Jurnal Linguistik Komputasional
Indonesian Association for Computational Linguistics

JLK
Vol 2, No 1
Maret 2019

# Analisis Morfologi untuk Menangani *Out-of-Vocabulary Words* pada *Part-of-Speech Tagger* Bahasa Indonesia Menggunakan *Hidden Markov Model*

Febyana Ramadhanti [1#], Yudi Wibisono [2#], Rosa Ariani Sukamto [3#]

*#Ilmu Komputer, Universitas Pendidikan* Indonesia
[1]*febyana.ilkom@student.upi.edu,* [2]*yudi@upi.edu,* [3]*rosa.ariani@upi.edu*

**Abstrak—*Part-of-speech* (PoS) *tagger* merupakan salah satu *task* dalam bidang *natural language processing* (NLP) sebagai proses penandaan kategori kata (*part-of-speech*) untuk setiap kata pada teks kalimat masukan. *Hidden markov model* (HMM) merupakan algoritma *PoS tagger* berbasis probabilistik, sehingga sangat tergantung pada *train corpus*. Terbatasnya komponen dalam *train corpus* dan luasnya kata dalam bahasa Indonesia menimbulkan masalah yang disebut *out-of-vocabulary* (OOV) *words*. Penelitian ini membandingkan *PoS tagger* yang menggunakan HMM+AM (analisis morfologi) dan *PoS tagger* HMM tanpa AM, dengan menggunakan *train corpus* dan *testing corpus* yang sama. *Testing corpus* mengandung 30% tingkat OOV dari 6.676 token atau 740 kalimat masukan. Hasil yang diperoleh dari sistem HMM saja memiliki akurasi 97.54%, sedangkan sistem HMM dengan metode analisis morfologi memiliki akurasi tertinggi 99.14%.**

**Kata kunci—*Bahasa Indonesia, natural language processing, part-of-speech tagging, hidden markov model, morphological analysis, out-of-vocabulary.***

## I. PENDAHULUAN

Komunikasi merupakan salah satu hal paling penting yang dibutuhkan manusia sebagai mahluk sosial. Dalam suatu negara, masyarakat umumnya berkomunikasi dengan menggunakan bahasa resmi negara tersebut, seperti bahasa Indonesia. Bahasa Indonesia merupakan bahasa resmi negara Indonesia sebagai identitas bangsa dan lambang kebanggan nasional, yang secara luas dan umum digunakan sebagai alat komunikasi oleh 222 juta orang [1]. Meskipun bahasa Indonesia dituturkan oleh sebagian besar orang di negara tersebut, tetapi ketersediaan alat pemroses bahasa untuk kepentingan penelitian masih terbatas. Sehingga, pengembangan sistem dan penelitian dibidang pemrosesan bahasa alami, khususnya bahasa Indonesia bagi masyarakat luas menjadi penting.

*Natural Language Processing* (NLP) merupakan suatu pengembangan teknik komputasi bahasa alami dalam menganalisis dan merepresentasikan teks ataupun lisan untuk mencapai pemrosesan bahasa seperti bahasa manusia [2]. Salah satu task dalam bidang NLP yaitu proses pelabelan kata dalam suatu kalimat masukan berdasarkan pada kategori katanya, atau yang disebut dengan *part-of-speech* (PoS) *tagger*. Karena pelabelan kelas kata secara manual akan memakan banyak waktu, melelahkan dan dengan biaya yang mahal, sehingga diperlukan pengembangan sistem PoS *tagger* secara otomatis. Contoh implementasi PoS *tagger* dalam aplikasi NLP yaitu pada aplikasi seperti *grammar checker*, *speech recognition*, *question answering* dan *machine translation* [3].

Salah satu metode PoS *tagger* yang telah dikembangkan [4] yaitu *hidden markov model* (HMM) dengan pendekatan *probabilistic-based* sehingga sangat tergantung pada *train corpus*. HMM merupakan pengembangan dari *Markov Model* yang mengasumsikan bahwa kata secara probabilistik bergantung pada dua atau lebih kategori kata sebelumnya.

Masalah utama PoS *tagger* menggunakan HMM disebabkan oleh adanya *out-of-vocabulary* (OOV) *words* pada saat proses masukan [5]. OOV merupakan kata yang tidak dikenali kelas katanya oleh sistem, yang disebabkan karena kata tersebut tidak terdapat dalam *train corpus* tetapi ada dalam *testing corpus* [6]. Dengan *train corpus* yang terbatas dibandingkan dengan kata dalam bahasa Indonesia yang sangat banyak, tentu sangat mungkin munculnya OOV *words*. Diperlukan suatu metode untuk dapat menyelesaikan masalah OOV tersebut.

