Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK) DOI: 10.25126/jtiik.202184483
Vol. 8, No. 5, Oktober 2021, hlm. 1039-1048 p-ISSN: 2355-7699
Akreditasi KEMENRISTEKDIKTI, No. 36/E/KPT/2019 e-ISSN: 2528-6579

KLASIFIKASI KELAS KATA (*PART-OF-SPEECH TAGGING*) UNTUK BAHASA MADURA MENGGUNAKAN ALGORITME VITERBI

Ilham Firmansyah*[1], Putra Pandu Adikara[2], Sigit Adinugroho[3]

[1,2,3]Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya Malang
Email: [1]firman.ilham05@gmail.com, [2]adikara.putra@ub.ac.id, [3]sigit.adinu @ub.ac.id
*Penulis Korespondensi

(Naskah masuk: 10 Desember 2020, diterima untuk diterbitkan: 19 Oktober 2021)

Abstrak

Bahasa manusia adalah bahasa yang digunakan oleh manusia dalam bentuk tulisan maupun suara. Banyak teknologi/aplikasi yang mengolah bahasa manusia, bidang tersebut bernama *Natural Language Processing* yang merupakan ilmu yang mempelajari untuk mengolah dan mengekstraksi bahasa manusia pada perkembangan teknologi. Salah satu proses pada *Natural Language Processing* adalah *Part-Of-Speech Tagging*. *Part-Of-Speech Tagging* adalah klasifikasi kelas kata pada sebuah kalimat secara otomatis oleh teknologi, proses ini salah satunya berfungsi untuk mengetahui kata-kata yang memiliki lebih dari satu makna/arti (ambiguitas). *Part-Of-Speech Tagging* merupakan dasar dari *Natural Language Processing* lainnya, seperti penerjemahan mesin (*machine translation*), penghilangan ambiguitas makna kata (*word sense disambiguation*), dan analisis sentimen. *Part-Of-Speech Tagging* dilakukan pada bahasa manusia, salah satunya adalah bahasa Madura. Bahasa Madura adalah bahasa daerah yang digunakan oleh suku Madura dan memiliki morfologi yang mirip dengan bahasa Indonesia. Penelitian pada *Part-Of-Speech Tagging* pada bahasa Madura ini menggunakan algoritme Viterbi, terdapat 3 proses untuk implementasi algoritme Viterbi pada pada *Part-Of-Speech Tagging* bahasa Madura, yaitu *pre-processing* pada data *training* dan *testing*, perhitungan data latih dengan *Hidden Markov Model* dan klasifikasi kelas kata menggunakan algoritme Viterbi. Kelas kata (*tagset*) yang digunakan untuk klasifikasi kata pada bahasa Madura sebanyak 19 kelas, kelas kata tersebut dirancang oleh pakar. Pengujian sistem pada penelitian ini menggunakan perhitungan *Multiclass Confusion Matrix*. Hasil pengujian sistem mendapatkan nilai *micro average accuracy* sebesar 0,96 dan nilai *micro average precision* dan *recall* yang sama sebesar 0,68. *Precision* dan *recall* masih dapat ditingkatkan dengan menambahkan data yang lebih banyak lagi untuk pelatihan.

Kata kunci: *ekstraksi informasi, pemrosesan bahasa alami, Part-Of-Speech Tagging, bahasa Madura, Hidden Markov Model, Viterbi*

CLASSIFICATION OF WORDS CLASS (PART-OF-SPEECH TAGGING) FOR BAHASA MADURA USING VITERBI ALGORITHM

Abstract

Natural language is a form of language used by human, either in writing or speaking form. There is a specific field in computer science that processes natural language, which is called Natural Language Processing. It is a *study of how to process and extract natural language on technology development. Part-Of-Speech Tagging is a method to assign a predefined set of tags (word classes) into a word or a phrase. This process is useful to understand the true meaning of a word with ambiguous meaning, which may have different meanings depending on the context. Part-Of-Speech Tagging is the basis of the other Natural Language Processing methods, such as machine translation, word sense disambiguation, and sentiment analysis. Part-Of-Speech Tagging used in natural languages, such as Madurese language. Madurese language is a local language used by Madurese and has a similar morphology as Indonesian language. Part-Of-Speech Tagging research on Madurese language using Viterbi algorithm, consists of 3 processes, which are training and testing corpus pre-processing, training the corpus by Hidden Markov Model, and tag classification using Viterbi algorithm. The number of tags used for words classification (tagsets) on Madurese language are 19 class, those tags were designed by an expert. Performance assessment was conducted using Multiclass Confusion Matrix calculation. The system achieved a micro average accuracy score of 0,96, and micro average precision score is equal to recall of 0,68. Precision and recall can still be improved by adding more data for training.*

Keywords: *information extraction, natural language processing, Part-Of-Speech Tagging, Madurese, Hidden Markov Model, Viterbi*

1. PENDAHULUAN (huruf besar, 10pt, tebal)

Perkembangan teknologi telah merambah ke bidang linguistik, contohnya pada bahasa manusia yang telah banyak dikembangkan dengan cara mengolah dan mengekstraksi bahasa manusia untuk dijadikan model komputasinya. Salah satu perangkat lunak yang mengolah dan mengekstraksi informasi dalam bahasa manusia adalah *Natural Language Processing* (NLP). Pengolahan bahasa manusia pada NLP memerlukan beberapa proses, salah satunya adalah *part-of-speech* (POS) *tagging* yang diterapkan dalam mesin penerjemah (*machine translation*), pencarian kata ambigu (word sense disambiguation) dan temu kembali informasi (information retrieval) (JURAFSKY & MARTIN, 2019). POS *tagging* adalah klasifikasi kelas kata secara otomatis pada suatu kalimat atau paragraf, sehingga hasilnya berfungsi untuk membedakan kata dengan susunan huruf yang sama tetapi memiliki arti yang berbeda (ambigu).

