KLASIFIKASI ARGUMEN SEMANTIK MENGGUNAKAN KOMBINASI FITUR *NAMED ENTITY IN CONSTITUENT*, *HEAD WORD POS*, DAN *SYNTACTIC FRAME*

SEMANTICS ARGUMENT CLASSIFICATION USING *NAMED ENTITIES IN CONSTITUENT*, *HEAD WORD POS*, AND *SYNTACTIC FRAME* FEATURES COMBINATION

Nisaa' 'Ainulfithri [1], Moch. Arif Bijaksana, Ph.D. [2], Siti Saadah, S.T, M.T. [3]

[1,2,3]Prodi S1 Teknik Informatika, Fakultas Teknik, Universitas Telkom

[1]nisaainul@gmail.com, [2]arifbijaksana@gmail.com, [3]sitisaadah@telkomuniversity.ac.id

Abstrak

***Natural Language Processing* (NLP) merupakan salah satu cabang ilmu komputer yang berfokus pada pengolahan bahasa natural/bahasa manusia. Sebagian besar *task* NLP seperti *Question Answering, Semantic Role Labeling,* dan *Information Extraction* memerlukan informasi 5W (*Who, What, Where, When, Why*) dan 1H (*How*) untuk mengekstrak informasi yang dibutuhkan. Klasifikasi argumen semantik merupakan proses pelabelan argumen berdasarkan aturan semantik dimana aturan semantik dapat merepresentasikan informasi 5W+1H tersebut.**

Dalam melakukan klasifikasi argumen semantik diperlukan fitur-fitur yang dapat membantu proses klasifikasi. Pada penelitian ini, fitur yang akan digunakan adalah fitur dasar dan tiga fitur tambahan yaitu *Named Entities in Constituent, Head Word POS*, dan *Syntactic Frame*. Penggunaan ketiga fitur tambahan tersebut terbukti dapat meningkatkan akurasi. Algoritma yang digunakan dalam proses klasifikasi adalah *Sequential Minimum Optimization* (SMO) yang merupakan pengembangan dari *Support Vector Machine* (SVM). Algoritma SMO dapat mengatasi permasalahan *multi-class* dan dapat melakukan proses *learning* dengan waktu yang lebih singkat daripada SVM.

Berdasarkan pengujian yang telah dilakukan, penggunaan tiga fitur tambahan yaitu *Named Entities in Constituent, Head Word POS*, dan *Syntactic Frame* dapat meningkatkan hasil akurasi dengan kenaikan akurasi sebesar 11,82%.

Kata kunci : klasifikasi argumen semantik, Sequential Minimum Optimization (SMO), Natural Language Processing (NLP), *Named Entities in Constituent, Head Word POS, Syntactic Frame*

Abstract

Natural Language Processing (NLP) is a chapter of computer science which is focused on natural language/human language processing. Mainly task in NLP like Question Answering, Semantic Role Labeling, and Information Extraction require 5W (Who, What, Where, When, Why) and 1H (How) for extracting needed information. Semantic argument classification is arguments labeling process based on semantic role.

Some features are needed in semantic argument classification process. In this research, base feature and 3 additional features i.e. Named Entities in Constituent, Head Word POS, and Syntactic Frame are used. Those features have been proven to improve system accuration, For the classification process, Sequential Minimum Optimization (SMO) algoritm is used, SMO is development algoritm from Support Vector Machine (SVM). SMO algoritm can handle multi-class problem and be able to do learning process faster than SVM.

Based on testing that was done, the use of three additional features Named Entities in Constituent, Head Word POS, and Syntactic Frame can improve accuracy results with an increase of 11.82%.

Keywords : semantic argument classification, Sequential Minimum Optimization (SMO), Natural Language Processing (NLP), Named Entities in Constituent, Head Word POS, Syntactic Frame

1. Pendahuluan

Beberapa task NLP seperti *Semantics Role Labeling*, *Information Extraction*, dan *Question Answwering* memerlukan pelabelan 5W + 1H untuk untuk mendapatkan informasi. Sehingga diperlukan sebuah sistem yang dapat membantu proses pelabelan.