Salah satu bentuk kata yang paling banyak muncul sebagai OOV dalam bahasa Indonesia yaitu kata yang

Submitted 07-01-2019; Revised 18-03-2019; Accepted 25-03-2019

dihasilkan dari proses morfologi yaitu proses pembentukan kata [6]. Proses morfologi yang dimaksud yaitu afiksasi atau proses pembentukan kata yang memiliki imbuhan, seperti kata *membantu* dengan imbuhan *mem-* atau *berlari* dengan imbuhan *ber-*. Kata yang memiliki imbuhan *mem-* dan *ber-* termasuk kedalam kelas kata verba atau kata kerja [7]. Sehingga, imbuhan (afiks) dapat menjadi panduan dalam proses penentuan kelas kata. Berdasarkan panduan tersebut metode analisis morfologi dapat menjadi solusi untuk menangani permasalahan OOV dalam sistem PoS *tagger* menngunakan HMM.

Penelitian [6] menerapkan *tools* yang disebut MorpInd [8] dalam menangani OOV. Sedangkan, penelitian ini akan mengembangkan sistem PoS *tagger* bahasa Indonesia dengan menggunakan *HMM* dan metode analisis morfologi yang dikembangkan berdasarkan pada aturan morfologi bahasa Indonesia untuk menangani permasalahan OOV.

## II. PENELITIAN TERKAIT

Berbagai metode dan sistem PoS *tingging* dengan pendekatan *rule-based* [9], *probabilistic-based* [10][11] dan *transformation-based* [12] telah banyak dipublikasikan. Tidak hanya dalam bahasa Indonesia [11], PoS *tagger* juga telah dikembangkan dalam berbagai bahasa, seperti bahasa Arab [10], bahasa Malaysia [11] dan bahasa India [14].

Wicaksono, dkk [11] menggunakan algoritma *Hidden markov model* (HMM) untuk PoS *tagger* dalam kalimat bahasa Indonesia. Masalah utama PoS *tagger* berbasiskan probabilistik tersebut adalah *out-of-vocabulary* (OOV) [6][5]. OOV merupakan suatu token yang muncul dalam data uji tetapi tidak terdapat dalam data latih [6].

OOV ditangani [11] menggunakan metode *Affix Tree*, yaitu suatu metode yang digunakan untuk memperoleh probabilitas emisi pada OOV. Selain itu, OOV juga dapat ditangani menggunakan metode Analisis morfologi (AM) [6]. Metode AM pada [6] mengatasi afiksasi (imbuhan) dengan menggunakan *tool* Morphind [8] untuk menentukan PoS *tag* bagi token OOV.

## *III. PART-OF-SPEECH (POS) TAGGER*

*Part-of-speech* (PoS) *tagger* atau biasa disingkat PoS *tagger* merupakan salah satu *task natural language processing* (NLP) dalam proses pelabelan kategori kata (*part-of-speech*) pada setiap kata dalam teks kalimat masukan terhadap suatu bahasa tertentu [10]. Karena dengan adanya PoS *tag* maka dapat diketahui bagaimana struktur sintaksis pada keseluruhan kalimat. Sehingga, PoS *tagging* sangat penting untuk aktivitas *syntatic parsing*, *grammer checker*, proses penerjemah bahasa tertentu kebahasa lainnya atau *machine translation*, hingga dalam memproduksi pelafalan untuk *speech recognition* [15].

Sebagai contoh implementasi PoS *tagging* yaitu pada kalimat masukan "*Ibu pergi ke pasar .",* maka keluarannya akan menjadi Ibu/PRP pergi/VB ke/IN pasar/NN ./Z . PoS *tag* yang terdapat pada keluaran kalimat tersebut diperoleh berdasarkan *tagset* pada Tabel I [16].

Ada tiga pendekatan yang digunakan dalam pengembangan PoS *tagger* yaitu pendekatan *probabilistic-based*, rule-*based* dan pendekatan *transformation-based* [3]. Salah satu algoritma PoS *tagger* dengan pendekatan *probabilistic-based* yaitu algoritma *Hidden Markov Model* (HMM). HMM merupakan pengembangan model statistik dari *Markov Model*. Dalam *Markov Model*, menghitung probabilitas setiap kejadian dapat terlihat langsung, dimana setiap busur antar state berisi nilai probabilitas yang dapat mengidentifikasi kemungkinan urutan jalur yang diambil. Tetapi, terkadang ada urutan kejadian yang ingin diketahui tetapi tidak dapat diamati. Oleh karena itu, dikembangkan model baru yang dapat memodelkan kejadian yang tersembunyi (*hidden*), yaitu *Hidden Markov Model*. Dalam proses PoS *tagging*, kejadian atau urutan tag tidak dapat terlihat secara langsung atau tersembunyi (*hidden state*), tetapi urutan kata yang bergantung terhadap tag tersebut dapat terlihat (*observed state*).