POS *tagging* telah banyak dilakukan pada beberapa bahasa manusia, salah satu contoh bahasa manusia adalah bahasa Madura. Bahasa Madura adalah salah satu bahasa daerah yang banyak digunakan di Indonesia dan sangat berpengaruh dalam Bahasa Sumber Serapan (BSS) bahasa Indonesia pada beberapa aspek tertentu (AZHAR, 2011), tetapi keberadaan bahasa Madura mulai terancam punah karena proses globalisasi dan urbanisasi yang menyebabkan asimilasi dan akulturasi budaya (RAHILAH, et al., 2013). Bahasa Madura juga memiliki morfologi yang sama dengan bahasa Indonesia, sehingga bahasa Madura dapat terjadi kata yang ambigu (SHOLIHIN, et al., 2013) . Oleh karena itu, bahasa Madura perlu dipertahankan dan dikembangkan melalui teknologi, salah satunya dengan cara melakukan POS *tagging* pada bahasa Madura.

Penelitian tentang POS *tagging* pada beberapa bahasa di Indonesia telah banyak dilakukan dengan berbagai metode. Penelitian yang membandingkan berbagai metode dan corpus dari tahun 2008 sampai 2019, menghasilkan metode terbaik yaitu arsitektur neural network dengan menggunakan bidirectional LSTM dan CRF dengan nilai akurasi sebesar 95,68% (KAMAYANI, 2019). Penelitian tentang POS *Tagging* berbasis aturan dan distribusi probabilitas maximum entropy pada bahasa Jawa Krama mendapatkan hasil akurasi 97.67% (PRAMUDITA, et al., 2016)**.** Penggunaan algoritme Brill Tagger pada POS *Tagging* Bahasa Indonesia mendapatkan hasil akurasi sebesar 89.70% **(**SETYANINGSIH**,** 2017**)**. Penggunaan Hidden Markov Model (HMM) dengan menggunakan algoritme Viterbi pada penerapan analisis morfologi untuk implementasi POS *Tagging* Bahasa Indonesia, mendapatkan tingkat akurasi 99.14% dengan menggunakan data yang sama pada data latih dan uji **(**RAMADHANTI**,** et al**.,** 2019**).**

2. METODE PENELITIAN

Untuk dapat melakukan klasifikasi kelas kata atau yang dikenal sebagai *Part-of-Speech* (POS) *Tagging* maka sebelumnya harus mengetahui bagaimana struktur bahasa Madura terlebih dahulu. Bahasa Madura adalah bahasa daerah yang digunakan oleh suku Madura sebagai alat komunikasi, untuk memperlihatkan identitas sebagai salah satu suku yang ada di Indonesia yaitu suku Madura. Bahasa Madura memiliki 4 dialek yang tersebar di seluruh wilayah berbeda, yaitu dialek Bangkalan, Sampang, Pamekasan, dan Sumenep. Dialek yang menjadi acuan standar bahasa Madura adalah dialek sumenep yang merupakan pusat kerajaan dan kebudayaan Madura pada zaman dahulu. Bahasa Madura terpengaruh dari bahasa lainnya, seperti bahasa Jawa, Melayu, Bugis, Tionghua dan lainnya (EFFENDY, 2017). Bahasa Madura memiliki 3 tingkat bahasa, yaitu:

1. Tingkat "*Engghi Bunten*" setara dengan tingkat *krama* pada Bahasa Jawa, tingkatan ini digunakan untuk orang yang lebih tua, seperti orang tua, kakak, guru dan lainnya. Tingkatan ini digunakan pada acara-acara adat atau resmi, contoh kosa kata bahasa Madura pada tingkat "*Engghi Bhunten*" yaitu "*kaule*" artinya aku, "*meyos*" artinya pergi, "*ponapa*" artinya kenapa dan lain sebagainya.
2. Tingkat "*Engghi Enten*" setara dengan tingkat *madya* pada Bahasa Jawa, tingkatan ini digunakan untuk orang yang setara, seperti teman kerja dan lainnya. Contoh kosa kata bahasa Madura pada tingkat "*Engghi Enten*" yaitu "*Sengko"*" artinya aku, "*Pasera*" artinya siapa, "*Aneko*" artinya ini dan lain sebagainya.
3. Tingkat "*Enje' iye*" setara dengan tingkat *ngoko* pada Bahasa Jawa, tingkatan ini digunakan untuk orang yang lebih muda, seperti adik dan lainnya. Tingkatan ini biasanya banyak digunakan pada kehidupan sehari-hari, contoh kosa kata bahasa Madura pada tingkat "*Enje' iye*" yaitu "*Engko"*" artinya aku, "*Ajelen*" artinya berjalan, "*Reya*" artinya ini dan lain sebagainya.

Dalam bahasa Indonesia yang dikenal pada umumnya, suatu teks berisi kalimat. Kalimat adalah susunan kata-kata yang memiliki makna yang lengkap, dan juga satuan bahasa terkecil dalam menyampaikan suatu pemikiran. Dalam ilmu linguistik, kalimat merupakan satuan bahasa yang terdiri dari beberapa klausa dan pola intonasi. Secara sintaksis kalimat memiliki beberapa unsur dalam sebuah penyusunannya, yaitu subjek yang berfungsi sebagai pelaku perbuatan, predikit sebagai penjelas pada unsur subjek, objek sebagai target dalam perkerjaan subjek, pelengkap dan keterangan sebagai unsur tambahan.

Jenis-jenis kalimat dibedakan peran subjek dan predikat, cara penyampaian, bentuk sintaksis dan jumlah klausa. Pada jenis kalimat yang dilihat antara

peran subjek dan predikit dibedakan menjadi 2 bagian, yaitu kalimat aktif dan pasif, sedangkan jenis kalimat pada cara penyampaiannya adalah kalimat langsung dan tidak langsung. Berdasarkan bentuk kalimat, ada 4 jenis kalimat yaitu kalimat perintah, tanya seru dan berita. Banyaknya jumlah klausa pada kalimat dibedakan menjadi 2 jenis, yaitu kalimat majemuk dan tunggal (PRIHANTINI, 2015).

