

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Beberapa peneliti sudah membuat *tagger* Bahasa Jawa. Askhabi et al. (2020) membuat *tagger* Bahasa Jawa dengan pendekatan statistik dan menggunakan metode SVM (Support Vector Machine). Tujuan pembuatan *tagger* Askhabi et al. (2020) adalah membuat sistem pelabelan kelas kata dalam Bahasa Jawa. Namun, kelemahan *tagger* yang dibuat Askhabi et al. (2020) tidak bisa diakses oleh umum dan tidak dijelaskan secara terperinci *resource* yang digunakan.

Pramudita et al. (2016) juga membuat *tagger* dalam Bahasa Jawa. *Tagger* yang mereka buat menggunakan pendekatan statistik dengan metode Maximum Entropy dan *rule-based*. Namun, *resource* yang mereka aplikasikan memiliki akses yang tertutup dan tidak dijelaskan dengan detail. Pramudita et al. (2016) hanya menyebutkan bahwa mereka menggunakan kamus yang diambil dari buku Kosa Kata Bahasa Jawa yang ditulis oleh Rusyidi et al. (1985).

Adapun Pratama et al. (2020) membuat *tagger* Bahasa Jawa dengan pendekatan statistik. Metode yang digunakan mereka dalam pembuatan *tagger* adalah metode HMM (Hidden Markov Model). *Tagger* yang dibuat oleh Pratama et al. (2020) memiliki kelemahan yang sama dengan *tagger* yang dibuat oleh Askhabi et al. (2020) dan Pramudita et al. (2016) yakni *resource* dengan akses yang tertutup dan tidak bisa diakses oleh umum.

Meskipun sangat terbatas, pembuatan *tagger* Bahasa Jawa sudah dilakukan sebelumnya, diantaranya oleh Askhabi et al. (2020); Pramudita et al. (2016); dan Pratama et al. (2020) . Namun, *resources* dan sistem dari ketiga peneliti tersebut tidak dapat diakses oleh kalangan umum dan memiliki antarmuka yang tidak ramah pengguna. Penelitian Askhabi et al. (2020) menggunakan aplikasi Library Scikit-Learn untuk membuat *tagger*. Aplikasi Library Scikit-Learn memiliki antarmuka yang tidak ramah pengguna karena harus diakses dengan orang yang menguasai dasar-dasar bahasa pemrograman Python. Askhabi et al. (2020) menggunakan berita-berita daring sebagai *resource* penelitian. Namun, *resource* ini tidak dijelaskan lebih detail dan tertutup aksesnya.

Adapun penelitian dari Pramudita et al. (2016) menggunakan aplikasi OpenNLP untuk membuat *tagger*. . Aplikasi OpenNLP tidak memiliki antarmuka yang ramah pengguna karena pengguna disarankan menguasai bahasa pemrograman Java. Selain itu, mereka menggunakan artikel-artikel yang ada pada Jurnal BENING Volume 4, Juli 2015 sebagai data korpus atau *resource*-nya. Namun, *resource* yang dibuat oleh mereka memiliki akses yang tertutup

Pembuatan *tagger* dalam penelitian Pratama et al. (2020) menggunakan aplikasi datasaur.ai/beta. Aplikasi tersebut adalah aplikasi pembuatan anotasi yang berbasis situs web. Jika dibandingkan dengan aplikasi Library Scikit-Learn dan OpenNLP, datasaur.ai/beta memiliki antarmuka yang lebih ramah pengguna. Namun, beberapa fitur lanjut seperti membutuhkan pengetahuan Python. Pratama et al. (2020) menggunakan korpus dari [berita solopos.com/jagad-jawa](https://solopos.com/jagad-jawa) sebagai

resource. Namun, korpus yang sudah dianotasi oleh *tagger* tersebut tidak secara eksplisit dapat diakses untuk umum.

