belum dilakukan.

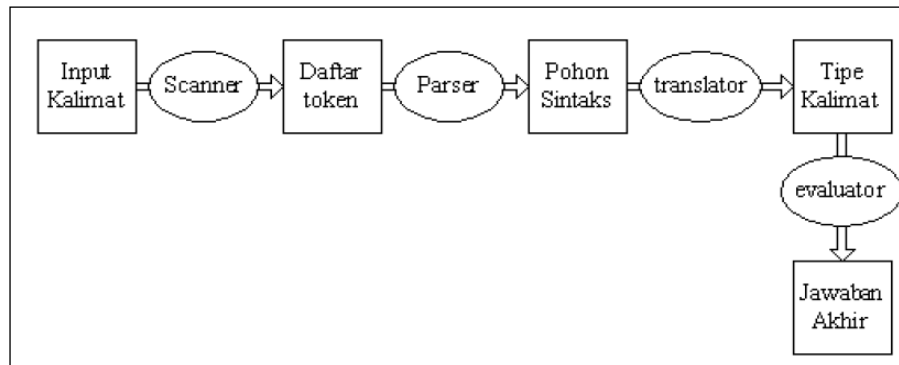
## 2 Metodologi Penelitian

Penelitian ini memiliki beberapa tahap pekerjaan, sebagaimana terlihat pada Gambar 1.



**Gambar 1.** Tahap Penelitian

Pada Gambar 1 terlihat bahwa penelitian ini membutuhkan aturan untuk penyusunan pemrosesan bahasa alami yang kemudian diimplementasikan menjadi Sistem dan Aplikasi Pengolah Bahasa Alami. Pemrosesan Bahasa Alami adalah suatu kemampuan suatu komputer untuk memproses bahasa lisan atau tulisan yang digunakan oleh manusia dalam percakapan keseharian [6]. Pengolah bahasa alami memiliki komponen yang saling bertahap satu sama lain, sebagaimana ditunjukkan pada Gambar 2.



**Gambar 2.** Komponen Pengolah Bahasa Alami

Proses kerja aplikasi pengolah bahasa alami sebagaimana terlihat pada Gambar 2, diawali dengan aktifitas pengguna yang memasukkan bahasa perintah untuk mencari data yang diinginkan dari database. Aplikasi pengolah bahasa alami memiliki komponen berupa *Scanner*, *Parser*, *Translator* dan *Evaluator*.

Pada elemen *Scanner*, dilakukan pemeriksaan atas kalimat bahasa alami yang dimasukkan untuk selanjutnya kalimat dikategorikan menjadi kata per kata yang dinamakan dengan token (tokenizing). Token adalah sederetan karakter yang membentuk satu kesatuan informasi dalam sebuah dokumen teks. Proses pemisahan token pada dokumen disebut sebagai tokenize. Sementara program yang melakukan proses tersebut disebut sebagai tokenizer (Jones, 1994) [7].

Daftar token yang telah terkumpul dari masukan pengguna dibandingkan dengan daftar token yang telah ditetapkan, pada elemen yang dinamakan *Parser*. Dengan upaya perbandingan tersebut maka diperoleh kata-kata yang masuk klasifikasi token yang dapat diproses dan daftar kata yang tidak dapat diproses.

*Parser* adalah metode untuk memetakan daftar token sesuai dengan aturan produksi (production rule) yang telah ditetapkan. Kegiatan Parsing adalah suatu proses menganalisa suatu kumpulan kata dengan memisahkan kata tersebut dan menentukan struktur sintaksis dari tiap kata tersebut. Production rule adalah susunan struktur pembentuk kalimat.