TABEL I. PART-OF-SPEECH TAGS [16]

| *No* | *Tag* | *Description* |
|---|---|---|
| 1. | CC | *Coordinating conjunction* |
| 2. | CD | *Cardinal number* |
| 3. | OD | *Ordinal number* |
| 4. | DT | *Determiner / article* |
| 5. | FW | *Foreign word* |
| 6. | IN | *Preposition* |
| 7. | JJ | *Adjective* |
| 8. | MD | *Modal and auxiliary verb* |
| 9. | NEG | *Negation* |
| 10. | NN | *Noun* |
| 11. | NNP | *Proper noun* |
| 12. | NND | *Classifier, partitive and measurement noun* |
| 13. | PR | *Demonstrative pronoun* |
| 14. | PRP | *Personal pronoun* |
| 15. | RB | *Adverb* |
| 16. | RP | *Particle* |
| 17. | SC | *Subordinating conjunction* |
| 18. | SYM | *Symbol* |
| 19. | UH | *Interjection* |
| 20. | VB | *Verb* |
| 21. | WH | *Question* |
| 22. | X | *Unknown word* |
| 23. | Z | *Punctuation* |

Dalam HMM untuk mendapatkan urutan PoS *tag* terbaik adalah dengan menggunakan *decoding*. Tujuan dari HMM *decoding* adalah memilih urutan *tag* yang paling mungkin yang berdasarkan pada urutan *observased state* yaitu n kata ($w_1^n$) [17]. Adapun persamaan HMM *bigram* ditunjukan pada persamaan (1),

$$t_1^n = \prod_{i=1}^{n} P(w_i|t_i)\ P(t_i|t_{i-1}) \quad (1)$$

dengan $P(t_i|t_{i-1}) = \frac{Count(t_{i-1},t_i)}{Count(t_{i-1})}$ dan $P(w_i|t_i) = \frac{Count(t_i,w_i)}{Count(t_i)}$.

Terdapat kasus yang muncul ketika perhitungan probabilitas emisi dan transisi, yang diakibatkan oleh munculnya OOV *words*. OOV yaitu kata yang tidak dapat ditemukan dalam *train corpus*, sehingga jumlah dari kata tersebut menjadi nol, sehingga tidak memungkinkan menentukan urutan kelas kata jika nilai probabilitas *bigram* nol. Untuk menyelesai kasus tersebut [18] menggunakan metode *laplace smoothing* yang ditunjukan pada persamaan (2),

$$P(w_i|t_i) = \frac{Count(t_i,w_i)+1}{Count(t_i)+|V|} \quad (2)$$

Dalam bahasa Indonesia, sebuah kata dapat memiliki satu atau lebih kelas kata, sehingga dalam proses *decoding*, algoritma viterbi merupakan algoritma terbaik dalam menyesaikan masalah tersebut secara cepat (dynamic programming) untuk menentukan PoS *tag* terbaik, yang ditunjukan pada persamaan (3) [17],

$$v_t(j) = max_{i=1}^{n}\ v_{t-1}(i)\ a_{ij}\ b_j(o_t) \quad (3)$$

dimana:
$v_t(j)$ = probabilitas HMM *state* $q_j$ pada waktu *t* setelah melalui observasi,
$v_{t-1}(i)$ = probabilitas viterbi path sebelumnya dari waktu sebelumnya,
$a_{ij}$ = probabilitas transisi dari state $q_i$ ke *state* $q_j$,
$b_j(o_t)$ = probabilitas emisi dari *observasi state* $o_t$ pada *state j*.

A. *Out-of-Vocabulary (OOV)*

Masalah yang muncul dalam PoS tagger menggunakan metode *hidden markov model* disebabkan adanya *out-of-vocabulary* (OOV) *words* [5]. Dalam proses PoS *tagging* berbasiskan probabilistik, *out-of-vocabulary words* diartikan sebagai kata atau token yang tidak dikenali sebab token tersebut muncul dalam *testing corpus* tetapi tidak terdapat dalam *train corpus* [6].

Dalam proses training sistem PoS tagger menggunakan HMM, kata-kata atau token yang merupakan OOV diklasifikasikan sebagai token yang tidak dapat dikenali karena frekuensi dari token tersebut bernilai nol (kosong), sehingga tidak memiliki nilai probabilitas emisi (kosong).

B. Analisis Morfologi (AM)

Proses morfologi adalah proses pembentukan kata dari sebuah bentuk dasar [7]. Sedangkan analisis morfologi merupakan suatu analisis terhadap proses morfologi itu sendiri, misal proses morfologi pada imbuhan *mem-* + *bantu* menjadi *membantu*, maka analisis morfologi akan menganalisis kata *membantu*, sehingga dapat diketaui bahwa kata tersebut terdiri dari imbuhan *mem-* dan kata dasar *bantu*.

Afiksasi merupakan salah satu proses morfologi pada kata turunan baik berkategori verba (kata kerja), nomina (kata benda) maupun ajektiva (kata sifat) [7]. Afiksasi dibagi ke dalam tiga kategori yaitu afiksasi pembentuk verba (kata kerja), afiksasi pembentuk nomina (kata benda) dan afiksasi pembentuk ajektiva (kata sifat).