Secara garis besar, *tagger* dapat dibuat menggunakan metode data-driven atau rule-based, yang akan dijelaskan pada 2.2.1. Pada bagian ini, peneliti berfokus pada performa *tagger* yang sudah ada. *Tagger* Bahasa Jawa yang sudah dibuat oleh para peneliti seperti Askhabi et al. (2020); Pramudita et al. (2016); dan Pratama et al. (2020) memiliki tingkat akurasi yang berbeda. *Tagger* Bahasa Jawa dengan metode Hidden Markov Model yang dibuat oleh Pratama et al. (2020) memiliki tingkat akurasi 92.6%. Namun, *tagger* tersebut terbatas hanya untuk Bahasa Jawa *ngoko* saja. Sedangkan, Pramudita et al. (2016) membuat *tagger* Bahasa Jawa dengan berbasis aturan dan *data-driven*. Namun, *tagger* mereka hanya untuk Bahasa Jawa *krama* dengan tingkat akurasi 97.67%. Adapun, Askhabi et al. (2020) membuat *tagger* Bahasa Jawa dengan metode SVM (Support Vector Machine). Namun, tingkat akurasi *tagger* SVM kurang dari 90% yakni 77%. Meski berbeda metode, pada prinsipnya, semua *tagger* di atas bersifat probalistik, metode yang umum digunakan pada ranah ilmu komputer.

Pembuatan *tagger* Bahasa Jawa menggunakan metode *rule-based* atau pendekatan linguistik masih sangat terbatas dalam Bahasa Jawa. Adapun pembuatan *tagger rule-based* juga diimplementasikan untuk bahasa lain. Prihantoro (2021b) membuat *tagger* untuk Bahasa Indonesia dengan akurasi 99%. Voutilainen (1997) membuat *tagger* untuk Bahasa Inggris dengan akurasi 99.55%. Meskipun keduanya menggunakan metode *rule-based*, tingkat akurasi kedua *tagger* tersebut tidak jauh berbeda dengan *tagger* pada umumnya seperti CLAWS Garside

(1987) dan Penn Treebank Tagger (Marcus, 1993). Adapun Lin (2010) membuat *tagger* secara parsial yakni hanya membuat *tagger* untuk kelas kata nomina saja dalam Bahasa Mandarin. Namun, tingkat akurasi tidak disebutkan.

Menurut Kridalaksana (1982) afiksasi adalah hasil ataupun proses pembentukan kata dengan menambahkan afiks pada dasar atau akar kata. Kajian afiksasi dalam Bahasa Jawa sudah dilakukan oleh para peneliti sebelumnya. Nabilla & Anekawati (2025) mengkomparasikan afiksasi pada Bahasa Jawa, Bahasa Indonesia dan Bahasa Inggris. Menurut Nabilla & Anekawati (2025) afiks Bahasa Jawa memiliki tingkat produktivitas yang lebih tinggi dibandingkan dengan Bahasa Inggris dan Bahasa Indonesia. Bahasa Jawa memiliki lebih banyak jenis afiks dibandingkan Bahasa Inggris. Penelitian Nabilla & Anekawati (2025) tidak menjelaskan secara detail frekuensi kemunculan setiap afiks.

Asri et al. (2025) membahas produktivitas afiksasi Bahasa Jawa pada sebuah cerita pendek. Berdasarkan hasil penelitian Asri et al. (2025), konfiks menjadi afiksasi yang memiliki paling beragam jenisnya. Mereka menemukan 9 jenis konfiks dalam cerita pendek. Jenis afiksasi yang ditemukan oleh mereka masih terbatas karena diambil dari satu cerita pendek. Adapun Yanti & Surana (2021) meneliti produktivitas afiksasi dalam novel *Wedhus Gembel*. Penelitian mereka menyebutkan bahwa proses prefiksasi memiliki paling banyak jenis afiks dibandingkan proses afiksasi lain. Namun, penelitian mereka tidak menjelaskan frekuensi kemunculan setiap afiks.