Klasifikasi argumen semantik merupakan proses pemberian label berdasarkan aturan semantik. Setiap argumen dari sebuah predikat akan diberi label dari ARG0 hingga ARG5 berdasarkan peran semantiknya.

Pada jurnal ini dibahas penggunaan tiga fitur tambahan yaitu *Named Entities in Constituent*, *Head Word POS*, dan *Syntactic Frame* yang nantinya akan dikombinasikan dengan fitur dasar. Penggunaan fitur *Named Entities in Constituent* dan *Head Word POS* terbukti dapat meningkatkan akurasi pada penelitian yang dilakukan oleh *Daniel Jurafsky* pada tahun 2005 [1]. Selain itu fitur *Syntactic Frame* merupakan fitur baru yang digagas oleh *Martha Palmer* dan menghasilkan akurasi tertinggi [2]. Proses klasifikasi dibantu dengan algoritma *Support Vector Machine* (SVM) yang telah dioptimalkan menggunakan *Sequential Minimal Optimization* (SMO) [3].

2. Dasar Teori

2.1. Natural Language Processing

Natural Language Processing (NLP) adalah salah satu bidang ilmu *Artificial Intelligence* (Kecerdasan Buatan) yang mempelajari komunikasi antara manusia dengan komputer melalui bahasa alami [4]. Pada prinsipnya bahasa alami adalah suatu bentuk representasi suatu pesan yang ingin dikomunikasikan antar manusia yang berupa ucapan/suara tetapi sering pula dinyatakan dalam bentuk tulisan.

2.2. Klasifikasi Argumen Semantik

Menurut Kamus Besar Bahasa Indonesia, semantik adalah 1) ilmu tentang makna kata dan kalimat; pengetahuan mengenai seluk-beluk dan pergeseran arti kata; 2) bagian struktur bahasa yang berhubungan dengan makna ungkapan atau struktur makna suatu wicara [5]. Sedangkan argumen adalah nomina atau frasa nominal yang bersama-sama predikat membentuk preposisi [5].

Klasifikasi argumen semantik merupakan proses ekstraksi struktur semantik dari sebuah kalimat dengan melakukan identifikasi kejadian (predikat) serta bagian dari kejadian tersebut (argumen) [6] lalu memberikan label kelas ARG0 hingga ARG4 maupun ARGM. Proses klasifikasi argumen semantik merupakan lanjutan dari identifikasi argumen yang merupakan *task* dari *semantics role labeling* [7].

2.3. Fitur yang Digunakan

2.3.1. Fitur Dasar

1. *Predicate*: merupakan fitur yang menjelaskan tentang predikat pada sebuah kalimat.
2. *Phrase Type*: kategori sintaksis berdasarkan argumen semantik. Contoh: (NP, PP, S, VP, dll).
3. *Position*: Posisi komponen yang akan diklasifikasikan dengan memperhatikan predikat (sebelum atau sesudah predikat).
4. *Voice*: merupakan fitur yang menjelaskan apakah kalimat tersebut pasif atau aktif.
5. *Head Word*: kata kunci dari sebuah frasa. Jika diberikan sebuah frasa *a young lady*. Maka *head word* dari frasa tersebut adalah *lady*.
6. *Sub-categorization*: merupakan aturan stuktur frasa yang merupakan turunan dari *parent* predikat.

2.3.2. Fitur Tambahan

1. *Named Entities in Constituent*
   Named Entities in Constituent (NEC) merupakan fitur yang merepresentasikan 7 entitas (PERSON, ORGANIZATION, LOCATION, PERCENT, MONEY, TIME, DATE) dan menambahkan entitas tersebut sebagai fitur biner. Dengan penambahan fitur NEC, maka klasifikasi argumen khususnya ARGM LOC dan ARGM TMP akan lebih mudah.
2. *Head Word Part-Of-Speech (POS)*
   Head Word POS merupakan fitur yang menyempurnakan fitur head word. Fitur ini akan mengidentifikasi *part-of-speech* sebuah *head word* dari sebuah *constituent*. Jika sebuah head word telah diketahui kategori sintaksisnya maka akan mempermudah proses klasifikasi argumen semantik.
3. *Syntactic Frame*
   Fitur ini mendeskripsikan pola sekuensial pada *noun phrase* dan *predicate* pada sebuah kalimat. Dalam mengaplikasikan fitur *sub-categorization*, diperlukan pelengkap yaitu fitur *syntactic frame*. Fitur ini menandai predikat dan NP sebagai 'pivot' atau poros kemudian komponen lain akan berelasi pada poros tersebut.

