Implementasi *Information Gain* sebagai *Feature Selection* pada *Word Sense Disambiguation* Bahasa Indonesia dengan Teknik Klasifikasi *Decision List*

Sakti Dewantoro[1], Anisa Herdiani[2], Diyas Puspandari[3]

[1,2,3]Fakultas Informatika, Universitas Telkom, Bandung
[1]saktyd@students.telkomuniversity.ac.id, [2]anisaherdiani@telkomuniversity.ac.id,
[3]diyaspuspandari@telkomuniversity.ac.id

Abstrak

***Word sense disambiguation* (WSD) merupakan metode pencarian makna asli dari sebuah kata ambigu dalam konteks tertentu. Berbagai jenis *classifier* dapat digunakan di WSD, salah satunya adalah pendekatan *supervised* dengan metode *decision list.* Metode klasifikasi *decision list* mampu menjadi kompetitor terbaik pada senseval 1 diantara partisipan *supervised*. Pendekatan *supervised,* tidak diragukan lagi bekerja lebih baik daripada pendekatan lain. Namun, pendekatan ini perlu mengandalkan banyaknya ketersediaan korpora yang digunakan untuk membuat dataset. Semakin banyak korpora yang digunakan maka semakin banyak atribut yang terdapat pada dataset. Banyaknya atribut yang diolah *classifier* akan berdampak pada menurunnya kinerja *classifier. Feature selection* dapat digunakan untuk mengoptimalkan kinerja *classifier* dengan cara mengurangi atribut yang kurang relevan pada dataset. *Information gain* merupakan salah satu seleksi fitur terbaik dibanding seleksi fitur lainya pada penelitian yang telah dilakukan oleh Yang dan Pederson, Tan dan Yang serta Forman dalam hal klasifikasi dokumen. Karena keunggulan seleksi fitur *information gain* dan keunikan metode klasifikasi *decision list* tersebut, penelitian ini mengimplementasikan *information gain* sebagai seleksi fitur pada WSD bahasa Indonesia menggunakan metode klasifikasi *decision list.* Hasil penelitian ini, *information gain* dapat meningkatkan akurasi dengan selisih 0.5% dan selisih presisi 1.3% pada pengujian range *collocation* 2, serta selisih akurasi 0.3% dan selisih presisi 0.7% pada pengujian range *collocation* 3.**

Kata kunci : word sense disambiguation, decision list, feature selection, information gain

Abstract

Word sense disambiguation (WSD) is a method of searching for the original meaning of an ambiguous word in a particular context. Various types of classifiers can be used in WSD, one of which is the supervised approach to the decision list method. The decision list classification method is able to be the best competitor for the senses which 1 of the participants is supervised. Supervised approach, has no doubt works better than other approaches. However, this approach needs to rely on the large availability of korpora used to create datasets. The more korpora used, the more attributes are found in the dataset. The number of attributes processed by the classifier will affect the performance of the classifier. Feature selection can be used to reduce attributes that are less relevant to the dataset. Information gain is one of the best feature selection compared to other feature selection in the research conducted by Yang and Pederson, Tan and Yang and Forman. Feature selection can be used to optimize classifier performance due to the advantages of information gain feature selection and the uniqueness of the decision list classification method. This study implements information gain as a feature selection on Indonesian WSD using the decision list classification method. The results of this study, information gain can improve 0.5% accuracy and 1.3% precision in range collocation 2 testing and 0.3% accuracy and 0.7% precision in range collocation 3 testing.

Keywords: word sense disambiguation, decision list, feature selection, information gain

1. Pendahuluan

Latar Belakang

Word sense disambiguation adalah metode pencarian makna asli dari sebuah kata ambigu, kata polisemi dalam konteks tertentu [9]. Salah satu metode klasifikasi dalam *word sense disambiguation* yaitu *decision list* yang termasuk pendekatan *supervised* [8]. Berbagai pedekatan *supervised* lainnya misalnya *decision tree, naïve bayes, neural network* dan sebagainya. Diantara partisipan *supervised system* dalam senseval 1, *decision list* mampu menjadi kompetitor terbaik dengan tingkat akurasi 78.9% [17].

Pendekatan *supervised,* memiliki performa yang lebih baik daripada pendekatan lain [8]. Namun, pendekatan ini perlu mengandalkan banyaknya ketersediaan korpora yang digunakan untuk membuat *dataset*. Pada Kamus Besar Bahasa Indonesia (KBBI) terdapat kata polisemi sebanyak ± 9475 kata [14]. Apabila semua kata polisemi pada KBBI digunakan untuk membuat dataset maka perlu membutuhkan ketersediaan korpora yang banyak. Semakin banyak korpora yang digunakan maka semakin banyak atribut yang terdapat pada dataset. Banyaknya atribut yang diolah oleh *classifier* dapat berdampak pada menurunnya kinerja *classifier* [7].

Feature selection merupakan bagian penting untuk mengoptimalkan kinerja dari *classifier* [1]. *Feature selection* bekerja dengan mengurangi ruang fitur besar, seperti mengeliminasi atribut yang kurang relevan [2]. Penggunaan algoritma *feature selection* yang tepat dapat meningkatkan akurasi *classifer* [3]. *Feature selection* dapat dibedakan menjadi dua jenis yaitu tipe *filter* dan tipe *wrapper.* Contoh tipe *filter* antara lain *information gain, chi-square* dan *log-likelihood ratio*. Sedangkan contoh tipe *wrapper* yaitu *forward selection* dan *backward elimination.* Dari penelitian tentang membandingkan beberapa algoritma *feature selection* seperti yang dilakukan oleh Yang dan Pedersen [4], Forman [3] serta Tan dan Zang [5], didapatkan kesimpulan bahwa algoritma information gain yang paling baik pada kasus klasifikasi dokumen.