Peneliti lain yakni Nalendra et al. (2021) melakukan komparasi produktivitas afiksasi pada verba dan nomina antara Bahasa Jawa dan Bahasa

Indonesia. Penelitian Nalendra et al. (2021) menunjukkan adanya perbedaan afiksasi Bahasa Jawa dengan Bahasa Indonesia. Sebagai contoh, prefiks TAK- hanya muncul di Bahasa Jawa. Penelitian Nalendra et al (2021) terbatas hanya memberikan contoh-contoh afiksasi. Berdasarkan kajian-kajian sebelumnya, produktivitas afiksasi verba Bahasa Jawa belum dipetakan secara eksplisit. Sehingga, Peneliti akan mengkaji distribusi produktivitas afiksasi pada Verba Bahasa Jawa. Sumber data diambil dari korpus Javanese UD

Ada dua kelemahan pada *tagger* Bahasa Jawa yang sudah dibuat. Pertama, *tagger* Bahasa Jawa yang sudah dibuat tidak memiliki *resource* yang terbuka. Kedua, *tagger* Bahasa Jawa yang sudah dibuat belum memiliki antarmuka yang ramah pengguna. Berdasarkan hal tersebut, peneliti akan mengisi celah kelemahan dengan membuat *tagger* yang memiliki *resource* dengan akses terbuka dan mempunyai antarmuka yang ramah pengguna. Berdasarkan penelitian sebelumnya distribusi produktivitas afiksasi belum diuraikan secara jelas. Oleh sebab itu, peneliti juga membahas distribusi produktivitas afiksasi korpus pada UD Javanese.

2.2 Landasan Teori

2.2.1 *Tagger*

Tagging atau pembuatan anotasi otomatis dilakukan oleh bantuan komputer dan perangkat lunak. *Tagging* otomatis memiliki kelebihan yakni bisa melakukan anotasi teks dengan skala yang besar (Voutilainen, 1999). Menurut Voutilainen, (1999) berikut langkah-langkah pembuatan *tagger* otomatis yakni:

- 1) *tokenisasi* yakni tahap memotong teks menjadi unit lebih kecil sesuai yang dibutuhkan peneliti

- 2) *lookup module* yang terdiri dari dua tahap yakni leksikon dan *guesser module*. Tahap leksikon adalah tahap analisis morfosintaksis yang mengidentifikasi daftar kata dan afiksasi. Tahap *guesser module* yakni tahap menganalisis daftar kata yang tidak ada di dalam leksikon
- 3) *disambiguation module* yakni tahap menentukan *tagger* yang tepat untuk leksikon yang memiliki ambiguitas atau makna ganda

Jika pembuatan anotasi otomatis dibantu oleh komputer atau perangkat lunak, maka *tagging* atau anotasi secara manual dilakukan oleh manusia (annotator). *Tagging* manual layak diimplementasikan untuk teks dengan skala yang kecil. Kelebihan *tagger* manual yakni lebih akurat karena anotasi dilakukan satu per satu oleh manusia (Voutilainen, 1999).

Pendekatan dalam pembuatan *tagger* dibagi dalam tiga jenis yakni (1) pendekatan linguistik atau *rule-based*, (2) pendekatan data-driven, dan (3) gabungan dari keduanya. *Tagger* yang dibuat dengan pendekatan linguistik membutuhkan *rule* atau rumus. Oleh sebab itu, *tagger* dengan pendekatan linguistik disebut juga dengan *tagger rule-based*. Sistem *tagger* dengan pendekatan linguistik adalah sistem *tagger* yang aturan-aturannya dibuat oleh *grammarians* secara manual atau disebut aturan heuristik. Aturan sistem pendekatan linguistik bersumber dari generalisasi pengamatan teks dalam skala besar, pengetahuan *grammarians* pada bidang bahasa, dan tata bahasa deskriptif (Voutilainen, 1999). Selain membutuhkan rumus, pembuatan *tagger* dengan pendekatan linguistik membutuhkan leksikon dan sistem konfigurasi. Leksikon dibutuhkan untuk mengetahui kata dasar dan afiksasi yang digunakan dalam sebuah kata. Selain itu,

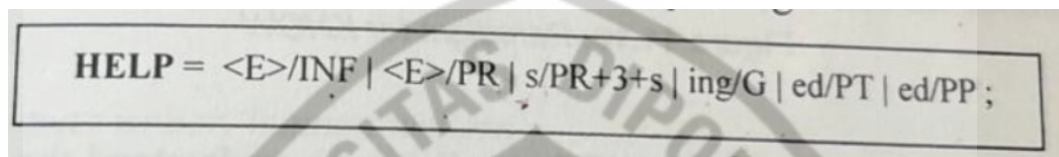
leksikon akan digunakan sebagai kamus dan menjadi *resource* dalam pembuatan *tagger*. Sistem konfigurasi dibutuhkan untuk menyusun urutan rumus dan leksikon dalam menciptakan *tagger*. Sistem konfigurasi memiliki tujuan untuk menghindari ambiguitas anotasi.