Bahasa Indonesia memiliki keterkaitan bahasa dengan bahasa Madura dari sisi morfologi, fonologi ataupun sintaksisnya. Bahasa Madura memiliki pola pola kata yang mirip dengan bahasa Indonesia. Bahasa Madura memiliki pola kata yang mirip dengan bahasa, dan imbuhan kata pada bahasa Madura sama dengan bahasa Indonesia yang berupa *Ter-ater* (Prefiks), *Pantoteng* (Sufiks), dan *Sesselan* (Sisipan). Bahasa Madura memiliki karakter khusus untuk memudahkan pembaca baik orang Madura ataupun bukan. Berikut karakter khusus pada bahasa Madura:

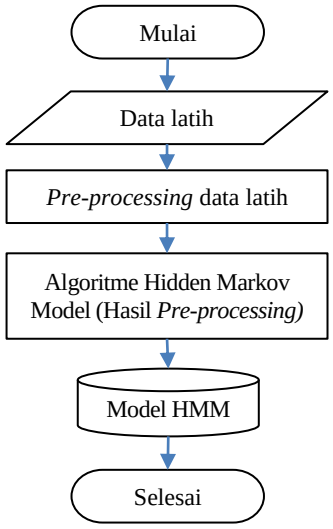
- *a* = dibaca *a* biasa seperti kata bawah.
- *â* = dibaca *e* seperti kata belajar.
- *e* = dibaca *e* biasa seperti kata kertas.
- *è* = dibaca *e* seperti kata bebas.
- *bh, gh, jh,* dan *dh* = dibaca lebih tebal, contohnya *bhâjâ*.
- Tanda petik (’) = dibaca seperti kata tidak.

Penyusunan kalimat pada bahasa Madura sama dengan bahasa Indonesia. Bahasa Indonesia terdapat bagian subjek (S), predikat (P), objek (O), keterangan (K), sedangkan bahasa Madura memiliki bagian yang sama. Contoh penulisan kalimat pada bahasa Madura (M) dan Indonesia (I):

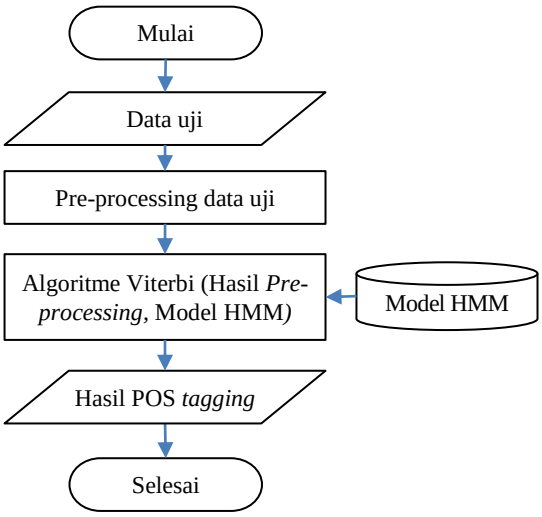
- Susunan S + P + O
  Madura : *sengko’ ngakan nase’.*
  Indonesia : aku makan nasi.
- Susunan S + P + O + K
  Madura : *Andi melle kalambi e toko.*
  Indonesia : Andi membeli baju di toko.

Dengan adanya kemiripan ini, maka dapat dilakukan proses yang umumnya terjadi di dalam bahasa Indonesia, misalnya POS *tagging* termasuk kelas kata (*tagset)* yang nantinya akan digunakan dalam penelitian ini. Penerapan metode yang dilakukan terbagi menjadi dua proses yaitu proses untuk pelatihan dan pengujian. Pada pelatihan, dimulai dengan melakukan perhitungan data latih menggunakan algoritme HMM, hasil dari perhitungan data latih disimpan pada *database* yang dijadikan HMM model dalam perhitungan algoritme Viterbi. Diagram alir dalam proses pelatihan data yang ditunjukkan pada **Error! Reference source not found.**.

Pengujian dimulai dengan *pre-processing* dan *parsing* pada data uji berdasarkan spasi dan tanda baca. Kemudian melakukan perhitungan algoritme Viterbi dengan menggunakan *database* yang menyimpan HMM model yang didapat dari data latih sebelumnya, hasilnya menjadi anotasi pada data uji yang disimpan pada dokumen hasil *POS Tagging*. Diagram alir dalam proses pengujian ditunjukkan pada **Error! Reference source not found.**.



Gambar 1. Diagram Alir Pada Proses Pelatihan



Gambar 2. Diagram Alir Pada Proses Pengujian

3.1. *Part-of-Speech Tagging*

Linguistik merupakan bidang keilmuan yang mempelajari tentang bahasa, salah satunya adalah morfologi yang merupakan bagian dari bahasa. Morfologi mempelajari bentuk dari sebuah kata dalam kalimat, sehingga kata tersebut bisa dibedakan pada beberapa kelas kata. Teknologi dalam perkembangan bahasa sangat membantu dalam proses morfologi linguistik, teknologi dalam mengembangkan bahasa biasa disebut *Natural Language Processing* (NLP). Dalam bahasa Inggris penentuan kelas kata pada suatu kalimat disebut *Part-of-Speech* (POS) *Tagging*. Penentuan kelas kata harus mengetahui ciri-ciri tiap kata pada sudut sintaksis, dan dikategorikan ke dalam kelas-kelas kata tertentu yang menjadi posisi penting dalam deskripsi suatu kalimat. Fungsi POS *tagging* adalah

menghilangkan kata ambigu pada suatu kalimat, terutama pada kata dengan lafal yang sama (homofon). Contoh dari kata bang dan bank yang mempunyai kelas kata yang berbeda tetapi dengan lafal yang sama (DINAKARAMANI, et al., 2014).

POS *tagging* merupakan proses dasar dalam mengolah bahasa pada NLP yang dilakukan secara otomatis oleh teknologi, contohnya digunakan pada proses dasar dalam aplikasi question answering (QA), analisis sentimen, dan named entity recognition (NER). POS *tagging* memiliki beberapa kelas kata tertentu yang menjadi posisi penting dalam deskripsi suatu kalimat pada bahasa tertentu, kelas kata tersebut biasanya dikenal dengan istilah tagset. POS *tagging* yang telah dilakukan pada beberapa bahasa, setiap bahasa memiliki tagset sendiri untuk digunakan pada POS *tagging* bahasa tersebut. Setiap tagset dirancang untuk menyesuaikan sintaksis dalam suatu kalimat, karena setiap bahasa memiliki sintaksis yang berbeda. Bahasa Indonesia memiliki *tagset* sendiri yang telah banyak dirancang, agar *tagset* sesuai dengan sintaksis Bahasa Indonesia.