TABEL II. AFIKASI PEMBENTUK VERBA, NOMINA DAN AJEKTIVA [7]

| | Kategori Kata | | |
|---|---|---|---|
| | Verba | Nomina | Ajektiva |
| Afiks | per- | -nya | se- |
| | -kan | pem- | -an |
| | -i | pen- | pe- |
| | per-kan | peny- | ter- |
| | per-i | peng- | ke-an |
| | ber-an | penge- | |
| | ber-kan | pe-an | |
| | me-kan | pem-an | |
| | me-i | pen-an | |
| | ke- | peng-an | |
| | ber- | penge-an | |
| | me- | per-an | |
| | mem- | ke- | |
| | men- | ter- | |
| | meny- | se- | |
| | meng- | -an | |
| | menge- | pe- | |
| | di- | ke-an | |
| | ter- | | |
| | ke-an | | |

## IV. HASIL DAN PEMBAHASAN

Penelitian ini membandingkan dua sistem PoS *tagger*, yaitu PoS *tagger* menggunakan HMM saja dan PoS *tagger* (HMM) yang didukung dengan metode analisis Morfologi.

### A. Analisis Data Penelitian

Analisis dilakukan pada *train corpus* dan *testing corpus* sebagai bahan utama penelitian. *Train corpus* yang digunakan pada penelitian ini berisi kumpulan kalimat bahasa Indonesia yang diperoleh dari penelitian yang telah dilakukan [19], lengkap dengan label kelas katanya yang diberikan secara manual. Data tersebut berisikan 10.026 kalimat atau 261.878 token yang telah dimodifikasi, yaitu dengan mengubah tanda pemisah antar kata dengan *tag* menjadi *vertical line* (|), mengubah bentuk data menjadi satu kalimat perbaris, mengubah pemisah antar token dengan spasi dan terakhir menghilangkan *bug* seperti " ├é┬ú ".

*Testing corpus* yang akan digunakan pada penelitian ini, diperoleh dari sumber media massa *online*. Kemudian data mentah tersebut akan diubah menjadi bentuk satu kalimat perbaris dan dilakukan proses pemilihan hingga menghasilkan 6676 token dengan 1990 OOV atau 30% tingkat OOV dari data keseluruhan.

TABEL III. JUMLAH OOV SETIAP KELAS KATA PADA *TESTING CORPUS*

| *No* | *Tag* | *Jumlah OOV* |
|---|---|---|
| 1. | CC | *0* |
| 2. | CD | *44* |
| 3. | OD | *0* |
| 4. | DT | *1* |
| 5. | FW | *13* |
| 6. | IN | *12* |
| 7. | JJ | *65* |
| 8. | MD | *5* |
| 9. | NEG | *0* |
| 10. | NN | *784* |
| 11. | NNP | *418* |
| 12. | NND | *12* |
| 13. | PR | *4* |
| 14. | PRP | *16* |
| 15. | RB | *0* |
| 16. | RP | *5* |
| 17. | SC | *9* |
| 18. | SYM | *0* |
| 19. | UH | *1* |
| 20. | VB | *582* |
| 21. | WH | *2* |
| 22. | X | *14* |
| 23. | Z | *3* |

### B. *Hidden Markov Model (HMM)*

Pada sistem PoS *tagger*, algoritma *hidden markov model* merupakan algoritma berbasis probabilistik yang digunakan untuk menentukan *tag* (kelas kata) terbaik pada setiap kata dalam data *testing.* Dalam implementasinya, HMM terlebih dahulu masuk pada tahap *training*, yaitu proses perhitungan terhadap *train corpus* untuk menghitung nilai probabilitas transisi dan probabilitas emisi yang kemudian nilai-nilai tersebut disimpan. Gambar I menunjukan mekanisme kerja proses *training*.

Berdasarkan hasil proses *training* dihasilkanlah 474 macam transisi dan 18.919 macam emisi dengan nilai probabilitasnya masing-masing kemudian disimpan. Nilai-nilai tersebut, kemudian akan menjadi panduan yang digunakan pada saat proses *tagging* terhadap *testing corpus*.

Pada proses *tagging*, sistem akan terlebih dulu melakukan tokenisasi pada kalimat masukan. Kemudian, akan masuk pada tahap perhitungan probabilitas viterbi (*forward step*) untuk memilih *tag* terbaik berdasarkan nilai probabilitias viterbi tertinggi. Jika ditemukan token yang merupakan OOV, nilai probabilitas emisinya akan dihitung menggunakan *laplace smoothing*. Setelah semua token sudah melalui tahap *forward*, kemudian masuk pada tahap *backward* untuk pemilihan jalur terbaik.

TABEL IV. HASIL PERHITUNGAN PROBABILITAS TRANSISI DAN EMISI UNTUK KALIMAT "SAYA SEDANG MELIHAT PERTANDINGAN OLAHRAGA."