*Part of Speech* adalah pengklasifikasian kata secara grammar. Klasifikasi kata dapat terdiri dari kata benda, kata kerja, kata penghubung, dan sebagainya. Ada 3 pendekatan pengklasifikasian kata yang dapat digunakan. Pendekatan pertama adalah pendekatan formal. Pada pendekatan ini, anatomi dari kata digunakan sebagai penentu klasifikasi dari kata. Sebagai contoh, kata yang berawalan me- dapat digolongkan langsung sebagai kata kerja. Pendekatan kedua adalah pendekatan sintaktik. Pendekatan ini menggunakan klasifikasi dari kata lain di dekat kata yang tidak teridentifikasi. Sebagai contoh, kata Adj dalam bahasa Indonesia biasanya muncul tepat setelah kata benda: Jendela kotor, Meja baru, dan sebagainya. Pendekatan terakhir adalah pendekatan notional atau konteks. Pada pendekatan ini, sebuah klasifikasi dipahami maknanya dan digunakan sebagai penentu penggolongan kata. Sebagai contoh, kata benda adalah kata yang merepresentasikan suatu objek. Pendekatan ini sangat sulit diformulasikan dan karenanya kurang begitu dipakai dalam pengklasifikasian kata (Jones, 1994).

*Stemming* adalah proses pemotongan imbuhan pada kata untuk mendapatkan kata dasarnya (Porter, 1980). Dengan proses *stemming*, kata mencadangkan akan dipisahkan menjadi bentuk me-cadang-kan. Dengan kata dasar cadang yang berhasil diekstraksi [7]. Hasil dari *Parser* akan diproses pada *Translator* untuk mendapatkan tipe kalimat yang dapat diterima untuk pemrosesan berikutnya. Dari tipe kalimat yang dihasilkan itulah maka dilakukan pemrosesan akhir kalimat pada *evaluator* untuk mendapatkan perintah pencarian informasi yang diinginkan dan memunculkan hasilnya. Teknologi pengolah bahasa ini menggunakan format XML untuk pencarian informasinya.

XML kependekan dari eXtensible Markup Language, dikembangkan mulai tahun 1996 dan mendapatkan pengakuan dari W3C pada bulan Februari 1998. XML adalah turunan dari SGML (Standard Generalized Markup Language) yang telah dikembangkan pada awal 80-an dengan sintaks yang dinamakan HTML (Hyper Text Markup Language) [8]. Dokumen XML digunakan untuk mendeskripsikan dokumen itu sendiri dengan sintaks yang sederhana. XML bersifat mudah dibaca dan ditulis oleh manusia maupun komputer, sehingga XML merupakan sebuah format yang dapat digunakan untuk pertukaran data (interchange) antar aplikasi dan platform yang berbeda (platform independent).

XML Schema adalah bahasa yang digunakan untuk mendefinisikan struktur dokumen XML. Salah satu karakteristik dari XML Schema adalah sintaknya didasarkan pada XML itu sendiri. Hal ini memudahkan untuk dibaca dan tidak diperlukan *parser*, editor dan elemen lain-lain. XML QUERY LANGUAGE (XQUERY) XQuery adalah sebuah bahasa untuk melakukan query terhadap data XML. XQuery didisain dari XML Query Working Group dari W3C dengan tujuan tunggal untuk meng-query data yang tersimpan dalam format XML (Walmsley, 2007) dalam penelitian Andayani [4].

Pemrosesan bahasa alami yang digunakan pada penelitian ini, diterapkan pada database yang diperoleh dari hasil pengolahan ekstraksi data twitter tentang kondisi lalu lintas dan jalan raya yang berasal dari akun @tmcpoldametro. Berikut ini contoh data skunder dari akun twitter yang diperoleh:

15.34	Kondisi lalu lintas di Jl.Raya Serpong mengarah Alam Sutera terpantau <b>ramai lancar</b> , cuaca <b>hujan</b>
15:28	Truk <b>mogok</b> di KM 15 Tol Bekasi timur (arah ke Cikarang), lalin <b>tersendat</b> & msh penanganan
15.10	Hati2 melintas di ruas tol Cipularang KM 50 krn <b>hujan</b> deras disertai <b>angin</b> , menyebabkan jarak pandang terbatas
14:50	<b>Tawuran</b> antar Mahasiswa di dpn Kampus Mercu buana Meruya, agar hati2 apabila sdg melintas & msh penanganan.
14.58	<b>Traffic light</b> Mangga Dua Jl.P. Jayakarta <b>terganggu</b> menyebabkan <b>kepadatan</b> lalin, Petugas Dishub sudah dihubungi
13.37	<b>Kecelakaan</b> 2 mobil Avanza B 1392 VOE dan B 1916 FFK di KM 22 tol Bandara arah Pluit