2.4. Sequential Minimum Optimization (SMO)

Kelebihan SVM dalam menyalesaikan masalah dalam *pattern recognition*, serta *text categorization* [8] membuat SVM tidak diikuti dengan keefektifan. Sequential Minimum

Optimization (SMO) adalah algoritma yang dapat mengatasi permasalahan *quadratic programming* (QP) [9]. SMO digunakan pada proses training SVM yang dapat menghasilkan solusi atas permasalahan optimisasi dan dapat melakukan proses training dengan waktu yang lebih singkat. Singkatnya waktu training dikarenakan SMO tidak membutuhkan penyimpanan matriks, sehingga data yang berkuran besar dapat diatasi oleh SMO.

SMO akan menyelesaikan permasalahan optimasi seminimal mungkin pada setiap tahapan. Setiap tahap, SMO akan memilih dua komponen *langrange multipliers* untuk dioptimasi secara bersamaan dan memperbaharui SVM dengan hasil optimasi yang baru.

2.5. Pengukuran Evaluasi

Untuk mengetahui performansi hasil klasifikasi, diperlukan sebuah teknik untuk pengukuran evaluasi. Pengukuran dilakukan menggunakan *confusion matrix*. *Confusion Matrix* merupakan suatu metode penghitungan yang terdiri dari informasi mengenai hasil klasifikasi yang dilakukan oleh sistem baik yang benar maupun yang salah. *Confusion matrix* dapat dilihat pada Tabel 1 di bawah :

Tabel 1 *Confusion Matrix*

		Prediksi	
		Positif	Negatif
Aktual	Positif	TP	FN
	Negatif	FP	TN

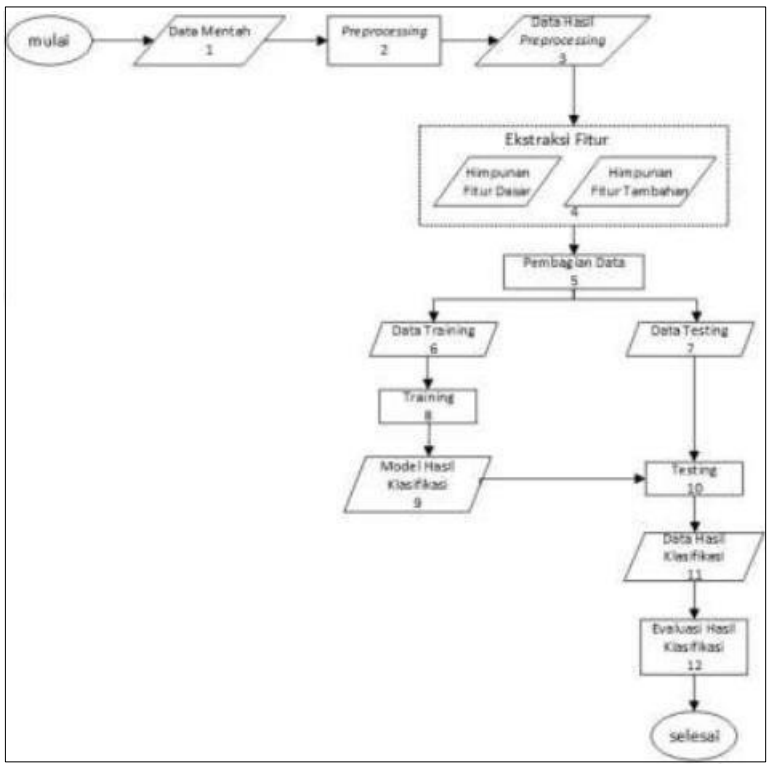
Baris prediksi merupakan baris hasil klasifikasi yang dilakukan oleh sistem, sedangkan kolom aktual merupakan hsail prediksi yang sebenarnya dan dilakukan secara manual. *True Positive (*TP) adalah kelas yang diprediksi positif dan benar, *True Negative* (TN) adalah kelas yang diprediksi negatif dan benar, *False Positif* (FP) adalah kelas yang diprediksi positif dan salah, sedangkan *False Negative* (FN) adalah kelas yang diprediksi negatif dan salah.