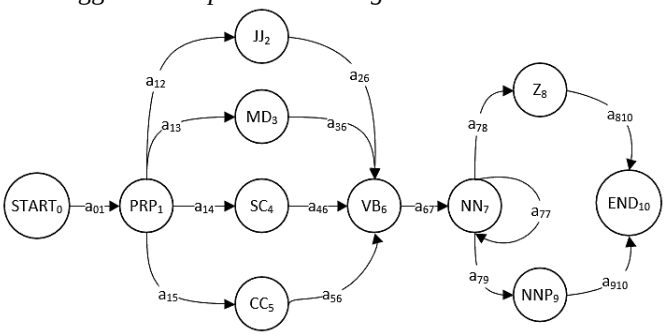
| *i* | Kata/token | x10$^2$ | |
|---|---|---|---|
| | | Probabilitas Emisi | Probabilitas Transisi |
| 0 | Saya | Saya\|PRP = 0.57441847 | <start> PRP = 3.9998002 |
| 1 | sedang | sedang\|MD = 0.9704918<br>sedang\|JJ = 0.01471959<br>sedang\|CC = 0.01720578<br>sedang\|SC = 0.0130736 | PRP MD = 3.213331<br>PRP JJ = 0.7620997<br>PRP CC = 0.5061707<br>PRP SC = 2.0872434 |
| 2 | melihat | melihat\|VB = 0.20140748 | MD VB = 0.268<br>JJ VB = 4.322653<br>CC VB = 6.7159899<br>SC VB = 4.3081884 |
| 3 | pertandingan | pertandingan\|NN = 2.65301249e-03 | VB NN = 27.10233869 |
| 4 | olahraga | olahraga\|NN = 0.02520361 | NN\|NN = 27.7982649 |
| 5 | . | .\|Z = 27.30399845<br>.\|NNP = 0.10285995 | NN\|Z = 7.09946143<br>NN\|NNP = 8.2004616 |

Gambar I menunjukan arsitektur HMM untuk kalimat "*Saya sedang melihat pertandingan olahraga .*" berdasarkan pada tabel IV.

C. HMM dan Analisis Morfologi (AM)

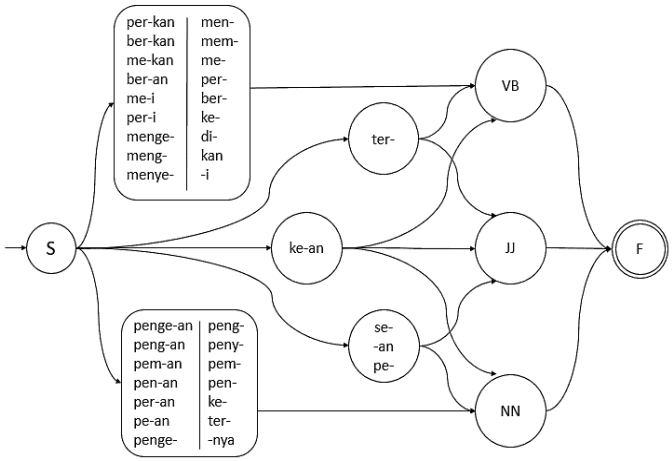
Sama seperti sistem sebelumnya (PoS *tagger* HMM), sebelum masuk pada proses *tagging* sistem terlebih dulu melakukan proses *training* untuk memperoleh nilai probabilitas transisi dan probabilitas emisi.

Pada sistem PoS *tagger* ini, metode analisis morfologi akan diimplementasikan kedalam proses *tagging*. Peran analisis morfologi adalah untuk menangani masalah *out-of-vocabulary* (OOV) yang muncul pada saat proses *tagging.* Token yang merupakan *out-of-vocabulary* (OOV) nilai probabilitas emisinya akan dihitung menggunakan *laplace smoothing*.




Gambar I. Arsitektur HMM pada kalimat *"Saya sedang melihat pertandingan olahraga ."*