Teknik ekstraksi pada data twitter menggunakan pendekatan data mining, sehingga diperoleh database XML tentang kondisi lalin dan jalan raya untuk dilakukan pencarian informasi menggunakan input teks dengan bahasa alami. Penyajian data dalam format XML memiliki keunggulan seperti kemudahan dalam integrasinya dengan aplikasi cloud seperti Application Programming Interface (API) yang diperlukan untuk mengembangkan arah penelitian lebih lanjut. Berikut ini adalah basis data kondisi jalan dan lalu lintas yang diperoleh:

```

<row>
  <idtraffic>3446</idtraffic>
  <hari>Rabu</hari>
  <tanggal>2012-11-14</tanggal>
  <>waktu>17:50</waktu>
  <informasi>17:50 Lbk bulus arah ke Ciputat lalin padat merayap, arah
  sebaliknya ramai lancar.</informasi>
</row>
<row>
  <idtraffic>4160</idtraffic>
  <hari>Rabu</hari>
  <tanggal>2012-11-14</tanggal>
  <waktu>20.36</waktu>
  <informasi>20.36 Arus lalu lintas Fatmawati menuju Cilandak terpantau masih
  padat</informasi>
</row>

```

Terlihat pada code XML di atas bahwa seluruh komponen informasi jalan dan lalu lintas telah berhasil dipetakan menjadi elemen seperti *idtraffic*, tanggal, waktu dan informasi kepadatan lalu lintas.

### Analisis Kebutuhan

Untuk dapat melakukan pemrosesan bahasa alami pada database XML tentang kondisi jalan dan lalu lintas, diperlukan suatu daftar kebutuhan informasi dari pengguna jalan yang ingin memanfaatkan database yang tersedia. Adapun daftar kebutuhan informasi tersebut dituangkan dalam contoh-contoh pertanyaan atau pernyataan dari pengguna jalan sebagai berikut:

*saya ingin informasi jalan sudirman  
 saya butuh info wilayah gunung sahari  
 beritahukan saya info lalin grogol  
 munculin sudirman  
 berikan saya keadaan daerah pancoran  
 tolong tampilkan kondisi wilayah ciputat  
 saya minta info lalin pondok indah  
 beritahukan saya info daerah matraman  
 munculin keadaan jalan hasyim ashari  
 gw minta info daerah grogol  
 mohon diinformasikan keadaan flyover pancoran  
 tolong informasikan situasi lalin ciputat  
 mohon keluarkan info pasar minggu  
 berikan saya kondisi perempatan gunung sahari  
 berikan saya kondisi neremntatan tamini saurare*

Agar kebutuhan informasi tersebut dapat diterima oleh pengguna jalan tersebut, maka diperlukan suatu pengolahan bahasa alami atas pernyataan/pertanyaan sehingga informasi akan dihasilkan berdasarkan pencarian (*Query*) pada database jalan dan lalu lintas. Berikut ini adalah contoh dari informasi yang muncul dari pengolahan database lalu lintas dan kondisi jalan:

**Data terkini:**

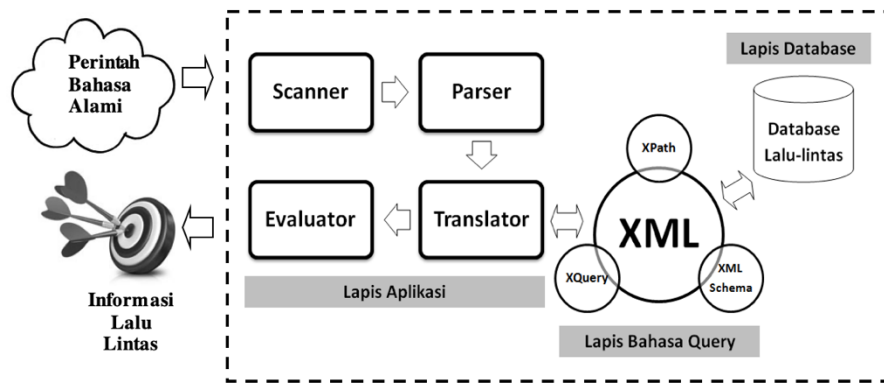
10.03 Kondisi lalin Jl. Gatsu dari pancoran arah Semanggi kedua arah ramai lancar  
 09.07 Kondisi lalin Jl. Gatsu dari Pancoran menuju perempatan Kuningan padat sebaliknya ramai lancar  
 06.17 Kondisi lalu lintas Jl Pramuka mengarah Matraman terpantau padat, tersendat di dpn Ps Genjing  
 09.51 Kondisi lalin JL WR Supratman Galur arah Cempaka Putih ramai lancar <http://t.co/D7Dma2YSGN>  
 06.13 Situasi lalu lintas dari Cempaka Putih menuju Senen terpantau ramai lancar

Berdasarkan database yang diperoleh tentang kondisi jalan dan lalu lintas hasil ekstraksi Twitter, diharapkan akan menjadi output hasil pengolahan bahasa alami untuk memunculkan informasi yang bermanfaat bagi pengguna jalan. Mengingat data kondisi jalan dan lalu lintas bersifat multi-series (menunjukkan kondisi dari menit ke menit atau jam ke jam bahkan hari ke hari), maka dapat diolah data seputar obyek lalu lintas (jalan/perempatan/bundaran dan sebagainya) berikut kondisinya (padat merayap, macet, ramai dan sebagainya) dengan informasi yang sama atau dalam waktu dan tanggal sebelumnya.

Tahap selanjutnya adalah penyusunan aturan produksi. Aturan produksi diperlukan untuk memastikan bahwa struktur kalimat bahasa alami yang diinput memenuhi persyaratan untuk diolah menjadi input teks untuk query pada basis data jalan dan lalu lintas. Berikut ini adalah daftar aturan produksi yang digunakan dalam penelitian ini:

```

<kalimat> → <awalan><perintah><dedicate><state><obyek><detail><detail2><derajat>
<awalan> → tolong|mohon|minta|ayo|coba|
<perintah>→tampilkan|tampilkan|munculkan|munculin|informasikan|informasiin|
            infokan|infoin|minta|kasih|kasihkan|beri|berikan|keluarkan|keluarin|
            ditampilkan|ditampilkan|dimunculkan|dimunculin|diinformasikan|
            diinformasiin|beritahu|diberitahu|diberitahukan|beritahukan|
            dikeluarin|dikeluarkan|minta|perlu|butuh
<dedicate> → saya|sy|gw|gue|ane|inyong|aye
<state> → situasi|kondisi|keadaan|suasana|info|informasi
<obyek> → jalan|jalanan|jln|lalin|daerah|drh|perempatan|tol|flyover|bundaran
<detail> → <letter><letter>*
<detail2> → <letter><letter>*
<letter> → |a|A|b|B|c|C|...|z|
  
```



**Gambar 3.** Arsitektur Sistem Pengolah Bahasa Alami pada penelitian ini (diadopsi dari Zuliarso & Hartati, 2008)

Terlihat pada Gambar 3, bahwa tahap *Parser*, merupakan proses pengolahan hasil kata-kata yang ditangkap oleh *Scanner* untuk dianalisis sintaksnya. Analisis dilakukan berdasarkan aturan produksi yang telah ditentukan untuk menghasilkan pohon sintaks. Untuk proses Parsing diperlukan daftar kelompok token agar *Parser* dapat langsung memisahkan mana kata yang diijinkan untuk diterima untuk diproses lebih lanjut serta mana kata atau token yang tidak didapat digunakan untuk pemrosesan selanjutnya. Pada Tabel 1 terlihat ada 7 (tujuh) kelompok token yang digunakan pada penelitian ini.