Berdasarkan Tabel 1, hasil akurasi dapat dihitung dengan rumus :

(2.1)

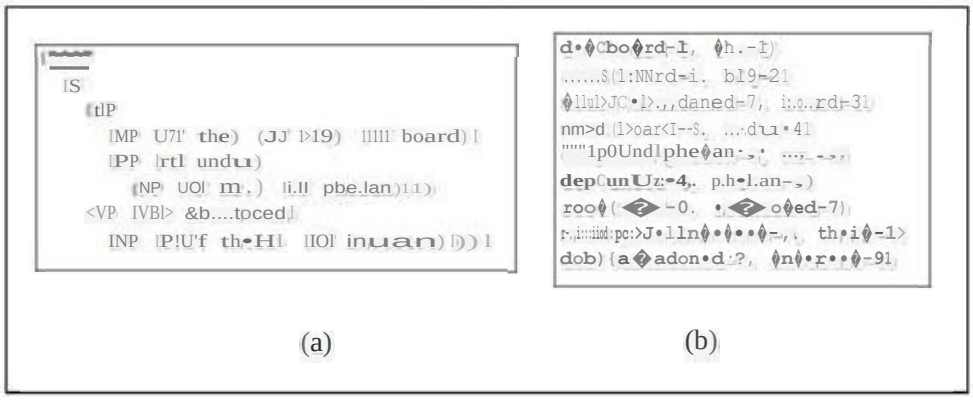
3. Hasil Perancangan

Dalam melakukan klasifikasi argumen semantik, berikut diagram alur yang menggambarkan tahapan yang harus dilakukan :



Gambar 3-0-1 : Gambaran umum sistem

Input dari sistem ini adalah kalimat berbahasa Inggirs yang berasal dari PropBank. Selanjutnya kalimat tersebut akan masuk kedalam proses pembersihan data atau yang biasa disebut *preprocessing*. Metode *preprocessing* yang digunakan adalah *case folding* dan pembentukan pohon sintaksis. Pembentukan pohon sintaksis dilakuakan setelah kalimat baru terbentuk. Pembentukan kalimat dilakukan karena penelitian ini hanya berfokus pada klasifikasi argumen, sehingga hanya argumen dan predikat yang akan diambil dari kalimat tersebut.



Gambar 2 Contoh pohon semantik (a) dan daftar dependensi (b) Stanford Parser

Proses selanjutnya adalah ekstraksi fitur. Fitur dasar dan fitur tambahan akan diekstraksi berdasarkan pohon sintaksis yang telah terbentuk pada Gambar 2 (a). Hasil ekstraksi fitur akan menghasilkan tabel seperti Tabel 2 berikut :

Tabel 2 : Hasil ekstraksi fitur

Pre	Vo	Sc	Pt	Hw	Po	Pers	Loc	Org	Money	Time	Date	Perc	Hwpos	Sf
abandoned	active	VP-VBD-NP	NP	board	L	0	0	1	0	0	0	0	NN	NP-V-NP
abandoned	active	VP-VBD-NP	PP	phelan	L	1	0	0	0	0	0	0	NN	null
abandoned	active	VP-VBD-NP	PP	interest	R	0	0	0	0	0	0	0	NN	NP-V-NP