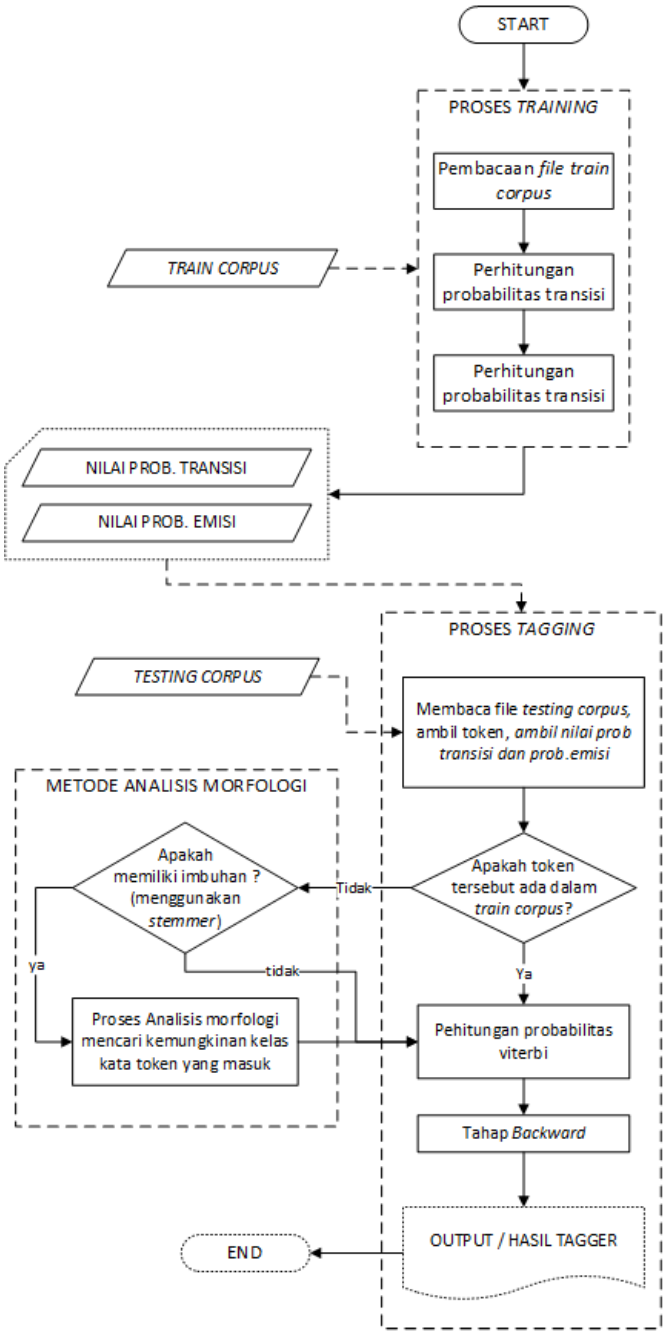
Token OOV yang ditangani oleh metode analisis morfologi adalah token yang miliki imbuhan (afiks). Metode AM mengambil imbuhan token OOV tersebut sebagai acuan dalam menentukan kelas kata, seperti kata *pelemparan* yang merupakan *noun* (NN) yang memiliki imbuhan *pe-an*, dan kata *mendamaikan* sebagai *verb* (VB) yang memiliki imbuhan *men-kan*. Oleh karena itu, berdasarkan ciri-ciri berupa imbuhan tersebut, metode analisis morfologi dapat menangani OOV pada kata yang memiliki imbuhan, sehingga dapat diketahui kelas kata yang tepat. Gambar II menunjukan *finite state automata* (FSA) metode analisis morfologi (AM) sebagai referensi pada proses AM dalam penentuan *tag* kelas kata, dengan berbasis aturan atau *rule based* tanpa menggunakan kamus kata dasar. Serta Gambar III menunjukan mekanisme kerja sistem PoS *tagger* HMM+AM.




Gambar II. FSA metode analisis morfologi

Berikut ini merupakan langkah-langkah pada proses HMM dan metode AM.

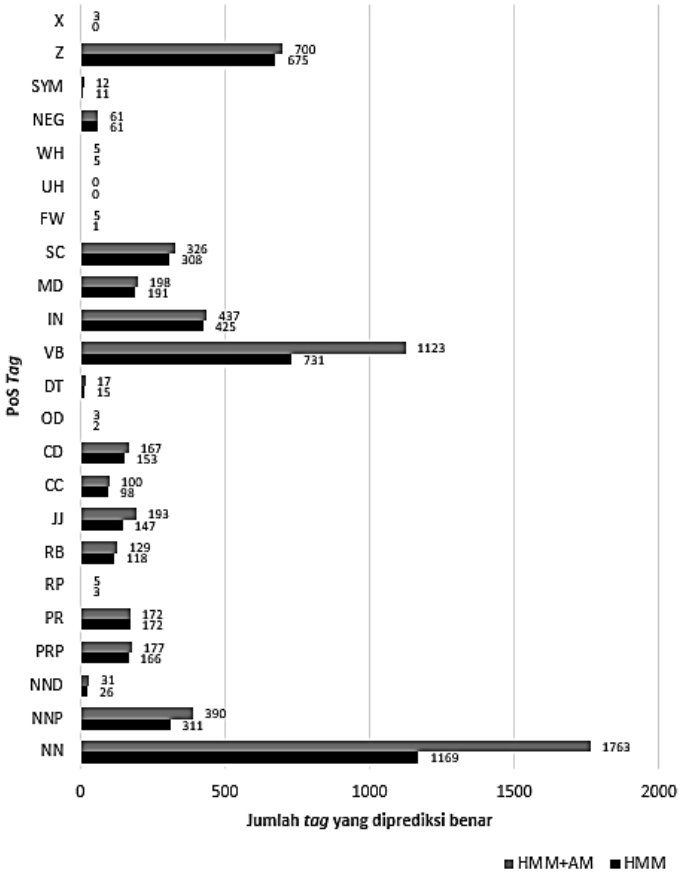
1. Token yang bukan merupakan OOV akan langsung dihitung nilai probalibilitas viterbinya.
2. Token yang merupakan OOV akan diambil kata dasarnya menggunakan modul *stemmer* yaitu *stemmer* sastrawi, kemudian kata dasar itu akan dicocokan dengan token yang sebelum di-*stemmer*, jika berbeda, artinya kata tersebut merupakan kata yang memiliki imbuhan. Hal tersebut dilakukan untuk menghindari pemrosesan AM pada kata dasar yang memiliki ciri-ciri yang sama dengan imbuhan seperti kata *beras* (imbuhan ber-), *peluit* (imbuhan pe-), *makan* (imbuhan -kan) dan sebagainya.
3. Kata yang memiliki imbuhan akan dipriksa beberapa karakter depan dan/atau karakter belakangnya untuk memperoleh imbuhan kata tersebut, sehingga dapat diketahui kelas katanya. Kata yang hanya memilki satu kemungkinan *tag* seperti kata dengan imbuhan *ber-* pada kelas kata *verb* (VB) dan *pem-* pada kelas kata *noun* (NN), akan langsung masuk pada tahap perhitungan probabilitas emisi dan perhitungan probabilitas viterbi. Sedangkan, kata yang memiliki lebih dari satu kemungkinan kelas katanya seperti imbuhan *ter-* (VB/JJ) akan ditentunkan *tag* terbaiknya dengan berdasarkan juga pada nilai probabilitas transisinya.
4. Setelah point 1-3 selesai pada semua token dalam kalimat masukan, token-token tersebut akan masuk pada tahap *backward* (*backpointer*) untuk menentukan jalur terbaiknya.