Dengan akurasi yang didapatkan metode klasifikasi *decision list* pada senseval 1 [17]. Dan manfaat penggunaan *feature selection* terutama algoritma *information gain* yang menjadi algoritma terbaik pada perbandingan algoritma *feature selection* penilitian sebelumnya[4],[3] dan [5]. Maka pada penelitian ini, penulis menerapkan *feature selection* algoritma *information gain* pada sistem WSD bahasa Indonesia untuk membedakan makna dari kata ambigu menggunakan metode klasifikasi *decision list.*

Topik dan Batasan

Pendekatan *supervised* perlu mengandalkan banyaknya ketersediaan korpora yang digunakan untuk membuat dataset. Semakin banyak korpora yang digunakan maka semakin banyak atribut yang terdapat pada dataset. Banyaknya atribut yang diolah oleh *classifier* akan berdampak pada menurunnya kinerja *classifier* [7]. Atribut yang digunakan dalam klasifikasi *decision list* adalah atribut yang diambil dari dataset berupa *feature vector. Feature vector* terdiri dari *collocation feature* dan *co-ocurrence feature.* Banyaknya atribut yang diambilpun dipengaruhi oleh pengujian range *collocation.* Pada umumnya penggunaan range *collocation* berkisar (±2). Maka dari itu dibutuhkan *feature selection* untuk mengurangi ruang fitur, seperti mengeliminasi atribut yang kurang relevan. Apabila *feature selection* digunakan dengan tepat dapat meningkatkan akurasi hasil klasifikasi.

Pada penelitian ini, dataset yang digunakan adalah data yang diambil dari koran harian kompas. Data tersebut terdiri dari daftar kalimat yang mengandung kata daun, buah, kursi, bulan dan akar dan sudah ditentukan *sense*-nya terlebih dahulu. Setiap kalimat dalam dataset hanya mengandung satu sense saja.

Tujuan

Tujuan yang diharapkan dari pembuatan tugas akhir ini yaitu untuk mengetahui besar akurasi dan presisi metode klasifikasi *decision list* dengan menggunakan *feature selection* algoritma *information gain* saat membedakan *sense* kata pada kalimat bahasa Indonesia.

Organisasi Tulisan

Pada bagian studi terkait akan menjelaskan landasan teori yang berkaitan dengan penelitian ini. Pada bagian sistem yang dibangun, menjelaskan alur proses pembuatan sistem secara rinci. Pada bagian evaluasi menjelaskan hasil pengujian sistem yang dibangun dan analisa hasil pengujian sistem. Pada bagian kesimpulan, menjelaskan kesimpulan dari tujuan penelitian dengan hasil penelitian.

2. Studi Terkait

2.1 Word Sense Disambiguation

Word sense disambiguation (WSD) merupakan sebuah teknik untuk menemukan arti atau makna sesungguhnya dari suatu kata polisemi [6]. Penelitian mengenai WSD sudah dimulai pada tahun 1940 [10]. Hal yang menjadi tantangan pada penelitian WSD adalah bagaimana mesin dapat mengenali dan mengetahui arti bahasa manusia dengan keberagaman makna kata. Karena menurut Zipf, dalam teori "*Law of Meaning"* penelitian tahun 1949 menjelaskan relasi antara kata yang sering diucapkan cenderung memiliki lebih banyak makna dibanding kata yang jarang diucapkan. WSD memiliki dua varian tugas yaitu sampel leksikal (atau WSD yang ditargetkan) dan semua kata WSD [8]. Yang dimaksud sampel leksikal adalah sistem diperlukan untuk mendisambiguasikan suatu set kata target yang terbatas biasanya terjadi satu per kalimat. Sedangkan semua kata, sistem melakukan disambiguasi semua kata pada teks. Dalam pengaplikasiannya, umumnya WSD digunakan pada mesin penerjemah, mesin tanya jawab, pengekstrak informasi dan perangkum opini.

2.2 Decision List

Decision List merupakan generalisasi yang ketat dari kelas fungsi *Boolean* dengan menggabungkan keunggulan dari klasifikasi *decision tree, N-gram taggers* dan *Bayesian* [19] [16]. Pada *decision list* diasumsikan terdapat satu *sense* tiap *collocation property* [15]. *N-gram taggers* merupakan urutan yang berdekatan dari n item dari sampel teks atau ucapan yang diberikan [22]. Penggunaan *n-gram* pada *decision list* terletak pada *collocation*-nya. *Collocation* merupakan kata yang terdekat dari kata polisemi yang ditunjuk yang digunakan bersama untuk membentuk satu kesatuan makna. *Collocation* digunakan sebagai petunjuk yang kuat dan konsisten dari target kata yang akan disambiguasikan.

Dalam perhitungannya, bobot *collocation* pada *decision list* dihitung dengan menggunakan distribusi probabilitas sebagai berikut [11].

$$weight(sense_i, feature_k) = \log\left(\frac{Pr(sense_i \mid feature_k)}{\sum_{j \neq i} Pr(sense_j \mid feature_k)}\right) \quad (2\text{-}1)$$

dengan
weight : bobot suatu *feature* pada sense tertentu
$Pr(sense_i \mid feature_k)$: Probabilitas *feature* k ber-*sense* i
$Pr(sense_j \mid feature_k)$: Probabilitas *feature* k ber-*sense* j (selain *sense* i)

Formula 2-1 digunakan untuk menghitung nilai rasio *log-likelihood* dalam *decision list.* Nilai tersebut akan masuk ke dalam tabel *decision list* apabila bernilai positif. Namun, apabila hasil bagi probabilitasnya kurang dari 1 dan menyebabkan nilai logaritmanya menjadi negatif maka nilai rasio *log-likelihood*-nya tidak dimasukkan ke dalam tabel *decision list* [11]. Kemudian apabila nilai penyebut dalam pembagian formula 2-1 bernilai 0, maka akan diganti menjadi 0.0001. Semakin besar nilai rasio *log-likelihood* yang didapatkan maka semakin besar prediksi kemunculannya. *Collocation* akan diurutkan dan disusun dalam tabel *decision list* berdasarkan urutan nilai *log-likelihood* terbesar.