Banyak *tagset* bahasa Indonesia yang telah dirancang oleh para peneliti di Indonesia, salah satunya dirancang oleh Dinakaramani dan kawan-kawan di Universitas Indonesia (2014). Penelitian perancangan *tagset* tersebut memiliki 2 tahapan, yaitu mendefinisikan POS *tagging* awal dengan *tagset* yang dirancang berdasarkan 2 *tagset* yaitu *tagset* pertama memiliki 17 kelas kata dan 8 tanda baca (ANDRIANI, et al., 2009). *Tagset* kedua memiliki 19 kelas kata (LARASATI, et al., 2011). Tahapan kedua adalah pengujian *tagset* POS *tagging* pada korpus bahasa Indonesia secara manual. Penelitian tersebut menghasilkan 2 output utama, yaitu *tagset* POS *tagging* Bahasa Indonesia yang telah dibuktikan pada korpus penelitian dan korpus Bahasa Indonesia yang terdiri dari ±250.000 token leksikal secara manual. Tabel 1 adalah hasil *tagset* Bahasa Indonesia pada penelitian yang dilakukan oleh Dinakaramani, et. al.

Penelitian ini merancang sebuah *tagset* untuk bahasa Madura yang dibuat oleh pakar dalam bahasa Madura yaitu Ibu Risalatul Maulidiyah, S.Pd. yang menjadi seorang guru sekolah bahasa Inggris dan Madura. Hasil rancangan *tagset* pada penelitian ini disesuaikan menurut *tagset* bahasa Indonesia pada penelitian yang dilakukan oleh Arawinda dan kawan-kawan di Universitas Indonesia pada tahun 2014. Tabel 2 merupakan hasil *tagset* bahasa Madura pada penelitian ini.

3.2. Hidden Markov Model

Hidden Markov Model (HMM) adalah algoritme *machine learning* yang menggunakan model statistik dengan menemukan *state* yang tersembunyi, data latih pada HMM membutuhkan klasifikasi kelas kata pada setiap kalimat (DIRGANTARA, et al., 2018). HMM memiliki *Markov chain* yang berfungsi untuk menghitung setiap probabilitas kemungkinan adanya *state* yang saling terhubung, dan mengamati *state* yang tersembunyi (*hidden tag)* pada sebuah kalimat. Gambar 3 merupakan contoh dari *Markov chain* untuk kata-kata, menunjukkan *state* dan transisi pada setiap *state*. Tanda panah pada setiap kata merupakan probabilitas transisi dari setiap kemungkinan *state*, jumlah probabilitas transisi pada setiap *state* harus berjumlah satu (1).

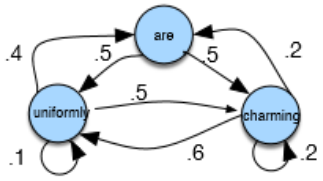
Tabel 1 Rancangan *Tagset* Bahasa Madura

No	Tagset	Contoh Bahasa Madura	Contoh Bahasa Indonesia
1	*Coordinating Conjunction* (CC)	*ban, namong, otaba, ghabay.*	dan, namun, atau, untuk.
2	*Cardinal Number* (CD)	*settong, duwa', tello', empa'.*	satu, dua, tiga, empat.
3	*Determiner* (DT)	*sabennya', sakone'na, sajumla, para, sa, ratosan.*	sebanyak, sedikatnya, sejumlah, para, se- , ratusan.
4	*Foreign Word* (FW)	*event, fashion, soft, opening, grand, final, the, visit, city, tour.*	*event, fashion, soft, opening, grand, final, the, visit, city, tour*.
5	*Preposition* (IN)	*moso, antara, e, ka, dari, kalaban, ghabay, dalem, padana, sareng.*	bersama, antara, di, ke, dari, untuk, dalam, seperti, dengan.
6	*Adjective* (JJ)	*senneng, seddhi, bhaghus, anyar, jhube', bheres, sala, abit, pae', raddhin.*	senang, sedih, bagus, baru, jelek, sehat, salah, lama, pahit, cantik.
7	*Modal and Auxiliary Verb* (MD)	*pasthe, bhakal, bisa, moso, kodhu.*	pasti, akan, bisa, dengan, harus.
8	*Negation* (NEG)	*enja', ta', jha', ella, ghita'.*	tidak, jangan, belum.
9	*Noun* (NN)	*naghara, katua, acara, paseser, tase'.*	negara, ketua, acara, pesisir, laut, tahun.
10	*Proper Noun* (NNP)	*Madura, Jhaba, Kabupaten Sumenep, Busyro.*	Madura, Jawa, Kabupaten Sumenep, Busyro.
11	*Classifier, Partitive and Measurement Noun* (NND)	*buwa, oreng, wisatawan, wisman*	buah, orang, wisatawan, wisman
12	*Demonstrative Pronoun* (PR)	*reya/neka/jareya, rowa/arowa, ka'dissa', jadiya*	ini, itu, sana, sini.
13	*Personal Pronoun* (PRP)	*bula/sengko', kabbhi, sengko' ban ba'na kabbhi, ba'na, aba'dhibi', dhibi'na.*	aku, mereka, kamu, kami, kita, dia.
14	*Adverb* (RB)	*sajan, keya, coma, ghun, segghut, ce', pagghun.*	makin, juga, Cuma, hanya, sering, sangat, tetap.
15	*Subordinating Conjunction* (SC)	*marena, se, mon, nangeng, samolaye, sajjheggha, tapeh, bakto, bila.*	setelah, yang, kalau, namun, sejak, tapi, waktu, ketika.
16	*Verb* (VB)	*ajelasaghi, acareta, ngoca', ngenep, namen,*	menjelaskan, bercerita, mengungkapkan, menginap, menanam.
17	*Question*	*apa, beremma, bila,*	apa, bagaimana,

No	Tagset	Contoh Bahasa Madura	Contoh Bahasa Indonesia
	(WH)	*sapa, edhimma, napa/ponapa.*	kapan, siapa, dimana, kenapa
18	*Punctuation* (Z)	“”, ( ), /, !, ?, .	
19	*Unknown* (X)	*statement*	