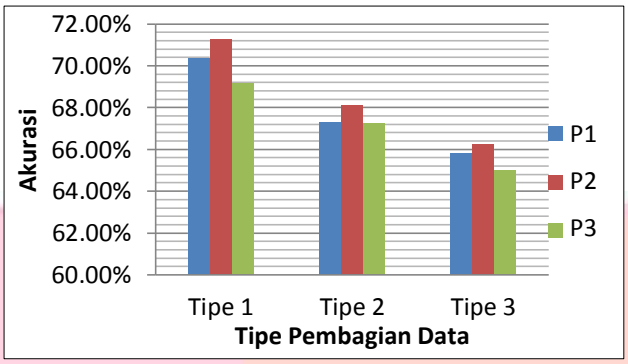
Setelah melakukan ekstraksi fitur, selanjutnya akan dilakukan proses klasifikasi. Sebelum melalukan klasifikasi, data akan dibagi menjadi 2 yaitu data training dan data testing. Pembagian data dilakukan menggunakan *percentage split* dimana data akan dibagi secara acak berdasarkan input sistem. Klasifikasi dilakukan mengguakan algoritma *Sequential Minimun Optimization* (SMO)

3.1. Analisis Pengujian

Pengujian pertama adalah pengujian pengaruh jumlah data *testing* dan data *training*. Pengujian ini dilakukan dengan cara membagi data *training* dan data *testing* menjadi 3 jenis *dataset*. *Dataset* pertama yaitu membagi data *training* sebanyak 90% dan data *testing* 10%, *dataset* kedua membagi data *training* sebanyak 80% dan data *testing* 20%, dan *dataset* terakhir adalah membagi data *training* sebanyak 70% dan data *testing* 30%. Pengujian pembagian data *training* dan data *tersing* dilakukan untuk melihat pengaruh pembagian jumlah data *training* dan data *testing* terhadap hasil klasifikasi. Berikut hasil pengujian yang telah dilakukan :

Tabel 2 : Tabel hasil pengujian pengaruh jumlah data testing dan data training

Tipe	Pembagian data		Akurasi			
	Data training	Data testing	P1	P2	P3	Rata-rata
1.	90%	10%	70,37%	71,27%	69,17%	70,27%
2.	80%	20%	67,32%	68,13%	67,26%	67,26%
3.	70%	30%	65,84%	66,26%	65,02%	65,71%



Gambar 3-0-3 : Grafik hasil pengujian pengaruh pembagian data training dan data testing

Berdasarkan grafik pada Gambar 3-2, dapat disimpulkan bahwa semakin banyak data training maka model yang dihasilkan semakin banyak. Sehingga pola pada data testing dapat dikenali oleh model hasil klasifikasi.

Pengujian selanjutnya adalah pengujian penggunaan fitur dasar dan fitur tambahan.

Tabel 0-2: Tabel hasil pengujian kombinasi fitur dasar dan fitur tambahan

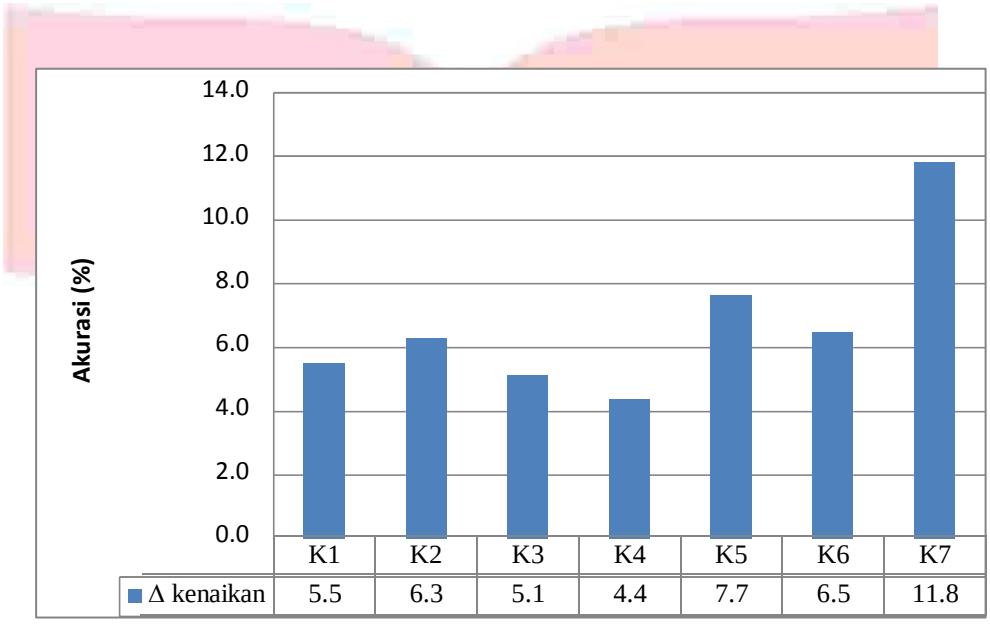
Kombinasi ke-	Kombinasi Fitur	Akurasi			
		P1	P2	P3	Rata-rata
1.	Semua fitur dasar	62,72%	62,84%	62,96%	62,84%
2.	Semua fitur dasar – *Predicate*	65,79%	66,04%	67,04%	66,29%
3.	Semua fitur dasar – *Phrase Type*	56,93%	54,68%	55,06%	55,56%
4.	Semua fitur dasar – *Voice*	62.92%	65.04%	64,04%	64,29%
5.	Semua fitur dasar – *Subcategorization*	64,17%	62,30%	60,90%	62,42%
6.	Semua fitur dasar – *Position*	48,06%	49,19%	47,44%	48,23%
7.	Semua fitur dasar – *Head Word*	56.30%	53,43%	54,31%	54,68%