2.3 Information Gain

Pada dasarnya seleksi fitur bertujuan mengurangi fitur-fitur yang tidak relevan sehingga dapat meningkatkan nilai akurasi dari sistem. Namun penggunaan seleksi fitur tidak menutup kemungkinan dapat membuat turun nilai akurasi nantinya mengingat tingginya kompleksitas komputasi terhadap pengenalan pola pada ruang dimensi. *Information gain* merupakan salah satu algoritma *feature selection* yang mengukur berapa banyak informasi yang terdapat maupun yang tidak dari suatu kata yang berperan untuk membuat keputusan klasifikasi. Untuk menghitung *information gain,* dapat menggunakan formula 2-2 dan 2-3 sebagai berikut [18].

$$Info\ (D) = -\sum_{i=1}^{c} p_i log_2\ (p_i) \quad (2\text{-}2)$$

dengan
c : jumlah nilai yang ada pada atribut target (jumlah kelas klasifikasi)
p_i : jumlah sampai untuk kelas i

$$Info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} x\ Info(D_j) \quad (2\text{-}3)$$

dengan
A : atribut
$|\boldsymbol{D_j}|$: jumlah sampel untuk nilai j
$|\boldsymbol{D}|$: jumlah seluruh sampel data
v : suatu nilai yang mungkin untuk atribut A

Kemudian nilai *information gain* yang digunakan untuk mengukur efektifitas suatu atribut dalam pengklafikasian data dapat dihitung dengan formula 2-4 dibawah ini.

$$Gain(A) = |Info\ (D) - Info_A(D)| \quad (2\text{-}4)$$

2.4 Pengukuran Evaluasi

Kualitas hasil klasifikasi dapat diukur menggunakan pengukuran matriks evaluasi. Sesuai kaidah statistika yang digunakan pada penelitan *natural language processing* (NLP) [12]. Pengukuran ini dapat digunakan untuk menghitung nilai akurasi dan presisi berdasarkan representasi nilai *true* dan *false* serta nilai positif dan negatifnya. Dengan tabel matrik *confusion,* nilai representasi tersebut dihubungkan seperti pada Tabel 2.1 berikut.

Tabel 2-1 Confusion matrix

Correctness	**Test Assertion**	
	Positive	Negative
True	TP (True Positive)	TN (True Negative)
False	FP (False Positive)	FN (False Negative)

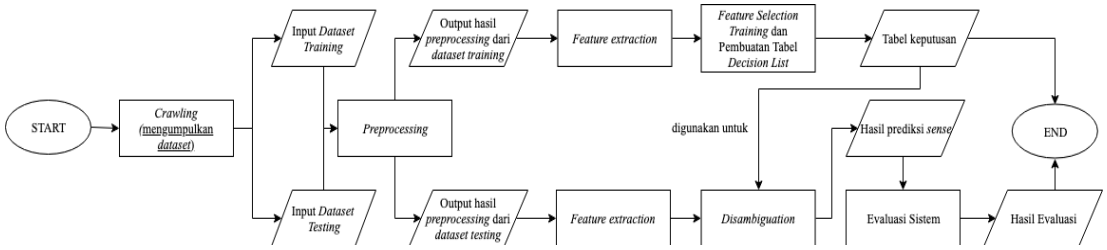
Setelah mendapakan karakteristik sesuai dengan Tabel 2-1, langkah berikutnya adalah menghitung nilai presisi dan dan nilai akurasi dengan menggunakan formula 2-5 dan 2-6 berikut.

$$\text{Presisi} : \frac{TP}{TP+FP} \quad (2\text{-}5) \qquad \text{Akurasi} : \frac{TP+TN}{TP+FP+TN+FN} \quad (2\text{-}6)$$

dengan
Akurasi : tingkat kedekatan pengukuran kuantitas terhadap nilai yang sebenarnya. [20]
Presisi : sejauh mana pengulangan pengukuran dalam kondisi yang tidak berubah mendapatkan hasil yang sama. [21]

3. Sistem yang Dibangun

3.1 Gambaran Arsitektur Sistem



Gambar 3.1 Arsitektur Sistem

Berdasarkan Gambar 3.1, tahap awal pembangunan sistem dimulai dengan melakukan *crawling* untuk mendapatkan *dataset* yang digunakan pada penelitian ini dan membagi menjadi dua macam *dataset training* dan *dataset testing*. Kemudian melakakukan proses *preproccesing* pada *dataset training* dan *dataset testing*. Tahap *preprocessing* merupakan tahap penyeragaman data agar dapat memudahkan pembacaan pada saat proses klasifikasi. Selanjutnya dilakukan proses uji *training* terhadap *dataset training*. Dalam proses uji *training* proses *feature selection* dan klasifikasi *decision list* juga dilakukan. Hasil dari uji *training* berupa tabel keputusan *decision list*, nantinya digunakan untuk menentukan *sense* dari kalimat yang terdapat pada *dataset testing* melalui uji *testing*.

3.2 Dataset

Pada peneletian ini, dataset pengujian didapatkan dari website koran harian kompas (https://www.kompas.com) dan portal berita detik (https://www.detik.com) berupa kumpulan kalimat yang mengandung kata polisemi sebanyak 5 varian, periode berita tahun 2007 – 2017 menggunakan *crawling* dari program yang sudah dibuat oleh peneliti. Kalimat polisemi yang digunakan pada *dataset* adalah yang mengandung kata **daun, buah, kursi, bulan** dan **akar.** Kata-kata tersebut merupakan beberapa kata yang paling sering digunakan, diucapkan sehari-hari baik formal maupun informal. Penelitian ini mengunakan Kamus Besar Bahasa Indonesia (KBBI) sebagai *sense* acuan dari tiap kata polisemi yang digunakan pada dataset dan menggunakan *wordnet* yang telah dibuat oleh Putra et al [13] pada penelitian sebelumnya. *Dataset* yang digunakan pada penelitian ini menggunakanan *JavaScript Object Notation* (JSON) format dengan dua jenis *dataset* yaitu *datatset training* dan *dataset testing*.