HMM memiliki pemodelan umum dalam penulisan sebuah permasalahan. Terdapat 5 komponen dalam menyusun pemodelan HMM, yaitu *observed state* (O), *hidden state* (Q), matriks probabilitas transisi (A), matriks probabilitas emisi (B) dan peluang keadaan awal (π) (JURAFSKY & MARTIN, 2019). Penjelasan 5 komponen sebagai berikut:

1. *Observed State* merupakan *state* yang dapat diamati pada *Markov chain*, direpresentasikan dengan simbol $O = O_1, O_2, O_3,\ldots O_t$. Permasalahan pada klasifikasi kelas kata yang dapat diamati adalah barisan kata pada setiap kalimat.
2. *Hidden state* adalah *state* yang tidak dapat diamati dan tersembunyi, direpresentasikan dengan simbol $Q = Q_1, Q_2,\ldots Q_n$. Pada permasalahan klasifikasi kelas kata yang tidak dapat diamati atau sebagai *hidden state* adalah kelas kata pada suatu kalimat.



Gambar 3. Contoh Markov Chain
Sumber: (JURAFSKY & MARTIN, 2019)

3. Matriks probabilitas transisi yang direpresentasikan dengan matriks $A = a_{01}, a_{02},\ldots a_{nn}$, dengan jumlah matriks probabilitas transisi sebanyak QxQ. Matriks tersebut berisi probabilitas transisi kelas kata (*tag*). Representasi probabilitas transisi tersebut ditunjukkan pada Persamaan 1. Semua peluang transisi akan disimpan pada matriks A dengan jumlah probabilitas transisi setiap *state* bernilai 1. Representasi matriks probabilitas transisi ditunjukkan pada Persamaan 2.

$$P_t(t_i|t_{i-1}) = \frac{C(t_{i-1},t_i)}{C(t_{i-1})} \quad (1)$$

$$A = \{a_{ij}\},\ \sum_{j=1}^{N} a_{ij} = 1\ \forall i \quad (2)$$

Keterangan:
P_t = Probabilitas transisi
t_i = kelas kata pada data latih
C = Penjumlahan suatu variabel
A = Matriks probabilitas transisi
a_{ij} = Representasi probabilitas transisi
N = Jumlah kelas kata

4. Matriks Probabilitas emisi adalah matriks untuk menyimpan peluang yang didapatkan dari perpindahan *state i* pada O_t (*Observed State*), matriks tersebut disimbolkan dengan matriks *B*. Probabilitas emisi merupakan nilai kemungkinan munculnya kata pada suatu tag tertentu. Perhitungan pada probabilitas emisi menggunakan *smoothing* untuk kata yang tidak memiliki kelas kata di dalam data latih. *Smoothing* yang digunakan adalah *Laplace smoothing*. Probabilitas emisi direpresentasikan pada Persamaan 3, dan disimpan pada matriks *B* sebanyak kelas kata dikalikan dengan banyaknya kata pada kalimat (QxO). Matriks Probabilitas emisi ditunjukkan pada Persamaan 4.

$$P_e(w_i|t_i) = \frac{C(t_i,w_i)+1}{C(t_i)+|V|} \quad (3)$$

$$B = b_i(O_t), \quad (4)$$

dengan

$$b_i(O_t) \geq 0\ \&\ \sum_{i=1}^{M} b_i(O_t) = 1$$

Keterangan:
P_e = Probabilitas emisi
w_i = Kata pada kelas tertentu
B = Matriks probabilitas Emisi

5. Probabilitas awal yang didapatkan dari semua kata pada data latih yang telah diklasifikasikan di setiap kelas kata. Klasifikasi kelas kata pada data latih dilakukan secara manual oleh pakar, kemudian menghitung probabilitas awal pada data latih dengan Persamaan 5.

$$\pi = P(q_1 = i), \quad (5)$$

dengan

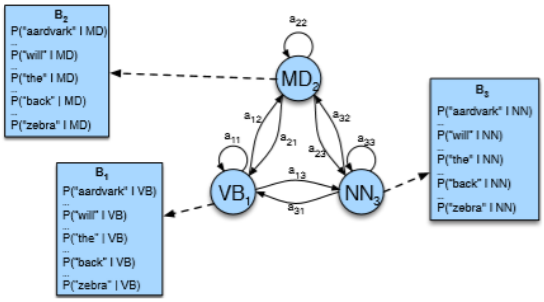
$$1 \leq i \leq N\ dan\ \sum_{i=1}^{N} \pi_i = 1$$

Keterangan:
π = Probabilitas awal.
q_1 = Kelas kata pada posisi pertama.
P = Menghitung peluang.

Hasil dari perhitungan algoritme HMM akan dimasukkan dalam pemodelan umum untuk perhitungan dasar algoritme Viterbi. Persamaan 6 adalah pemodelan umum yang digunakan pada perhitungan algoritme HMM, pemodelan tersebut direpresentasikan dengan simbol lambda (λ).

$$\lambda = (A, B, \pi) \quad (6)$$

Gambar 4 adalah gambaran pada matriks probabilitas transisi dan probabilitas emisi pada komponen sebelumnya, terdapat contoh 3 *tagset* untuk perhitungan HMM POS *tagging*. Simbol B merupakan matriks probabilitas emisi dan simbol a_{ij} adalah probabilitas transisi.



Gambar 4. Gambaran Matriks Probabilitas Transisi (A) dan Matriks Probabilitas Emisi (B)
Sumber: (JURAFSKY & MARTIN, 2019)

3.3. Algoritme Viterbi

Algoritme Viterbi merupakan *dynamic programming* yang berfungsi untuk mencari kemungkinan barisan yang tersembunyi (biasa disebut dengan Viterbi *path*) yang didapatkan dari barisan pengamatan kejadian (JURAFSKY & MARTIN, 2019). Beberapa proses yang ada pada algoritme Viterbi, yaitu tahap inisialisasi, rekursi dan terminasi. Persamaan tahap inisialisasi dinyatakan pada Persamaan 7 dan Persamaan 8, membuat matriks *backpointer* untuk menyimpan hasil *state* dan matriks Viterbi untuk menyimpan hasil perhitungan perulangan dengan nilai variabel s berada pada rentang nilai $1 \leq s \leq M$, variabel M adalah banyak kelas kata pada POS *tagging*, dan variabel "O" merupakan urutan observasi.

$$V_s(1) = \pi_s \,.\, b_s\,(O_1) \tag{7}$$

$$backpointer_s(1) = 0$$

Keterangan:
V_s = Nilai Viterbi.
π_s = Probabilitas awal pada kelas kata "s".
$b_s(O_1)$ = Probabilitas emisi dari kata pertama data uji pada kelas kata "s".