Tabel 3-4 merupakan tabel hasil pengujian kombinasi fitur dasar dan fitur tambahan. Berdasarkan hasil pengujian tersebut, dapat dilihat bahwa fitur dasar yang berpengaruh adalah *phrase type*, *position*, dan *head word*. *Phrase type* merupakan fitur yang berpengaruh karena fitur ini mengekstrak kategori sintaksis yang berperan penting dalam proses klasifikasi. Misalnya pada ARG0 mempunyai kategori sintaksis yaitu NP atau PP. Penggunaan fitur *head word* menjadi berpengaruh karena *constituent* dengan *head word* tertentu lebih memiliki kecenderungan diklasifikasikan menjadi argumen tertentu. Misalnya kata benda yang memiliki *head word* : *John*, *brother*, *he* memiliki kecenderungan diartikan sebagai *speaker*. Sehingga akan memiliki kemungkinan yang tinggi akan diklasifikasikan ke dalam kelas ARG0. Fitur *position* menjadi penting karena pada fitur ini, kalimat akan dibagi menjadi 2 *substring* berdasarkan fitur *predicate*. Sebagian besar kalimat memiliki pola S-P-O-K, sehingga fitur *position* sebelah kiri merupakan subjek dan sebelah kanan merupakan objek serta keterangan. Dengan dibantu fitur *position*, algoritma akan lebih mudah melakukan klasifikasi karena telah diberikan pola seperti yang telah disebutkan.

Pengujian selanjutnya yaitu dengan menggunakan pengujian fitur dasar dikombinasikan dengan fitur tambahan yaitu *Named Entities in Constituent*, *Head Word POS*, dan *Syntactic Frame*.

Kombinasi ke-	Kombinasi Fitur	Akurasi				Δ kenaikan
		P1	P2	P3	Rata-rata	
K1	Semua fitur dasar + *Named Entity*	65,40%	67,36%	66,18%	66,31%	5,52%
K2	Semua fitur dasar + *Head Word POS*	66.99%	66,14%	67,27%	66,80%	6,30%
K3	Semua fitur dasar + *Syntactic Frame*	66,30%	63,69%	65,82%	65,27%	5,12%
K4	Semua fitur dasar + *Named Entity* + *Head Word POS*	65,89%	66,20%	67,02%	66,37%	4,39%
K5	Semua fitur dasar + *Named Entity* +	68,70%	67,97%	68,29%	68,32%	7,65%

	Syantactic Frame					
K6	Semua fitur dasar + *Syntactic Frame* + *Head Word POS*	67,24%	68,26%	67,86%	67,79%	6,49%
K7	Fitur dasar + *Named Entity* +*Head Word POS*+ *Syntactic Frame*	69,17%	71,27%	70,37%	70,27%	11,82%