Tabel 3-1 Jumlah variasi data kata polisemi beserta makna katanya

Kata Polisemi	Jumlah data	Sense	Diskripsi Sense
Kursi	Data *training* : 154 Data *testing* : 101	Kursi-1	Tempat duduk yang berkaki dan bersandaran.
		Kursi-2	Kedudukan, jabatan (dalam parlemen, kabinet, pengurus, dan sebagainya): ia terpilih menduduki – ketua.
Daun	Data *training* : 123 Data *testing* : 100	Daun-1	Bagian tanaman yang tumbuh berhelai-helai pada ranting (biasanya hijau) sebagai alat bernapas dan mengolah zat makanan.
		Daun-2	Bagian barang yang tipis lebar (seperti -- dayung; -- jendela; -- pintu).
		Daun-3	Memperoleh nasib baik; menanjak; selalu menang atau selalu mendatangkan untung: hasil laut yg kini lagi naik -- sbg komoditas ekspor ialah rumput laut.
Akar	Data *training* : 134 Data *testing* : 100	Akar-1	Memperoleh nasib baik; menanjak; selalu menang atau selalu mendatangkan untung: hasil laut yg kini lagi naik -- sbg komoditas ekspor ialah rumput laut.
		Akar-2	Asal mula; pokok; pangkal; yang menjadi sebab(-sebabnya): yang perlu dibasmi adalah -- segala kejahatan.
		Akar-3	Bagian tubuh makhluk hidup: ~ gigi.
		Akar-4	Suatu operasi aljabar, yang biasanya dinyatakan dengan simbol ?, misalnya (akar a sama dengan b), berarti; b2 = a, jadi (akar pangkat _n dari _a sama dengan _c), berarti cn = a.
		Akar-5	Sebuah panggilan (nama, jenis): tikus --.
Buah	Data *training* : 147 Data *testing* : 99	Buah-1	Bagian tumbuhan yang berasal dari bunga atau putik (biasanya berbiji): pohon mangga itu banyak -- nya.
		Buah-2	Kata penggolong bermacam-macam benda: dua -- kapal; se -- negeri; dua -- rencana.
		Buah-3	Pokok; bahan: -- percakapan.
		Buah-4	Hasil: -- jerih payahnya kini dapat dinikmati oleh keturunannya;-- manis berulat di dalamnya, pb perkataan yang manis-manis biasanya mengandung maksud yang kurang baik; sebab -- dikenal pohonnya, pb dari perbuatan atau perangai seseorang dapat diketahui asalnya.
Buah	Data *training* : 147 Data *testing* : 99	Buah-5	Nama panggilan (nama orang, jalan, tempat): -- batu.
		Buah-6	Anggota suatu kelompok yang tingkatannya berada di bawah pimpinan: anak --.
Bulan	Data *training* : 134 Data *testing* : 100	Bulan-1	benda langit yang mengitari bumi, bersinar pada malam hari karena pantulan sinar matahari: pesawat antariksa Apollo berhasil mendarat di --; bumi bermandikan cahaya --.
		Bulan-2	Masa atau jangka waktu perputaran bulan mengitari bumi dari mulai tampaknya bulan sampai hilang kembali (29 atau 30 hari); masa yang lamanya 1/12 tahun: penataran itu berlangsung selama dua --; istrinya sedang hamil empat --.
		Bulan-3	Kiasan atau pribahasa: -- jatuh dalam ribaan, ki mendapat untung besar; bagai -- kesiangan, pb pucat dan lesu; bagai -- dengan matahari, pb sebanding; sesuai.

Tabel 3-2 Contoh Dataset Training

Kata Polisemi	Sense	Contoh Training
Kursi	Kursi-1	Kursi yang dipakai di kampus ini sudah mulai rusak.
Kursi	Kursi-2	DPD justru ricuh karena berebut kursi pimpinan.
Akar	Akar-1	Aksesori dari akar bahar dijual berdampingan dengan batu bacan maupun batu obi.

Tabel 3-3 Contoh Dataset Testing

Kata Polisemi	Sense Tepat	Contoh Testing	Prediksi
Bulan	Bulan-1	Bulan adalah benda langit yang paling terang setelah Matahari.	…..
Bulan	Bulan-2	Mendagri sangat kecewa dengan kasus OTT di tiga bulan terakhir ini.	…..
Daun	Daun-1	Daun pohon palem, misalnya, berukuran jauh lebih besar dari semanggi.	…..

3.3 Preprocessing

Tahap *preprocessing* bertujuan untuk menyeragamkan data agar dapat memudahkan pembacaan pada saat proses klasifikasi. Tahapan *preprocessing* meliputi:

1. Tahap *case folding,* setiap huruf kapital pada kalimat dalam *dataset training* dan *dataset testing* dirubah menjadi huruf kecil.
2. Tahap *tokenization,* kalimat dalam *dataset* dipecah perkata menjadi potongan-potongan kata atau *token*, tanpa tanda baca sesuai kebutuhan sistem.
3. Proses *stop word removal,* setiap kata diindetifikasi untuk digunakan atau dibuang bila sesuai dengan *stoplist. Stoplist* merupakan kata yang tidak deskriptif yang dapat dibuang dalam pendekatan *bag-of-words* [19]. *List stopword* yang dipakai pada penilitian ini adalah semua kata yang termasuk jenis kata penghubung bahasa Indonesia.
4. Tahapan *Stemming*, setiap kata yang mempunyai imbuhan, kata bentuk dua, kata bentuk tiga dirubah menjadi bentuk kata dasarnya.
5. Tahapan terakhir pada proses *preproccesing* adalah proses pembuatan *sense tags* untuk menampung informasi makna kata dari sebuah kata ambigu dalam kalimat di *dataset*. Tahap ini merupakan tahap penting dalam WSD dimana *sense tags* yang terdapat pada kalimat dalam *dataset training* digunakan untuk membuat tabel keputusan dan *sense tags* yang terdapat pada kalimat dalam *dataset testing* digunakan sebagai pembanding dengan hasil prediksi *sense* oleh sistem untuk dijadikan bahan evaluasi.

Tabel 3-4 Contoh Tahapan dalam Preprocessing

Kalimat Awal	Dalam perkembangannya, daun mint memiliki sejumlah khasiat baik untuk kulit.
Tahap *Case Folding*	dalam perkembangannya, daun mint memiliki sejumlah khasiat baik untuk kulit.
Tahap *Tokenization*	| dalam | perkembangannya | daun | mint | memiliki | sejumlah | khasiat | baik | untuk | kulit |
Tahap *Stop Word Removal*	| dalam | perkembangannya | daun | mint | memiliki | sejumlah | khasiat | kulit |
Tahap *Stemming*	| dalam | kembang | daun | mint | milik | jumlah | khasiat | kulit |
Tahap *Sense Tags*	| dalam | kembang | daun | mint | milik | jumlah | khasiat | kulit | : Daun-1