Tagger dengan pendekatan *data-driven* adalah sistem yang aturan-aturannya dibuat secara otomatis yakni berdasarkan statistik atau probabilitas dari teks dengan skala besar seperti HMM dan kolokasi matriks (Voutilainen, 1999). Sehingga, sistem *data-driven* membutuhkan korpus yang sudah dianotasi sebagai dasar aturan pembuatan sistem *tagger*. Sebagai ilustrasi, pada *tagger* morfem Bahasa Indonesia yang dibuat oleh Prihantoro (2021a), proses pembentukan kata dikodifikasi dan diterjemahkan menjadi tata bahasa komputasi. Tata bahasa ini digunakan sebagai salah satu *resource* untuk melakukan anotasi. Prihantoro (2024) juga membuat *tagger* menggunakan pendekatan probabilitas, dalam hal ini, tidak diperlukan tata bahasa komputasi, namun dibutuhkan korpus yang sudah dianotasi sebelumnya. Sistem akan mengonversi tata bahasa secara otomatis.

2.2.2 Proses Pembuatan Tata Bahasa untuk *Tagger*

Pada *tagger* yang dikembangkan menggunakan metode *rule-based*, tata bahasa menjadi penting. Perhatikan **Gambar 4**, gambar ini menunjukkan contoh format rumus tata bahasa morfologi dari kata *help* ‘membantu’ yang dibuat oleh Silberztein (2016). Pada **Gambar 4** kata *help* ‘membantu’ memiliki enam rumus tata bahasa. Salah satu rumusnya adalah $\langle E \rangle / \text{INF}$ yang terdiri dari $\langle E \rangle$ dan INF. Tanda $\langle E \rangle$ atau *empty string* digunakan untuk menyatakan bahwa kata kerja *help* ‘membantu’

tidak diawali, diikuti, atau disisipi oleh afiksasi. Tanda INF untuk menyatakan infinitif. Verba *help* ‘membantu’ tanpa afiksasi adalah verba infinitif. Contoh lain rumus tata bahasa *help* ‘membantu’ adalah ed/PT. Verba *help* ‘membantu’ yang disertai dengan sufiks -ED memiliki fungsi sebagai verba yang digunakan dalam bentuk lampau atau PT (past tense).



$$\text{HELP} = \langle E \rangle / \text{INF} \mid \langle E \rangle / \text{PR} \mid s / \text{PR} + 3 + s \mid \text{ing} / \text{G} \mid \text{ed} / \text{PT} \mid \text{ed} / \text{PP} ;$$

Gambar 4. Rumus Tata Bahasa Kata *help*

2.2.3 Verba Bahasa Jawa

Menurut Poedjosoedarmo (2015), proses produktivitas kata-kata dalam Bahasa Jawa melalui morfem bebas dan morfem terikat. Morfem terikat pada Bahasa Jawa salah satunya berupa afiksasi atau imbuhan. Proses Afiksasi atau imbuhan berupa awalan (prefiks), sisipan (infiks), akhiran (sufiks), dan gabungan dari imbuhan-imbuhan.

2.2.3.1 Afiksasi

Afiksasi adalah proses pembentukan kata dengan menambahkan afiks pada bentuk dasar (Lieber, 2016). Afiks adalah *bound morpheme* atau morfem terikat yang biasanya menempel pada *free morpheme*. Suatu morfem sering direalisasikan dengan bentuk yang berbeda. Sebagai contoh morfem N- (*underlying form*) bisa direalisasikan menjadi bentuk morf seperti ny-, ng-, atau m- (*surface form*). Sebagai contoh, verba *ombe* ‘minum’ jika ditambahkan morfem N- akan direalisasikan menjadi kata dengan morf ng-, *ngombe* ‘meminum’. Menurut Lieber (2016),

underlying form adalah bentuk dasar morfem. Sedangkan, *surface form* adalah realisasi dari *underlying form*.