Persamaan tahap rekursi dinyatakan pada Persamaan 9 dan Persamaan 10, tahap ini dilakukan secara perulangan dengan nilai variabel s berada pada rentang $1 \leq s \leq M$ dan nilai variabel t berada pada rentang $2 \leq t \leq T$, dengan variabel T merupakan jumlah kata pada data uji.

$$V_s(t) = \max_{1 \leq s' < N} V_{s'}(t-1) * a_{s',s} * b_s(O_t)$$
$$backpointer_s(t) = \operatorname*{argmax}_{1 \leq s' < N} V_{s'}(t-1) * a_{s',s} * b_s(O_t) \tag{9}$$

Keterangan:
$a_{s',s}$ = Probabilitas transisi dari kelas kata $a_{s'}$ ke kelas kata "a_s".
$b_s(O_t)$ = Probabilitas emisi dari kata "t" data uji pada kelas kata "s".

Tahap rekursi dengan setiap *tagging* (Q_t), tahapan tersebut dinyatakan dalam Persamaan 11 dan persamaan 12. Hasil tahapan rekursi pada algoritme Viterbi akan didapatkan barisan *state* terbaik yang mengikQ^*_{rT} hasil dari .

$$P^* = \max_{1<i<M} [viterbi_s(T)] \tag{11}$$

$$Q^*_{rT} = \operatorname*{argmax}_{1<i<M} [viterbi_s(T)]$$

Keterangan:
P^* = *array* untuk menyimpan probabilitas *state* terbaik
Q^*_{rT} = *array* untuk menyimpan titik *state* terbaik

3.4 Evaluasi

Hasil klasifikasi dilakukan pengujian dengan cara mengukur beberapa parameter sistem yang telah dibuat dalam penelitian ini. Pengujian dilakukan dengan membandingkan dengan data yang telah ditandai oleh pakar disebut *gold standard*. Evaluasi pada pengujian ini menggunakan *confusion matrix multiclass*. Pada penggunaan *confusion matrix*, terdapat 4 istilah sebagai representasi hasil proses klasifikasi, yaitu *True Positive* (TP), *True Negative* (TN), *False Positive* (FP) dan *False Negative (*FP). Tabel 3 merupakan *confusion matrix multiclass*. Berikut perhitungan untuk mencari nilai TP, FP, FN, dan TN direpresentasikan pada Persamaan 14, 15, 16 dan 17 secara berturut-turut.

Tabel 2 Confusion Matrix Multiclass

Prediksi \ Aktual	A	B	C
A	E_{AA}	E_{AB}	E_{AC}
B	E_{BA}	E_{BB}	E_{BC}
C	E_{CA}	E_{BC}	E_{CC}

$$TP_i = E_{ii}$$
$$FP_i = \sum_i^k E_{ik} - TP_i \tag{13}$$
$$FN_i = \sum_i^k E_{ki} - TP_i$$
$$TN_i = \sum_i^k \sum_j^k E_{ij} - FN_i - FP_i - TP_i$$

Hasil dari matriks tersebut dapat menghitung nilai evaluasi dengan 2 cara, yaitu *micro average* dan *macro average*. *Macro average* menghitung matriks secara merata untuk setiap kelas dan mendapatkan nilai rata-rata dari semua kelas, sedangkan *micro average* menjumlahkan semua kontribusi semua kelas untuk menghitung rata-rata matriks (JURAFSKY & MARTIN, 2019). Parameter evaluasi yang akan dihitung adalah *accuracy*, *precision*, *recall* dan *f*-measure, direpresentasikan pada Persamaan 17, 18, 19, dan 20.

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{17}$$

$$Precision = \frac{TP}{TP+FP}$$

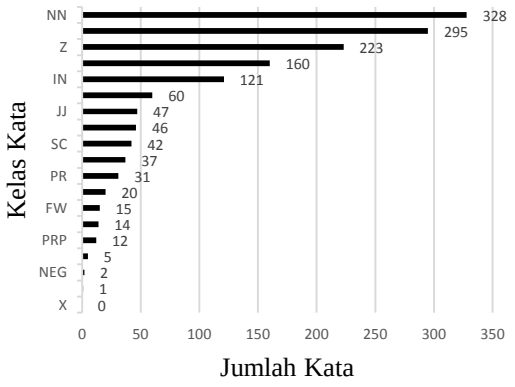
$$Recall = \frac{TP}{TP+FN}$$

$$F - Measure = \frac{2 \times Precision \times Recall}{(Precision+Recall)}$$

Penelitian klasifikasi kelas kata pada bahasa Madura menggunakan 70 data latih dan 35 data uji. Data yang didapatkan dari beberapa artikel di *website* tentang wisata di Madura. Artikel tersebut memakai bahasa Indonesia, sehingga artikel diterjemahkan dalam bahasa Madura oleh seorang pakar dalam bahasa Madura.

4. HASIL DAN PEMBAHASAN

Data yang digunakan ada jenis, yaitu data latih dan data uji. Data latih didapatkan dari kalimat-kalimat pada beberapa artikel tentang wisata di Madura, artikel tersebut menggunakan bahasa Indonesia. Data latih tersebut sebanyak 70 kalimat yang telah diterjemahkan dalam bahasa Madura dan telah diklasifisikasikan manual oleh seorang pakar yang bernama Risalatul Maulidiyah, S.Pd. Data latih ini terdiri 1459 kata dengan kelas kata sebanyak 19 kelas. Gambar 5 menjelaskan grafik persebaran kata pada setiap kelas.