3.4 Feature Extraction

Setelah melewati proses *preprocessing*, tahap selanjutnya adalah *feature extraction.* Pada tahap ini, akan dilakukan ekstraksi fitur dari hasil *preprocessing dataset training* maupun *dataset testing.* Ekstraksi fitur dilakukan dengan cara menentukan *feature vector* yang terdiri dari *collocation feature* dan *co-occurrence feature*. *Collocation feature* untuk menandai kata dalam rentang tertentu (biasanya ± 2) dari kata ambigu yang ditunjuk. Sedangkan *co-occurrence feature* untuk menampilkan jumlah kemunculan kata yang ditandai sebagai *collocation* dalam dokumen *dataset training* dan *dataset testing.* Hasil *feature extraction dataset training* digunakan untuk proses pembelajaran *feature selection* dan pembuatan tabel *decision list.* Sedangkan hasil *feature extraction dataset testing* digunakan untuk penentuan *sense* pada proses *disambiguation.*

Berikut contoh *feature vector* dari hasil *preprocessing* :

| dalam | kembang | daun | mint | milik | jumlah | khasiat | kulit |

Collocation feature range 2 : [dalam, kembang, mint, milik]

Co-occurrence feature : [5, 3, 1, 4]

3.5 Feature Selection Training dan Pembuatan Tabel Decision List

Tahap ini merupakan tahap pembelajaran sistem yang nantinya hasil pembelajaran tersebut digunakan juga pada proses *disambiguation. Feature vector* yang didapatkan dari ekstraksi fitur hasil *preprocessing dataset training* dilakukan *feature selection training* terlebih dahulu. *Feature selection* ini menggunakan metode algoritma *information gain* terhadap *feature vector* untuk mendapatkan nilai *gain* dari setiap *feature*. Nilai *gain* tersebut digunakan untuk menentukan *threshold* gain sebagai batas seleksi fitur. Apabila nilai *gain* dari suatu *feature* kurang dari *threshold gain*. Maka *feature* tersebut tidak digunakan pada proses klasifikasi sehingga pengurangan atribut sebagai tujuan *feature selection* terpenuhi. Selanjutnya, *feature* yang berhasil terseleksi untuk digunakan pada proses klasifikasi dilakukan pembobotan oleh algoritma *decision list* untuk mendapatkan bobot *log-likelihood*nya dan dikumpulkan dalam tabel keputusan *decision list* dengan urutan nilai terbesar ke terkecil beserta *sense tag*-nya. Berikut ilustrasi dari tabel keputusan yang dicontohkan pada Tabel 3-5 berikut.

Tabel 3.5 Contoh Tabel Keputusan Hasil Sistem WSD

Fitur	Log-likelihood	Sense
Mint	10.55	Daun-1
Pintu	10.40	Daun-1
Kembang	4.22	Daun-2
Naik	2.33	Daun-3

3.6 Disambiguation

Proses disambiguation adalah proses penentuan sense suatu kalimat dari *dataset testing.* Proses tersebut dilakukan dengan cara mencocokan *collocation feature* yang berasal dari *feature extraction dataset testing* yang sesuai atau sama dengan tabel keputusan. Kemudian membandingkan nilai *log-likehood*-nya dengan fitur lain dan memilih fitur yang memiliki nilai *log-likelihood* terbesar pada tabel keputusan untuk dijadikan acuan predikasi *sense* berdasarkan *sense tags* dari fitur yang berasal dari data *training.*

Berikut contoh hasil proses *disambiguation* :

Kalimat : Dalam perkembangannya, daun mint memiliki sejumlah khasiat baik untuk kulit.
Hasil *preproccesing* : | dalam | kembang | daun | mint | milik | jumlah | khasiat | kulit | : Daun-1
Collocation feature range 2 : [dalam, kembang, mint, milik]

Berdasarkan pencocokan fitur *collocation* dengan tabel keputusan (Tabel 3.5), didapatkan fitur "mint" sebagai fitur yang memiliki bobot terbesar dibanding fitur "kembang". Sehingga, *sense* yang melekat pada fitur "mint" yaitu *sense* Daun-1 pada tabel keputusan, menjadi *sense* acuan prediksi *sense* kalimat pada contoh diatas.

3.7 Evaluasi Sistem

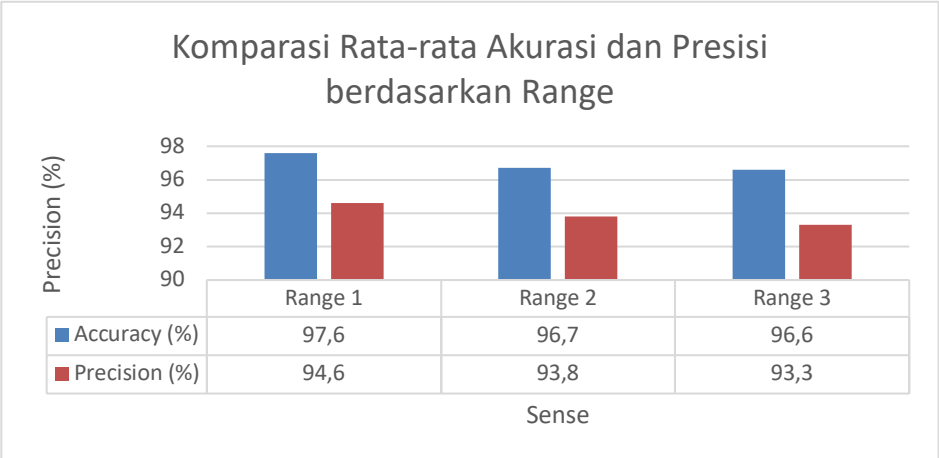
Pada tahap ini, hasil prediksi *sense* yang didapatkan dari proses *disambiguation* dibandingkan dengan *sense tags* dari *dataset testing* kemudian dievaluasi dan dianalisis akurasi beserta presisinya mengunakan formula 2-5 dan 2-6. Apabila hasil analisis akurasi 100%, maka pengujian tersebut memiliki nilai yang sama dengan nilai data yang diberikan. Penggunaan metode evaluasi ini diharapkan mampu mengukur kinerja penelitian terutama penggunaan *information gain* sebagai *feature selection* pada klasifikasi *decision list* sehingga menjadi tolak ukur pengembangan penelitian yang lebih baik kedepannya.