Penggabungan afiks dengan kata dasar dapat menyebabkan perubahan fonem atau bunyi (Uhlenbeck, 1978). Proses morfologi ini sering disebut istilah morfofonemik. Sebagai ilustrasi, morfem N- apabila menempel dengan verba *cekel* ‘bawa’ direalisasikan menjadi verba *nyekel* ‘membawa’ dengan morf. Morfem N- direalisasikan menjadi morf ny-. Bunyi /c/ pada verba *cekel* mengalami pelesapan.

Afiksasi Bahasa Jawa mirip dengan afiksasi Bahasa Indonesia, sebagai contoh prefiks N- dalam Bahasa Jawa memiliki fungsi yang sama dengan prefiks MEN- dalam Bahasa Indonesia. Adapun prefiks DI- pada Bahasa Jawa memiliki fungsi yang sama dengan prefiks DI- pada Bahasa Indonesia. Namun, Bahasa Jawa memiliki afiksasi yang bersifat subjungtif dan imperatif seperti pada afiksasi -A, -NA-, dan -EN. Contoh penggunaan afiksasi yang bersifat imperatif sebagai berikut:

- (1) *Tutupna jendela kamar!*
tutup-na jendela kamar!
tutup-IMP jendela kamar!
‘Tutup jendela kamar!’

Verba *tutupna* ‘tutup’ pada nomor (1) memiliki prefiks –NA. Sufiks ini memiliki fungsi sebagai penanda imperatif pada Bahasa Jawa. Verba *tutupna* ‘tutup’ dalam Bahasa Indonesia diterjemahkan menjadi ‘tutup’. Namun, prefiks -KAN dalam Bahasa Indonesia tidak hanya memiliki fungsi dalam kalimat imperatif, tetapi juga memiliki fungsi dalam kalimat kausatif dan benefaktif.

Verba Bahasa Jawa, dari banyaknya morfem pembentuk, dapat dibagi menjadi dua jenis monomorfemis dan polimorfemis (Subroto et al., 1991). Verba monomorfemis adalah verba yang terdiri dari satu morfem, belum atau tidak

mengalami proses afiksasi seperti *ombe* ‘minum’ atau *tuku* ‘beli’. Sebaliknya, verba polimorfemis adalah verba yang sudah mengalami proses agregasi morfemis termasuk afiksasi seperti *diombe* ‘diminum’ atau *dituku* ‘dibeli’. Ada empat jenis afiksasi yakni prefiksasi, sufiksasi, infiksasi, dan konfiksasi.

Proses prefiksasi verba dilakukan dengan cara menambahkan imbuhan di awal atau di depan kata dasar (Wedhawati et al., 2006). Sebagai contoh, Menurut (Uhlenbeck, 1978), verba *diobong* ‘dibakar’ memiliki morfem inti atau kata dasar *obong* ‘bakar’. Kemudian, kata dasar verba ini mendapatkan awalan atau prefiks DI-. Prefiks DI- pada verba *diobong* ‘dibakar’ menandakan sebagai verba pasif.

Proses sufiksasi verba dilakukan dengan cara menambahkan imbuhan di akhir atau di belakang kata dasar. Sebagai contoh, verba *maosaken* ‘membacakan’ memiliki kata dasar *maos* ‘membaca’. Kemudian, kata dasar verba ini mendapatkan akhiran atau sufiks -AKEN. Sufiks -AKEN pada verba *maosaken* ‘membacakan’ menandakan sebagai verba aktif (Wedhawati et al., 2006).

Verba yang berinfiks adalah verba yang terbentuknya menyisipkan imbuhan di dalam kata dasar. Sebagai contoh, verba *tinulis* ‘ditulis’ memiliki kata dasar *tulis* ‘tulis’. Kemudian, kata dasar tersebut mendapatkan sisipan -UM-, sehingga menjadi kata *tinulis* ‘ditulis’.