Gambar III. Mekanisme kerja POS *tagger* menggunakan HMM + AM

D. *Pengujian Sistem*

Evaluasi pada kedua sistem PoS *tagger* (HMM dan HMM+AM) dilakukan dengan menggunakan *train corpus* dan *testing corpus* yang sama.




Gambar IV. Perbandingan Jumlah *tag* yang diprediksi benar oleh kedua sistem

Hasil *tagging* yang diprediksi benar oleh kedua sistem ditunjukan pada Gambar IV. Secara keseluruhan dapat dilihat bahwa PoS *tagger* dengan menggunakan HMM+AM hampir pada semua PoS *tag* diprediksi dengan benar lebih banyak dibandingkan dengan PoS *tagger* menggunakan HMM saja.

Hasil pengujian PoS *tagger* menggunakan HMM yang disajikan pada Tabel IV menunjukan bahwa sistem tersebut memiliki akurasi **97.54%**. Masih terjadi kesalahan prediksi, yang pada umumnya terjadi pada token yang merupakan OOV. Kesalahan paling dominan terjadi pada *tag* NN, NNP dan VB. Hal itu terjadi karena dalam *testing corpus* banyak kata benda, kata kerja dan kata ganti nomina (nama orang, nama tempat).

Pada sistem, token OOV akan diasumsikan dengan semua kemungkin *tag* (23 *tag*), sehingga *tag* pada token tersebut dapat berupa apapun tergantung pada probabilitas viterbi dan pada saat proses *backward.* Munculnya OOV disebabkan karena komposisi *train corpus* yang terbatas, sehingga sistem ini sangat tergantung pada komposisi *train corpus.*

TABEL IV. HASIL KINERJA POS *TAGGER* HMM

| ***Confusion Matrix*** | | **Ukuran Kinerja** | | | |
|---|---|---|---|---|---|
| | | *Akurasi* | *Presisi* | *F-Score* | *Recall* |
| TP = 4788 | FP = 1888 | 97,54 % | 71,72 % | 71,72 % | 71,72 % |
| FN = 1888 | TN=144984 | | | | |

TABEL V. HASIL KINERJA POS *TAGGER* HMM+AM

| ***Confusion Matrix*** | | **Ukuran Kinerja** | | | |
|---|---|---|---|---|---|
| | | *Akurasi* | *Presisi* | *F-Score* | *Recall* |
| TP = 6017 | FP = 659 | 99.14% | 90,13 % | 90,13 % | 90,13 % |
| FN = 659 | TN =146213 | | | | |

Pada Tabel V, pengujian PoS *tagger* menggunakan HMM+AM memiliki akurasi **99.14%**, angka tersebut sudah sangat baik untuk PoS *tagger* bahasa Indonesia. Dapat dilihat pada Gambar V, prediksi PoS *tag* dengan benar dibandingkan dengan sistem HMM saja paling dominan terjadi pada *tag* VB dan NN, hal itu terjadi karena OOV pada sistem kedua ditangani oleh metode analisis morfologi. Walaupun kinerja sistem ini sudah sangat baik, tetapi masih ada kesalahan dalam memprediksi kelas kata. Kesalahan tersebut paling dominan terjadi pada tag NNP (*proper noun*) yang disebabkan karena token yang merupakan OOV seperti nama orang, nama tempat, nama lembaga dan sebagainya tidak dapat diproses oleh AM. Selain itu, kesalahan prediksi juga terjadi hampir pada semua *tag* kecuali PRP (*personal pronoun*), CC (*coordinating conjuction*), OD (*ordinal number*), MD (*modal and auxiliary verb*), NEG (*negation*), SYM (*symbol*) dan Z (*punctuation*). Sama seperti pada point sebelumnya, kesalahan tersebut terjadi karena token-token itu merupakan OOV yang bukan merupakan kata yang memiliki imbuhan, sehingga tidak dapat diproses oleh metode AM.