4. Evaluasi

Tujuan pengujian yang dilakukan pada penelitian ini adalah mengetahui parameter *range* yang optimal dalam sistem yang dibangun dan pengaruh *feature selection* terhadap akurasi dan presisi klasifikasi *Decision List.* Berikut skenario pengujian dan analisis hasil pengujian yang dilakukan:

4.1 Skenario-1 dan Analisis Hasil Pengujian

Pada skenario 1, dilakukan pengujian tingkat akurasi dan presisi dengan menggunakan *range collocation* yang berbeda (1, 2, 3) pada tiap kelas kata ambigu. Hasil pengujian pada bagian ini akan ditampilkan berupa tabel *range* fitur (*collocation*), tabel akurasi serta tabel presisi hasil klasifikasi menggunakan *decision list* tanpa seleksi fitur, dapat dilihat pada Tabel 1 - 5 lampiran.



Gambar 4.1 Komparasi Rata-rata Akurasi dan Presisi Seluruh Sense berdasarkan Range

Berdasarkan Gambar 4.1 mengenai hasil pengujian skenario 1, bahwa semakin kecil *range collocation* dapat menghasilkan akurasi dan presisi yang semakin besar. Hal ini disebabkan karena *range* yang kecil / *range* 1

cenderung menghasilkan *collocation* ber-*sense* tunggal karena fitur yang disaring lebih sedikit. *Collocation* ber-*sense* tunggal merupakan *collocation* yang kemunculannya hanya berada pada satu sense tertentu dan menghasilkan nilai *log-likelihood* yang besar. Sehingga disaat proses klasifikasi, *classifier* cenderung lebih memilih *collocation* tersebut sebagai acuan prediksi *sense* suatu kalimat karena bobot fiturnya yang besar.

Berikut contoh perhitungan nilai *log-likelihood* sebagai bobot penentuan *sense* pada *collocation* ber-*sense* tunggal dan yang ber-*sense multiple* :

Kalimat: "Museum Victoria di Melbourne yang terlibat dalam penemuan tikus akar tersebut mengatakan sangat bangga bahwa penemuan mereka masuk ke dalam daftar 10 besar spesies tersebut."
Sense: Akar-5
Range: 1
Collocation: (tikus, sebut)
Co-occurrence "tikus*"*:

- Muncul pada : Akar-5 dengan jumlah = 7

Total fitur pada *sense* Akar-5: 42

*Kata "tikus" adalah *collocation* ber-*sense* tunggal karena hanya muncul pada *sense* Akar-5.

Co-occurrence "*sebut"*:

- Muncul pada: Akar-2 dengan jumlah = 1

Total fitur pada sense Akar-2: 58

- Muncul pada: Akar-5 dengan jumlah = 1

Total fitur pada sense Akar-5: 42

*Kata "sebut" adalah *collocation* yang ber-*sense multiple* karena kemunculannya lebih dari satu *sense* yakni pada *sense* Akar-2 dan Akar-5.

$$\begin{aligned} Weight\ (Akar_5, tikus) &= \log\left(\frac{Pr(Akar_5, tikus)}{\sum_{j\neq 5}(Akar_j, tikus)}\right) \\ &= \log\left(\frac{7/42}{0}\right) \\ &= \log\left(\frac{7/42}{0.0001}\right) \\ &= 3.2218487496163557 \end{aligned} \quad (4\text{-}1)$$

$$\begin{aligned} Weight\ (Akar_5, sebut) &= \log\left(\frac{Pr(Akar_5, sebut)}{\sum_{j\neq 5}(Akar_j, sebut)}\right) \\ &= \log\left(\frac{1/42}{1/58}\right) \\ &= 0.1401787031650368 \end{aligned} \quad (4\text{-}2)$$

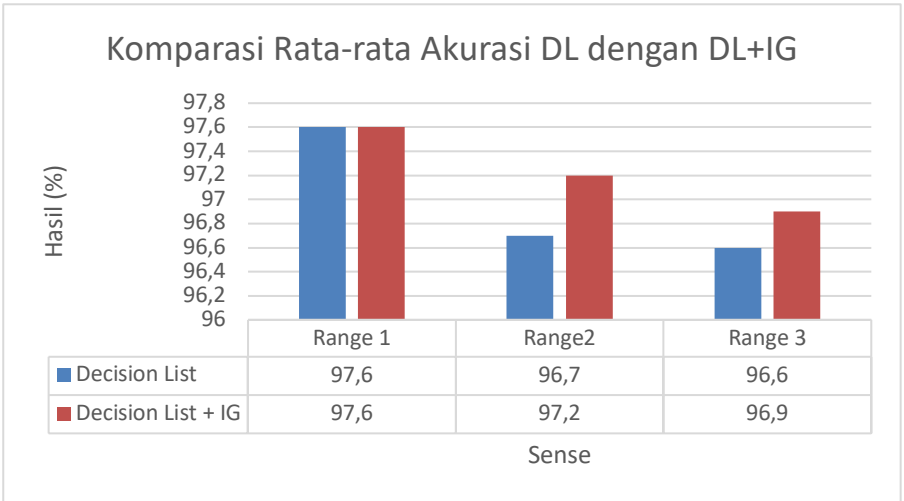
$$\begin{aligned} Weight\ (Akar_2, sebut) &= \log\left(\frac{Pr(Akar_2, sebut)}{\sum_{j\neq 5}(Akar_j, sebut)}\right) \\ &= \log\left(\frac{1/58}{1/42}\right) \\ &= -0.1401787031650368 \end{aligned} \quad (4\text{-}3)$$

Berdasarkan hasil perhitungan 4-1, 4-2 dan 4-3 didapatkan perhitungan 4-1 yang mempresentasikan bobot fitur "tikus" (*collocation* ber-*sense* tunggal) menghasilkan bobot yang lebih besar dibanding fitur "sebut" pada perhitungan 4-2 dan 4-3. Maka dari itu, fitur "tikus" ditentukan sebagai fitur ber-*sense* Akar-5 dan disimpan di tabel keputusan. Sedangkan untuk fitur "sebut" dapat ditentukan sensenya dengan membandingkan hasil perhitungan 4-2 dan 4-3. Karena hasil perhitungan 4-2 (fitur "sebut" *sense* Akar-5) lebih besar dari pada perhitungan 4-3 (fitur "sebut" *sense* Akar-2), maka fitur "sebut" ditentukan sebagai fitur ber-*sense* Akar 5 dan disimpan di tabel keputusan. Tabel keputusan yang berisi berbagai macam fitur, bobot dan *sense*-nya, saat proses klasifikasi akan dipilih fitur yang memiliki bobot terbesar sebagi penentu *sense* dari suatu kalimat pada saat proses *disambiguation*.