**Tabel 1.** Kategori token yang digunakan untuk pemrosesan.

<awalan>	<perintah>	<dedicate>	<state>	<obyek>	<detil1>	<detil2>
tolong	berikan	saya	situasi	jalan	hasyim	ashari
mohon	tampilkan	sy	kondisi	jalanan	cempaka	putih
ayo	munculkan	gw	keadaan	jln	pondok	indah
coba	munculin	gue	suasana	lalin	karet	bivak
	informasikan	ane	info	daerah	pasar	minggu
	informasiin	inyong	informasi	drh	yos	sudarso
	infokan	aye		perempatan	gunung	sahari
	infoin			tol	tanjung	duren
	minta			flyover	tamini	square
	kasih			bunderan	fatmawati	
	kasihkan			wilayah	ancol	
	beri				antasari	
	tampilkan				ciputat	
	keluarkan				matraman	
	keluarin				grogol	
	ditampilkan				pancoran	
	ditampilin				HI	
	dimunculkan				thamrin	
	dimunculin				biak	
	butuh				lodan	

Pada Tabel 1 terlihat bahwa komponen bahasa alami telah terpetakan kedalam tujuh kategori yaitu awalan, perintah, *dedicate*, *state*, obyek, detil1 dan detil2. Tahap pemrosesan selanjutnya adalah tahap *Scanner*, yaitu proses perubahan input kalimat dari tujuh kategori bahasa alami yang berasal dari pengguna menjadi daftar kata atau simbol yang selanjutnya diseleksi. Seleksi akan membuang kata-kata atau karakter yang dapat diabaikan sehingga memperoleh daftar kata yang memiliki arti (token) dalam basis data.

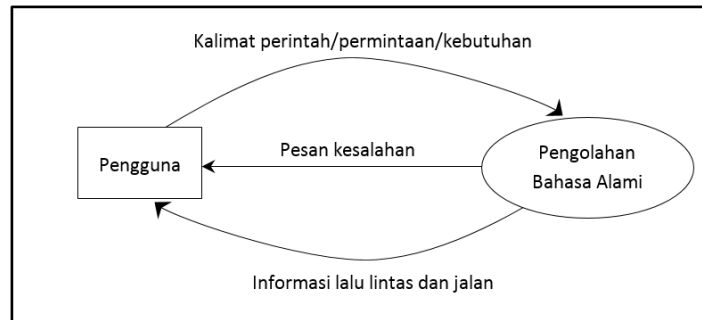
Penelitian ini membatasi jumlah informasi jalan dan lalu-lintas yang dapat diakses oleh aplikasi pengolah bahasa alami. Terlihat ppada token <detil1> dan <detil2> merupakan token berpasangan yang membentuk 20 (dua puluh) nama obyek utama di jalan raya di wilayah DKI Jakarta.

Tahap berikutnya adalah *Translator*, yang akan menghasilkan kode antara, sebelum memasuki proses evaluasi untuk menentukan jawaban akhir query. Tahap akhir, komponen Evaluator akan mengevaluasi daftar goal dalam proses penentuan jawaban query dan menyajikan informasi mengenai kondisi jalan & lalu lintas dalam basis teks.

### 3 Hasil Dan Pembahasan

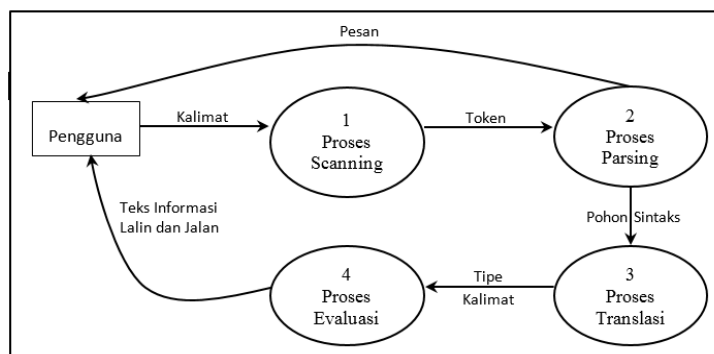
Bagian ini akan berisi mengenai rangkaian pengujian aplikasi pengolah bahasa alami atau NLP (Natural Language Processing) untuk mencari dan menyajikan data mengenai kondisi jalan dan lalu lintas yang sudah tersimpan dalam database XML.