Verba yang berinfiks adalah verba yang terbentuknya menambahkan imbuhan di depan dan di belakang kata dasar. Sebagai contoh, verba *dicritak(a)ke* ‘diceritakan’ memiliki kata dasar *crita* ‘cerita’. Kemudian, kata dasar tersebut mendapatkana awalan DI- dan akhiran -(A)KE.

Selain dilihat dari segi morfologis, verba Bahasa Jawa dapat dilihat dari segi fungsinya yakni berdasarkan diatesis dan *speech level* (tingkat tutur). Menurut Wedhawati et al. (2006), contoh afiksasi yang menandai verba aktif dalam Bahasa Jawa adalah N-, A-, MA-, MER-, KUMA, -UM-, dan KAMI-/EN, serta afiksasi yang menandai verba pasif contohnya DI-, DI-/I, KE-, dan KE-/AN. Diatesis dalam Bahasa Jawa cenderung ditandai dengan afiksasi. Verba *ngombe* ‘meminum’ (2) dan *diombe* ‘diminum’ (3) dari kata dasar *ombe* ‘minum’ memiliki diatesis yang berbeda. Kata *ngombe* ‘meminum’ pada contoh (2) memiliki diatesis aktif ditandai dengan prefiks NG-. Sedangkan, *diombe* ‘diminum’ pada contoh (3) memiliki diatesis pasif ditandai dengan prefiks DI-.

- (2) *Aku ngombe es teh.*
aku ng-ombe es teh
 1SG ACV-minum es teh
 ‘aku meminum es teh.’
- (3) *Es teh diombe karo aku.*
es teh di-ombe karo aku
 es teh PASS-minum oleh 1SG
 ‘es teh diminum oleh aku’

2.2.3.2 Diatesis

Menurut Sudaryanto et al. (1991) diatesis adalah kategori gramatikal yang menjelaskan hubungan antara subjek dengan perbuatan yang dinyatakan dengan verba pada suatu klausa. Bahasa Jawa memiliki empat jenis diatesis yakni aktif, pasif, resiprokal, dan reflektif. Pertama, pada diatesis aktif, subjek memiliki peran sebagai pelaku (agent) dan objek sebagai penderita (patient). Kedua, pada diatesis pasif, subjek memiliki peran sebagai penderita (patient). Ketiga, diatesis resiprokal

memiliki fungsi untuk menunjukkan hubungan timbal balik atau saling. Keempat, diatesis refleksif, diatesis yang melibatkan satu pihak. Subjek dalam diatesis refleksif berfungsi sebagai pelaku sekaligus penderita. Sehingga, subjek melakukan tindakan untuk dirinya sendiri.

2.2.3.3 Tingkat Tutur

Bahasa Jawa mengenal tingkat tutur (*unda-usuk*) atau *speech level*. Menurut Poedjosoedarma et al. (1979) terdapat 3 jenis tingkat tutur dalam Bahasa Jawa yakni tingkat tutur *ngoko*, *krama*, dan *madya*. Tingkat tutur *ngoko* digunakan apabila jarak penutur dan mitra tutur dekat dan akrab. Selain itu, orang-orang dengan kelas sosial yang lebih tinggi dianggap berhak menggunakan tingkat tutur ini, sebagai contoh, majikan terhadap asisten rumah tangga dan guru terhadap muridnya.

Tingkat tutur *krama* digunakan apabila penutur memiliki rasa segan terhadap mitra tutur. Biasanya, penutur menggunakan tingkat tutur ini kepada orang yang tidak dikenal dan memiliki status sosial yang lebih tinggi. Sebagai contoh, murid berbicara kepada gurunya menggunakan tingkat tutur *krama* karena status guru lebih tinggi.

Tingkat tutur *madya* berada di tengah-tengah antara *krama* dan *ngoko*. Penutur menggunakan tingkat tutur ini untuk menunjukkan sikap sopan, tetapi tingkat sopan *madya* tidak setinggi pada tingkat tutur *krama*. Tingkat tutur ini bisa digunakan apabila penutur berbicara terhadap tetangga yang tidak terlalu dekat atau kepada rekan kerja sebaya. Tingkat tutur terjadi karena adanya perbedaan tingkat sosial. Beberapa kelompok masyarakat harus dihormati, sehingga penutur diharuskan menggunakan tingkat tutur *krama*.