## KESIMPULAN

Penerapan metode AM pada sistem PoS *tagger* menggunakan HMM memiliki akurasi lebih baik dibandingkan PoS tagger tanpa AM dalam menangani OOV. Metode analisis morfologi yang dikembangkan belum sepenuhnya efektif karena hanya dapat mengatasi OOV yang memiliki imbuhan saja. Oleh karena itu, dibutuhkan penelitian lebih lanjut untuk mengembangkan metode AM yang dapat mengatasi OOV yang merupakan kata perulangan, kata penyerapan dan mengetahui kelas kata unutk angka, serta *proper noun*. Secara garis besar, sistem PoS *tagger* ini juga sangat tergantung pada komposisi *train corpus*.

## REFERENSI


[1] Lewis, M. P. (2009). *Enthnologue: Language of the World*, 6th ed., Dallas.

[2] Liddy, E. D. (2001). *Natural Language Processing . In Encyclopedia of Library and Information Science, 2nd Ed. NY. Marcel Decker, Inc.*

[3] Pisceldo, F., Adriani, M., & Manurung, R. (2009). *Probabilistic Part Of Speech Tagging for Bahasa Indonesia. in Proceedings of Third International Wokshop on Malay and Indonesian Language Engineering*, Singapore.

[4] Kumar, R., & Shekhawat, S. S. (2018). *Parts Of Speech Tagging For Hindi Languages Using Hmm . International Journal Of Scientific Research.*

[5] Brants, T. (2000). *Tnt - a Statastical Part-of-Speech Tagger. Proceeding of the sixth conference on Applied Natural Language Processing.*

[6] Muljono, Afini, U., & Supriyanto, C. (2017). *Marphology Analysis for Hidden Markov Model based Indonesian Part-of-Speech Tagger. 1st International Conference on Informatics and Computational Sciences (ICICoS).*

[7] Chaer, A. (2008). Morfologi Bahasa Indonesia. Jakarta: Rineka Cipta.

[8] Larasati, S.D., Kuboň, V. and Zeman, D., 2011, August. Indonesian morphology tool (morphind): Towards an indonesian corpus. In *International Workshop on Systems and Frameworks for Computational Morphology* (pp. 119-129). Springer, Berlin, Heidelberg.

[9] Alfred, R., Mujat, A. and Obit, J.H., 2013, March. A ruled-based part of speech (RPOS) tagger for Malay text articles. In *Asian Conference on Intelligent Information and Database Systems* (pp. 50-59). Springer, Berlin, Heidelberg.

[10] Shamsi, F. A., & Guessoum, A. (2006). *A Hidden Markov Model –Based POS Tagger for Arabic. Journées internationales d'Analyse statistique des Données Textuelle.*

[11] Wicaksono, A. F., & Purwarianti, A. (2010). *HMM Based Part-of-Speech Tagger for Bahasa Indonesia. Proceeding of the Fourth Internationul MALINDO Workshop* (MALINDO2010), Jakarta.

[12] Brill, E. (1992). *A Simple Rule-Based Part of Speech Tagger. Proceedings of the Third Conference on Applied Computational Linguistics.* Trento, Italy: *Association of Computational Linguistics.*

[13] Alfred, R., Mujat, A., & Obit, J. H. (2013). *A Ruled-Based Part of Speech (RPOS) Tagger for Malay Text Articles. Asian Conference on Intelligent Information and Database Systems.*

[14] Joshi, N., Darbari, H., & Mathur, I. (2013). *HMM BASED POS TAGGER FOR HIND. Proceeding of 2013 International Conference on Artificial Intelligence, Soft Computing.*

[15] Manurung, R. (2016). Tutorial: Pengenalan terhadap *POS Tagging* dan *Probabilistic Parsing*. *Workshop Nasional INACL*. Depok: Fakultas Ilmu Komputer Universitas Indonesia. .

[16] Dinakaramani, A., *Rashel, F., Luthfi, A., & Manurung, R. (2014). Designing an Indonesian Part of Speech Tagset and Manually Tagged Indonesian Corpus. In Asian Language Processing (IALP), Kuching*.

[17] Jurafsky, D., & Martin, J. H. (2014). *Speech and Language Processing*. Vol. 3. London: Pearson.

[18] Kikuchi, M., Yoshida, M., Okabe, M., & Umemura, K. (2015). *Confidence Interval of Probability Estimator of Laplace Smoothing. Institute of Electrical and Electronics Engineers.*

[19] Rashel, F., Luthfi, A., Dinakaramani, A., & Manurung, R. (2014). *Building an Indonesian Rule-Based Part-of-Speech Tagger. Asian Language Processing* (IALP), 2014 International Conference on. IEEE.