Gambar 5 Jumlah Persebaran Kata Pada Setiap Kelas

Pengujian dilakukan pada 19 kelas kata untuk bahasa Madura, dengan menghitung nilai *accuracy*, *precision*, *recall* dan *f-measure*. Perhitungan evaluasi pada penelitian ini menggunakan *confusion matrix* dengan jenis *multiclass*, Tabel 3 merupakan *confusion matrix* dari perhitungan *POS tagging* pada semua kelas.

Tabel 4 terdapat nilai akumulasi TP, FP, FN, dan TN dari semua kelas kata. Hasil *confusion matrix* pada Tabel 4, dapat menghitung nilai *micro average* sebagai tabel 4

Tabel 3 *Confusion Matrix* Hasil *Micro Average*

Aktual / Prediksi	Positif	Negatif
Positif	499	237
Negatif	237	11539

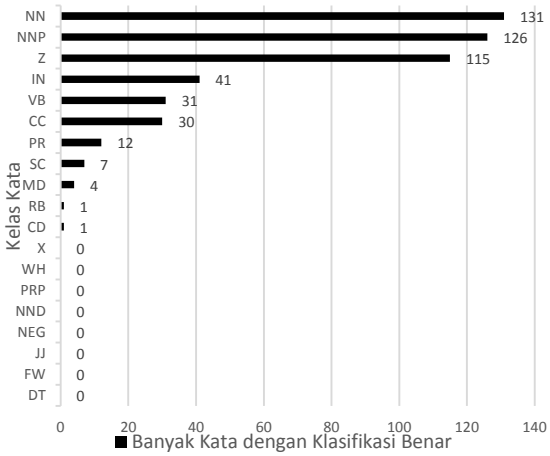
$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} = \frac{499+11539}{499+11539+237+237} = \frac{12038}{12512} = 0{,}96211636$$

$$Precision = \frac{TP}{TP+FP} = \frac{499}{499+237} = \frac{499}{736} = 0{,}67798913$$

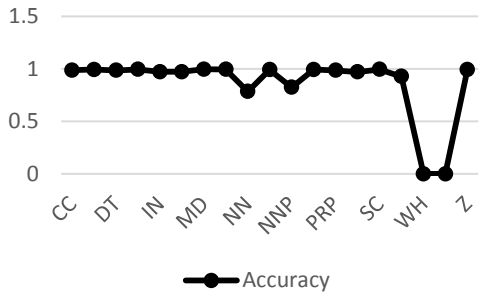
$$Recall = \frac{TP}{TP+FN} = \frac{499}{499+237} = \frac{499}{736} = 0{,}67798913$$

$$F-Measure = \frac{2\times Precision\times Recall}{(Precision+Recall)} = \frac{2\times 0{,}67798913\times 0{,}67798913}{0{,}67798913+0{,}67798913} = 0{,}67798470$$

Klasifikasi kelas kata pada 35 data uji yang terdiri dari 736 kata tanpa kelas kata terhadap kelas kata yang telah dirancang oleh seorang pakar bahasa Madura. Rancangan kelas kata tersebut terdapat 19 kelas kata, terdiri dari *coordinating conjunction* (CC), *cardinal number* (CD), *determiner* (DT), *foreign word* (FW), *preposition* (IN), *adjective* (JJ), *modal and auxiliary verb* (MD), *negation* (NEG), *noun* (NN), *proper noun* (NNP), *classifier*, *partitive and measurement noun* (NND), *demonstrative pronoun* (PR), *personal pronoun* (PRP), *adverb* (RB), *subordinating conjunction* (SC), *verb* (VB), *question* (WH), *punctuation* (Z) dan *unknown* (X). Evaluasi sistem dilakukan dengan menggunakan *confusion matrix multiclass* dengan beberapa parameter, yaitu *accuracy*, *precision* dan *recall*. Hasil klasifikasi sistem pada 736 kata, 499 kata yang berhasil diklasifikasi sesuai dengan hasil klasifikasi pakar. Gambar 6 merupakan banyaknya kata pada setiap kelas yang berhasil diklasifikasi dengan benar. Prediksi klasifikasi kelas dengan benar paling dominan terjadi pada kelas kata *noun* (NN), sedangkan kesalahan prediksi terjadi pada kelas *negation* (NEG).



Gambar 6 Jumlah Kata Pada Setiap Kelas dengan Prediksi Benar



Gambar 7 Nilai *Accuracy* Setiap Kelas Kata.

Gambar 7 merupakan nilai *accuracy* yang didapatkan dari setiap kelasnya. Nilai *accuracy* tertinggi sebesar 0.99592 pada kelas *modal and auxiliary verb* (MD), sedangkan nilai *accuracy* terendah adalah 0 (nol) pada kelas *question* (WH) dan *unknown* (X).

5. PENUTUP

Klasifikasi kelas kata pada penelitian ini terdapat 3 proses, yaitu *preprocessing*, *Hidden Markov Model* (HMM) dan klasifikasi dengan menggunakan algoritme Viterbi. Proses *preprocessing* dilakukan pada data latih dan data uji, proses ini melakukan *tokenizing*. Proses selanjutnya adalah perhitungan *Hidden Markov Model* (HMM) pada data latih. Perhitungan HMM terdapat 3 bagian, yaitu perhitungan probabilitas awal, probabilitas transisi dan probabilitas emisi, hasil perhitungan akan disimpan pada permodelan HMM untuk dijadikan dasar perhitungan pada algortime Viterbi. Proses terakhir yaitu klasifikasi kelas kata pada data uji dengan menggunakan algoritme Viterbi. Algoritme Viterbi memiliki 3 tahapan yaitu inisialisasi, rekursi, dan terminasi. Perhitungan tersebut menghasilkan nilai Viterbi pada setiap kelas kata. Penentuan kelas kata pada data uji didapatkan dari nilai maksimum dari nilai Viterbi pada setiap kata dalam data uji.