Diagram sistem konteks sistem aplikasi pengolah bahasa alami untuk mendapatkan informasi lalu lintas dan jalan raya pada penelitian ini ditunjukkan pada gambar berikut:



**Gambar 4.** Diagram Konteks Aplikasi Pengolah Bahasa Alami

Terlihat pada Gambar 4 bahwa proses pengolahan bahasa alami yang dilakukan pada penelitian ini diawali dari kegiatan penggunaan aplikasi yang telah dibangun untuk menangkap input kalimat dari pengguna yang ingin mencari informasi lalu lintas dan jalan raya. Kalimat yang diinput berupa kalimat perintah atau kalimat permintaan atau kalimat yang menyatakan kebutuhan informasi jalan dan lalu lintas. Adapun Diagram Alir Data (DAD) Level I ditunjukkan pada Gambar 5.



**Gambar 5.** Diagram Alir Data Level I Aplikasi Pengolah Bahasa Alami

Terlihat pada Gambar 5 bahwa proses pengolahan bahasa alami dimulai dari pengguna yang mengirimkan kalimat bahasa alami untuk kemudian melewati tahap *scanning* dan *parsing*. Selanjutnya sistem akan melakukan proses translasi agar diketahui tipe kalimat yang diinput oleh pengguna. Tahap akhir adalah proses evaluasi untuk mendapatkan informasi lalu lintas dan jalan dari database.

Tampilan aplikasi pengolah bahasa alami untuk mencari informasi jalan dan lalu-lintas ditunjukkan pada Gambar 6.



**Gambar 6.** Tampilan layar aplikasi pengolah bahasa alami

Terlihat pada Gambar 6 bahwa tampilan aplikasi yang dibangun dengan bahasa PHP agar memiliki kemudahan dalam integrasinya dengan database berformat XML serta agar dapat dikembangkan lebih lanjut pada perangkat bergerak (*mobile*). Dari *interface* yang dimiliki, pada *textbox* yang tersedia pengguna memasukkan teks bahasa alami yang dibatasi pada jenis kalimat perintah atau kalimat permintaan yang menyatakan kebutuhan informasi jalan dan lalu-lintas.

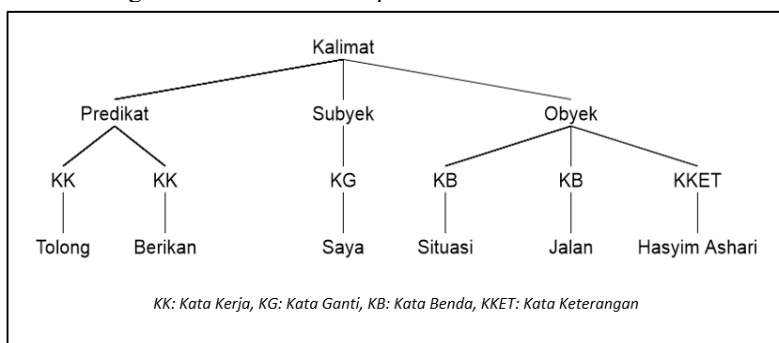
Kalimat input tersebut kemudian memasuki tahap Scanning yang memeriksa setiap karakter kalimat dan mengubah setiap kata menjadi token. Token-token yang diperoleh diseleksi dalam tahap Parsing agar diketahui token-token yang dapat digunakan untuk pemrosesan lebih lanjut. Adapun token yang diijinkan berada pada kategori token yang telah ditentukan seperti pada array berikut ini:

```

Public $arrayPerintah =
array('tampilkan','tampilin','munculkan','munculin','informasikan','informasiin','
infokan','infoin','minta','kasih','kasihkan','beri','berikan','keluarkan','kel
uarin','ditampilkan','ditampilin','dimunculkan','dimunculin',
'perlu','diinformasikan','diinformasiin','beritahu','diberitahu','diberitahuka
n','beritahukan','dikeluarin','dikeluarkan','minta','perlu','butuh');
public
$arrayobyek
array('jalan','jalanan','jln','lalin','daerah','drh','perempatan','tol','flyover',
'bunderan','wilayah');
public
$arraydedicate = array('saya','sy','gw','gue','ane','inyong','aye');
public
$arraydetail
array('hasyim','cempaka','pondok','karet','pasar','yos','gunung','tanjung',
'tamini','fatmawati','ancol',
'antasari','ciputat','matraman','grogol','pancoran','hi','thamrin','biak',
'sudirman');
public
$arraydetail2
array('ashari','putih','indah','bivak','minggu','sudarso','sahari','duren','square
',' ',' ',' ',' ',' ',' ',' ',' ',' ',' ');
public
$arrayderajat =
array('sangat','sangat','sangat','sangat','sangat','sangat');