Δ kenaikan dihitung dengan melakukan pengurangan rata-rata akurasi kombinasi fitur dengan rata-rata akurasi semua fitur dasar kemudian dibagi dengan rata-rata akurasi semua fitur dasar. Untuk kombinasi fitur tambahan, terdapat 2 jenis kombinasi yaitu kombinasi dengan 1 fitur tambahan, dan kombinasi menggunakan 2 fitur tambahan. Dapat terlihat jelas pada Tabel 3-4 bahwa kombinasi dengan 1 fitur tambahan memiliki akurasi lebih baik daripada akurasi jika hanya menggunakan fitur dasar. Namun penggunaan kombinasi 2 fitur memiliki hasil yang lebih baik dari pada yang hanya menggunakan kombinasi 1 fitur dasar.

Akurasi terbaik adalah 70,27% dengan Δ kenaikan sebesar 11,82% yaitu dengan penggunaan semua fitur dasar dan fitur tambahan *Named Entities in Constituent*, *Head Word POS*, dan *Syntactic Frame*.

4. Kesimpulan

Berdasarkan analisis terhadap pengujian yang telah dilakukan, dapat disimpulkan bahwa :

1. Fitur tambahan yaitu *Named Entities in Constituent*, *Head Word POS*, dan *Syntactic Frame* memberikan hasil klasifikasi yang cukup baik daripada hanya menggunakan fitur dasar dengan Δ kenaikan tertinggi yaitu 11,82% pada penggunaan semua fitur dasar beserta fitur *Named Entities in Constituent*, *Head Word POS*, dan *Syntactic Frame*. Sehingga semakin banyak fitur tambahan yang digunakan maka akurasi yang dihasilkan semakin tinggi.
2. Pengujian menggunakan *percentage split* menghasilkan akurasi tertinggi sebesar 70,27% dengan perbandingan jumlah data *training* dengan data *testing* yaitu 9:1. Semakin banyak data *training* yang digunakan dalam proses klasifikasi maka semakin baik model yang dihasilkan.
3. Fitur dasar yang berpengaruh dalam klasifikasi argumen semantik secara berurutan adalah *position*, *head word*, dan *phrase type*.
4. Hasil klasifikasi argumen semantik yang dilakukan sistem masih terdapat beberapa kesalahan. Kesalahan disebabkan oleh banyaknya nilai *null* pada hasil ekstraksi fitur *phrase type* dan *voice*.

Daftar Pustaka

[1] Sameer Pradhan and Daniel Jurafsky, "Support Vector Learning for Semantic Argument Classification," 2005.

[2] Nianwen Xue and Martha Palmer, "Calibrating Features for Semantic Role Labeling," 2005.

[3] Willy Sutina, "Pengaruh Algoritma Sequential Minimal Optimization pada Support Vector Machine untuk Klasifikasi Data," Telkom University, Bandung, Karya Ilmiah 2010.

[4] Suciadi James, "Studi Analisis Metode-Metode Parsing dan Interpretasi Semantik pada Natural Language Processing," *Jurnal Informatika*, vol. 2, No. 1, pp. 13-22, 2001.

[5] Alwi Hasan, *Kamus Besar Bahasa Indonesia*. Jakarta: Balai Pustaka, 2007.

[6] Gerber Matt and Chai Joyce Y, "Identification of Nominal Argument Structure Within and Across Sentence Boundaries," Michigan USA, 2009.

[7] Jiang Ping Zheng, Li Jia, and Ng Tou Hwee, "Semantics Argument Classification Exploiting Argument Interdependence," in *IJCAI International Joint Conference on Artificial Intelligence*, 2005.

[8] T Joachim, "Text Categorization with Support Vector Machine," University of Dortmund, TechReport 1997.

[9] John C. Platt, "Sequential Minimal Optimization : A Fast Algorithm for Training Support Vector Machine," Microsoft Research, USA, TechReport MSR-TR-98-14, 1998.

[10] Abidin Taufik Fuadi, "Accuracy Measure: Precision, Recall & F-Measure,".