Penelitian ini menghasilkan kelas kata untuk bahasa Madura dan data latih yang didapatkan dari terjemahan beberapa artikel tentang wisata di Madura, terdapat 19 kelas kata dan 1459 kata pada data latih yang dibuat oleh seorang pakar bahasa Madura. Setiap kelas kata tersebut memiliki jumlah kata sesuai dengan data latih, yaitu *coordinating conjunction* (CC) sebanyak 46 kata, *cardinal number* (CD) sebanyak 60 kata, *determiner* (DT) sebanyak 14 kata, *foreign word* (FW) sebanyak 15 kata, *preposition* (IN) sebanyak 121 kata, *adjective* (JJ) sebanyak 47 kata, *modal and auxiliary verb* (MD) sebanyak 20 kata, *negation* (NEG) sebanyak 2 kata, *noun* (NN) sebanyak 328 kata, *proper noun* (NNP) sebanyak 295 kata, *classifier, partitive and measurement noun* (NND) sebanyak 5 kata, *demonstrative pronoun* (PR) sebanyak 31 kata, *personal pronoun* (PRP) sebanyak 12 kata, *adverb* (RB) sebanyak 37 kata, *subordinating conjunction* (SC) sebanyak 42 kata, *verb* (VB) sebanyak 160 kata, *question* (WH) sebanyak 1 kata, *punctuation* (Z) sebanyak 223 kata dan *unknown* (X). Kelas kata tersebut disesuaikan berdasarkan kelas kata untuk bahasa Indonesia.

Berdasarkan hasil evaluasi sistem, nilai tertinggi pada parameter evaluasi yaitu *accuracy* dengan nilai rata-rata sebesar 0,96211636, sedangkan nilai terendah pada parameter evaluasi sistem adalah *precision* dan *recall* dengan nilai rata-rata sebesar 0,67798913. Hal tersebut membuktikan bahwa sistem baik dalam klasifikasi kata sesuai kelas kata yang benar, karena mendapatkan nilai *accuracy* yang tinggi. Nilai rata-rata *recall dan precision* sebesar 0,67798913, membuktikan bahwa sistem dapat mengklasifikasikan kata pada beberapa kelas kata namun juga masih banyak yang salah akibat ketidaksesuaian dengan kelas kata sebenarnya. Hal ini dapat diakibatkan karena kurangnya data latih dan data yang bervariasi termasuk meliputi banyak domain.

Masih banyak peningkatan yang dapat dilakukan dalam penelitian ini. Dalam penelitian menggunakan algoritme Viterbi ini, sebaiknya menggunakan data latih dan data uji dengan jumlah banyak untuk setiap kelas kata, karena evaluasi akan semakin baik jika semakin banyak jumlah kata pada setiap kelas kata. Selain itu, banyaknya kelas kata yang digunakan sangat berpengaruh pada klasifikasi kelas kata untuk bahasa Madura. Contohnya hasil kelas kata pada data uji tidak sesuai dengan hasil pakar, karena pada data latih tidak ada kelas kata tersebut. Oleh karena itu, banyaknya kelas kata disesuaikan dengan data yang dipakai untuk data latih.

DAFTAR PUSTAKA

ANDRIANI, M., MANURUNG, R. & PISCELDO, F., 2009. *Statistical Based Part Of Speech Tagger for Bahasa Indonesia*. Singapore, s.n.

AZHAR, I. N., 2011. *Pengkajian Bahasa Madura Dulu Kini Dan Di Masa Yang Akan Datang.* Semarang, s.n.

DINAKARAMANI, A., RASHEL, F., LUTHFI, A. & MANURUNG, R., 2014. *Designing an Indonesian Part of speech Tagset and Manually Tagged Indonesian Corpus.* Kuching, 2014 International Conference on Asian Language Processing (IALP), pp. 66-69.

DIRGANTARA, M. Y. S., FAUZI, M. A. & PERDANA, R. S., 2018. Penerapan Named Entity Recognition Untuk Mengenali Fitur Produk Pada E-Commerce Menggunakan Rule Template Dan Hidden Markov Model. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 2(10), pp. 3912-3920.

EFFENDY, M. H., 2017. Interferensi Gramatikal Bahasa Madura Ke Dalam Bahasa Indonesia. *jurnal bahasa, sastra, dan pendidikan bahasa dan sastra Indonesia*, 4(1), pp. 1-19.

JURAFSKY, D. & MARTIN, J. H., 2019. *Speech and language processing.* 3nd ed. Silicon Valley: Stanford.

KAMAYANI, M., 2019. Perkembangan Part-of-Speech Tagger Bahasa Indonesia. *Jurnal Linguistik Komputasional*, II(2), pp. 34-38.

LARASATI, S. D., KUBON, V. & ZEMAN, D., 2011. *Indonesian Morphology Tool (MorphInd): Towards an Indonesian Corpus.* Zurich, Switzerland, s.n.

PRAMUDITA, H. R., UTAM, E. & AMBOROWATI, A., 2016. Pengaruh Part of Speech *Tagging* Berbasis Aturan dan Distribusi Probabilitas Maximum Entropy untuk Bahasa Jawa Krama. *Jurnal Buana Informatika*, VII(4), pp. 235-244.

PRIHANTINI, A., 2015. *Master Bahasa Indonesia.* Yogyakarta: B First.

RAHILAH, SOLIHIN, F. & RACHMAN, F. H., 2013. Aplikasi Penerjemah Bahasa Madura-Indonesia Dan Indonesia-Madura Menggunakan Free Contex Parsing Algorithm. *Jurnal Sarjana Teknik Informatika*, II(1), pp. 295-304.

RAMADHANTI, F., WIBISONO, Y. & SUKAMTO, R. A., 2019. Analisis Morfologi untuk Menangani Out-of-Vocabulary Words pada Part-of-Speech Tagger Bahasa Indonesia Menggunakan Hidden Markov Model. *Jurnal Linguistik Komputasional*, 2(1), pp. 6-12.

SETYANINGSIH, E. R., 2017. Part Of Speech Tagger untuk Bahasa Indonesia dengan Menggunakan Modifikasi Brill. *Dinamika Teknologi*, IX(1), pp. 37-42.

SHOLIHIN, A., SOLIHIN, F. & RACHMAN, F. H., 2013. Penerapan Modifikasi Metode Enhanced Confix Stripping Stemmer Pada Teks Berbahasa Madura. *Jurnal Sarjana Teknik Informatika*, II(1), pp. 305-314.

Halaman ini sengaja dikosongkan