```

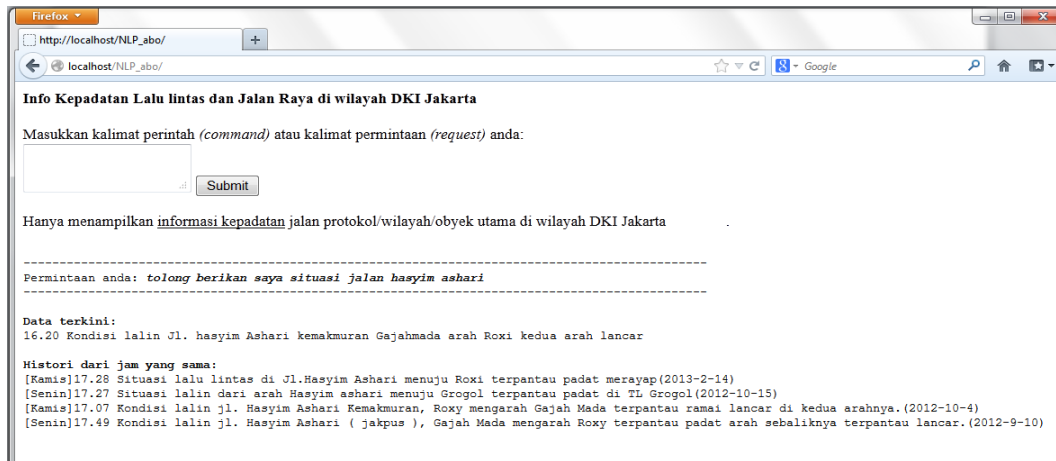
Dari sintaks terlihat bahwa seluruh komponen untuk menampilkan informasi telah memuat token yang dipetakan dari proses tokenisasi sebelumnya. Penulisan komponen tersebut dibuat bertujuan sebagai proses translasi berdasarkan pohon sintaks sebagaimana diilustrasikan pada Gambar 7.



**Gambar 7.** Pohon Sintaks untuk translasi

Terlihat pada Gambar 7 bahwa tahap translasi adalah proses mengadopsi struktur teks agar dapat diterima dan digunakan oleh aplikasi untuk mengakses informasi yang dibutuhkan bagi pengguna. Hasil adopsi teks dapat berupa tipe kalimat yang berpola (misalnya kalimat perintah dengan pola *Select \* from ... where ...*) untuk melakukan seleksi informasi pada basis data. Tipe kalimat yang berpola tersebut digunakan dalam tahap Evaluasi untuk melakukan pencarian (*querying*) informasi pada basis data jalan dan lalu-lintas. Apabila informasi yang dicari telah ditemukan, maka hasil diberikan kepada pengguna dalam format kalimat yang menunjukkan kondisi jalan dan lalu-lintas seperti terlihat pada Gambar 8.





**Gambar 8.** Informasi Jalan dan Lalu-lintas dari pengolahan bahasa alami

Dari Gambar 8 terlihat bahwa aplikasi yang dibangun mampu merespon inputan bahasa alami dari pengguna dan mengolah bahasa alami yang dientri hingga menampilkan informasi lalu lintas berdasarkan data twitter secara *real-time*. Proses pengujian dilakukan dengan membandingkan hasil penyajian sistem terhadap pemantauan video pada situs <http://lewatmana.com>. Verifikasi dilakukan pada empat ruas jalan nasional terhadap 35 (tiga puluh lima) salinan video jalan yang berhasil didapatkan dari <http://lewatmana.com> dalam satu pekan seperti terlihat pada Gambar 9.