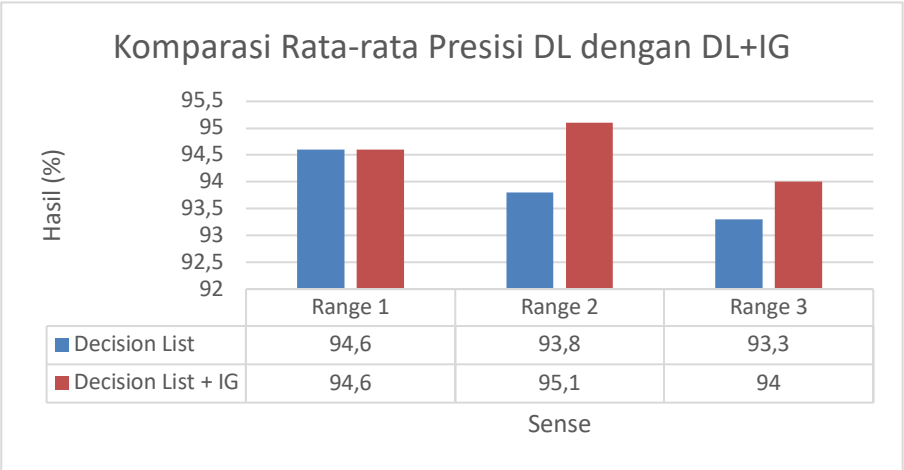
4.2 Skenario-2 dan Analisis Hasil Pengujian

Pada skenario 2 dilakukan pengujian tingkat akurasi dan presisi dengan menggunakan *range collocation* yang berbeda (1, 2, 3) dengan *threshold gain* pada tiap kelas kata ambigu yang menggunakan seleksi fitur *information gain.* Hasil pengujian pada bagian ini akan ditampilkan berupa tabel *range* fitur (*collocation*), tabel batas ambang (*threshold*) *gain*, tabel akurasi serta tabel presisi hasil klasifikasi menggunakan *decision list* dengan seleksi fitur, dapat dilihat pada Tabel 6.1 – 20.1 lampiran.

Parameter *threshold* lebih dari 0 dapat diartikan bahwa tidak ada fitur yang diseleksi untuk dihapus atau semua fitur tetap digunakan sehingga nilai akurasi dan presisi sama dengan hasil pengujian skenario 1. Parameter *threshold* lebih dari optimal *gain*, merupakan batas *gain* optimal yang mampu menghasilkan akurasi dan presisi terbaik dari batas *gain* lainnya. Parameter tersebut didapat dari pengukuran beberapa *threshold gain* yang mewakili *gain* keseluruhan untuk mencari *gain* optimal. Pada penelitian ini, syarat dari penentuan *gain* optimal yaitu *gain* yang menjadi *threshold* adalah *gain* yang dapat memberikan akurasi dan presisi terbaik dari proses klasifikasi. Hasil pengujian *threshold* tiap kelas *sense* dapat dilihat pada Tabel 6 - 20 lampiran.



Gambar 4.2 Komparasi Rata-rata akurasi decision list dengan decision list + inforamtion gain



Gambar 4.3 Komparasi Rata-rata presisi decision list dengan decision list + information gain

Berdasarkan Gambar 4.2 dan 4.3 mengenai hasil skenario pengujian 2, terjadi kenaikan akurasi dan presisi dengan klasifikasi *decision list* menggunakan *information gain* pada *range* 2 dan 3 dibanding klasifikasi *decision list* tanpa *information gain*. Hal tersebut sesuai dengan justifikasi pada latar belakang penelitian. Sebab penggunaan seleksi fitur yang tepat akan mereduksi atribut atau fitur yang tidak relevan sehingga kinerja *classifier* akan lebih optimal dan dapat meningkatkan hasil akurasi klasifikasi. Penggunaan seleksi fitur *information gain* pada penelitian ini terutama, mengalami peningkatan akurasi dan presisi klasifikasi apabila dapat menentukan *threshold gain* optimal sehingga dapat mempengaruhi hasil klasifikasi yang lebih baik lagi.

Namun, pada *range* 1 berbeda hasil pengujian dibanding pengujian *range* 2 dan 3. Pada pengujian *range* 1 menggunakan *information gain* pada *decision list* justru tidak mempengaruhi perbedaan hasil akurasi dan presisinya. Hal ini dikarenakan *collocation* yang sedikit pada *range* 1, cenderung memiliki *sense* tunggal yang apabila *collocation* tersebut tereduksi oleh *feature selection* dapat mengakibatkan kesalahan klasifikasi sehingga kegagalan prediksi (*unprediction*) *sense* meningkat. Namun karena penentuan *threshold gain* yang optimal, pereduksian ini dapat dihindari. Sehingga *collocation* fitur pada *range* 1 tidak tereduksi dan hasil akurasi dan presisipun tidak berubah sama seperti pengujian pada skenario-1

Dari hasil pengujian *range* 1 baik menggunakan *feature selection* atau tidak, menghasilkan pengujian yang lebih reliabel dibanding pengujian *range* 2 dan 3. Namun penggunaan *range* 1 *collocation* membutuhkan banyaknya teks (*corpus*) dalam *dataset training*. Hal tersebut dikarenakan pada *range* 1, fitur *training* yang

disaring lebih sedikit. Sehingga diperlukan *corpus* yang memadahi agar kemungkinan kegagalan prediksi atau kesalahan prediksi semakin berkurang. Kesalahan atau kegagalan prediksi karena kurangnya ketersediaan *corpus* dibuktikan pada Tabel 2 Lampiran mengenai pengujian *range* 1 kata "daun". Pada pengujian tersebut akurasi dan presisi hasil pengujian *range* 1 kurang baik apabila dibandingkan dengan hasil pengujian *range* 3. Pada pengujian *range* 1 kata "daun", terdapat fitur yang tidak terprediksi karena fitur tersebut tidak ada dalam tabel keputusan. Sehingga, pemilihan fitur lain yang memiliki *sense* yang tidak sesuai dengan *sense* asli kata polisemi targetpun dijadikan acuan untuk memprediksi *sense.* Akibatnya, terjadi kesalahan prediksi yang menyebabkan penurunan akurasi dan presisi. Selain memperbanyak ketersediaan *corpus* dalam *dataset training,* kesalahan atau kegagalan prediksi dapat ditanggulangi dengan memperbesar *range collocation*. Namun dengan perlakuan tersebut belum pasti akan lebih baik perubahan akurasi dan presisinya bila dibandingkan dengan perubahan akurasi dan presisi pada *range* 1 dengan kondisi ketersedian *corpus* yang memadahi seperti hasil pengujian range 1 lainnya.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan analisis hasil pengujian pada penelitian ini maka dapat disimpulkan sebagai berikut:

1. Pada pengujian dengan *range* yang berbeda-beda dalam klasifikasi *decision list* baik menggunakan *information gain* dan tidak, dapat disimpulkan bahwa semakin kecil *range* cenderung semakin lebih dipercaya (reliabel) dalam mendisambigusikan kata polisemi.
2. *Information gain* dapat meningkatkan akurasi *decision list* dengan selisih 0.5% pada pengujian *range* 2 dan kenaikan akurasi dengan selisih 0.3% pada pengujian *range* 3.
3. *Information gain* dapat meningkatkan akurasi *decision list* sebesar 0.5% pada pengujian *range* 2 dan kenaikan akurasi sebesar 0.3% pada pengujian *range* 3.
4. Penggunaan *range* 1 yang reliabel perlu ditunjang dengan ketersediaan *corpus* dalam *dataset training* agar kemungkinan terjadi kegagalan atau kesalahan prediksi dapat berkurang.

5.2 Saran

Berikut saran dari peneliti untuk penelitian selanjutnya terkait penelitian yang sudah dilakukan:

1. *Dataset training* perlu di perbanyak lagi agar kemungkinan peningkatan akurasi dan presisi pada pengujian *range* 1 dapat terjadi apabila menggunakan *feature selection* terutama *information gain.*

Daftar Pustaka

[1] Wang, S., Li, D., Song, X., Wei, Y., & Li, H. (2011). A feature selection method based on improved fisher's discriminant ratio for text sentiment classification. *Expert Systems with Applications*, *38*(7), 8696-8702.

[2] Koncz, P., & Paralic, J. (2011, June). An approach to feature selection for sentiment analysis. In Intelligent Engineering Systems (INES), 2011 15th IEEE International Conference on (pp. 357-362). IEEE.

[3] Forman, G. (2003). An extensive empirical study of feature selection metrics for text classification. *Journal of machine learning research*, *3*(Mar), 1289-1305.

[4] Yang, Y., & Pedersen, J. O. (1997, July). A comparative study on feature selection in text categorization. In *Icml* (Vol. 97, pp. 412-420).

[5] Tan, S., & Zhang, J. (2008). An empirical study of sentiment analysis for chinese documents. *Expert Systems with applications*, *34*(4), 2622-2629.

[6] Ide, N., & Véronis, J. (1998). Introduction to the special issue on word sense disambiguation: the state of the art. *Computational linguistics*, *24*(1), 2-40.

[7] Wang, S., Li, D., Zhao, L., & Zhang, J. (2013). Sample cutting method for imbalanced text sentiment classification based on BRC. *Knowledge-Based Systems*, *37*, 451-461.

[8] Navigli, R. (2009). Word sense disambiguation: A survey. *ACM computing surveys (CSUR)*, *41*(2), 10.

[9] Pal, A. R., & Saha, D. (2015). Word sense disambiguation: A survey. *arXiv preprint arXiv:1508.01346.*

[10] Agirre, E., & Edmonds, P. (Eds.). (2007). *Word sense disambiguation: Algorithms and applications* (Vol. 33). Springer Science & Business Media.

[11] Agirre, E., & Martinez, D. (2000, August). Exploring automatic word sense disambiguation with decision lists and the Web. In *Proceedings of the COLING-2000 Workshop on Semantic Annotation and Intelligent Content* (pp. 11-19). Association for Computational Linguistics.

[12] Mike. Nlp and knowledge statistics. http://wiki.opensemanticframework.org/index.php/NLP_and_Knowledge_Statistics, 2015. [Online; accessed 9-September-2017].

[13] Putra, D. D., Arfan, A., & Manurung, R. (2008, June). Building an Indonesian wordnet. In *Proceedings of the 2nd International MALINDO Workshop* (pp. 12-13).
[14] Indonesia, T. R. K. B. (2008). Kamus Bahasa Indonesia. *Jakarta: Pusat Bahasa Departemen Pendidikan Nasional*.
[15] Yarowsky, D. (1993, March). One sense per collocation. In *Proceedings of the workshop on Human Language Technology*(pp. 266-271). Association for Computational Linguistics.
[16] Yarowsky, D. (1997). Homograph disambiguation in text-to-speech synthesis. In *Progress in speech synthesis* (pp. 157-172). Springer, New York, NY.
[17] Yarowsky, D. (2000). Hierarchical decision lists for word sense disambiguation. *Computers and the Humanities*, *34*(1-2), 179-186.
[18] Han, J., Pei, J., & Kamber, M. (2011). *Data mining: concepts and techniques*. Elsevier.
[19] Rivest, R. L. (1987). Learning decision lists. *Machine learning*, *2*(3), 229-246.
[20] JCGM, J. (2012). *200: 2012—International Vocabulary of Metrology—Basic and General Concepts and Associated Terms (VIM)*. Technical Report.
[21] Taylor, J. (1997). *Introduction to error analysis, the study of uncertainties in physical measurements*.
[22] Broder, A. Z., Glassman, S. C., Manasse, M. S., & Zweig, G. (1997). Syntactic clustering of the web. *Computer networks and ISDN systems*, *29*(8-13), 1157-1166.

Lampiran

Lampiran dapat berupa detil data dan contoh lebih lengkapnya, data-data pendukung, detail hasil pengujian, analisis hasil pengujian, detail hasil survey, surat pernyataan dari tempat studi kasus, screenshot tampilan sistem, hasil kuesioner dan lain-lain.