**Gambar 9.** Klip video pembanding untuk hasil ekstraksi teks

Seluruh tahap verifikasi pada video-video yang ditunjukkan pada Gambar 9 menunjukkan bahwa dari 35 data video yang diverifikasi, sebagian besar kondisi lalu lintas pada 31 klip video atau sekitar 88,57% dinyatakan sesuai dengan informasi kepadatan lalu lintas yang disampaikan dari hasil ekstraksi teks. Sejumlah rekaman video dinyatakan kurang sesuai terhadap hasil ekstraksi teks, dikarenakan panjangnya ruas jalan yang diobservasi sementara jangkauan CCTV memiliki batasan.

## Kesimpulan

Dari penelitian yang telah dilakukan maka dapat disimpulkan hal-hal sebagai berikut:

1. Aplikasi pemrosesan bahasa alami dapat digunakan untuk mencari informasi (Querying) pada basis data berformat XML, yang berisi data historis tentang kondisi kepadatan jalan dan lalu-lintas hasil ekstraksi data twitter.
2. Aplikasi pemrosesan bahasa alami yang dibangun mampu mengenali kalimat permintaan atau kalimat perintah yang menyatakan kebutuhan pengguna akan informasi jalan dan lalu-lintas.
3. Aplikasi yang dibangun ini selain mampu menerima perintah bahasa alami untuk menyajikan data kondisi jalan dan lalu lintas yang bersifat historis juga dapat dikembangkan untuk menyajikan data yang bersifat real-time dari akun twitter yang sedang aktif (online).

Penelitian ini dapat dikembangkan lebih lanjut untuk mengakses basis data yang tidak terbatas pada obyek utama atau jalan raya protokol saja, dengan membangun Rules Production yang lebih kompleks dan menggunakan ekstraksi data jalan dan lalu lintas yang lebih bervariasi dari berbagai social media atau *microblogging* site lainnya.

## Referensi

- [1] Utami, E., Sri Hartati, 2010, *Pendekatan metode Rule Based dalam Mengalihbahasakan Teks Bahasa Inggris ke Teks Bahasa Indonesia*, Jurnal Petra, Universitas Kristen Petra, Surabaya.
- [2] Hartati, S., dan Zuliarso E., 2008, *Aplikasi Pengolah Bahasa Alami untuk Query Basisdata XML*, Dinamik, Vol. XIII, No. 2, Juli 2008, 168-175.
- [3] Supriyadi, 2019, *Penggunaan Bahasa Alami dalam Peningkatan Kualitas Citra Digital*, Jakarta
- [4] Andayani, S., 2002, *Query Bahasa Indonesia untuk Basis Data Akademik*, Tesis Magister Komputer, Program Pascasarjana Ilmu Komputer, Universitas Gadjah Mada, Yogyakarta.
- [5] Hanif Al Fatta, 2017, *Natural Language Processing dengan Teknik Machine Parser*, Jakarta
- [6] Prasetyantara, N, Kusriani, Nasiri, A., 2020, *Membangun Basis Pengetahuan Untuk Interaksi Obat Dengan Sosial Media*, Jurnal INFORMA Politeknik Indonusa, Vol. 6 Nomor 1.
- [7] Jones KS., 1994. *Natural Language Processing: she need something old and something new (maybe something borrowed and something blue, too)*. In *Proceeding of Association for Computational Linguistic*. England: Cambridge
- [8] Moh. Junaedi, 2019, *Pengantar XML*, Ilmukomputer.com.
- [9] Fauzi, 2016, *Pengolah Bahasa Alami Sebagai Query Fuzzy Tes Darah*, Semnas Teknomedia, Yogyakarta.
- [10] Soyusiawaty, D., Jones, A.H.S., *Pemanfaatan Bahasa Alami Dalam Penelusuran Informasi Skripsi Melalui Digital Library*, Jurnal Mobile and Forensic Vol. 2 No. 1.