SINONIM UNTUK EKSTRAKSI KATA KUNCI PADA PENGELOMPOKAN DOKUMEN MENGGUNAKAN FUZZY ASSOCIATION RULE MINING

Fahrur Rozi1), Rikie Kartadie2)
[1), 2)] Pendidikan Teknologi Informasi STKIP PGRI Tulungagung
Jl Mayor Sujadi Timur no.7. Tulungagung
Email : rozifahrur04@gmail.com[1)], rikie.kartadie@gmail.com[2)]

Abstrak

Pertumbuhan dunia digital dalam dokumen tekstual terutama di World Wide Web mengalami pertumbuhan pesat. Peningkatan dokumen tekstual ini menyebabkan terjadinya penumpukan informasi, sehingga diperlukan sebuah pengorganisasian yang efisien untuk pengelolaan dokumen tekstual. Salah satu metode yang dapat mengelompokkan dokumen dengan tepat adalah menggunakan fuzzy association rule. Tahap ekstraksi kata kunci serta tipe fuzzy yang digunakan berpengaruh terhadap kualitas pengelompokan dokumen. Penggunaan sinonim dalam ekstraksi kata kunci untuk mendapatkan suatu klaster label dapat memperluas makna dari klaster label, sehingga dapat diperoleh suatu meaningful klaster label, selain itu ambiguitas dan uncertainties yang terjadi di dalam aturan fuzzy logic systems (FLS) tipe-1 dapat diatasi dengan fuzzy set tipe-2. Penelitian ini mengusulkan sebuah metode yaitu sinonim untuk ekstraksi kata kunci pada pengelompokan dokumen menggunakan fuzzy association rule mining. Metode ini terdiri dari empat tahap, yaitu : preprocessing dokumen, ekstraksi key terms dari sinonim, ekstraksi kandidat klaster, dan konstruksi klaster tree. Pengujian terhadap metode ini dilakukan dengan tiga jenis data berbeda, yaitu Classic, Reuters, dan 20 Newsgroup. Pengujian dilakukan dengan membandingkan nilai overall f-measure dari metode tanpa semantic (non semantic), hipernim, dan sinonim. Berdasarkan pengujian didapatkan bahwa penggunaan sinonim dalam ektraksi kata kunci tidak mampu menghasilkan rata-rata overall f-measure yang lebih baik dibanding non semantic dan hipernim dengan nilai rata – rata overall f-measures sebesar 0.5372 untuk data classic, 0.3561 untuk data reuters, dan 0.5316 untuk data 20 newsgroup

***Kata kunci**: Fuzzy set tipe-2, sinonim, association rule, clustering dokumen.*

1. Pendahuluan

Clustering dokumen (pengelompokan teks) merupakan salah satu metode *text mining* yang dikembangkan untuk mengefisienkan pengelolaan teks serta peringkasan teks [**1**]. Beberapa hal yang dapat meningkatkan kualitas *cluster*ing dokumen antara lain : mengatasi dimensi tinggi yang diakibatkan besarnya jumlah dokumen dan jumlah kata dalam dokumen, meningkatkan skalabilitas agar mampu bekerja dengan jumlah dokumen dalam skala kecil ataupun besar (*scalable*), meningkatkan akurasi, memberikan label *cluster* yang bermakna, mampu mengatasi overlapping, serta memperhitungkan kesamaan konseptual istilah dari kata [**2**].

Beberapa metode telah dikembangkan untuk mendapatkan clustering dokumen dengan kualitas yang baik. Penggunaan fuzzy untuk clustering dokumen [**3**] dengan cara menerapkan *α-threshold Fuzzy Similiarity Classification Method* (α-FSCM) dan *Multiple Categories Vector Method* (MCVM). Penggunaan metode fuzzy tipe-1 ini mampu menghasilkan *cluster* yang *overlapping*. *High dimensionality* merupakan salah satu permasalahan dari *clustering* dokumen, untuk mengatasi permasalahan ini Beil dkk mengembangkan algoritma frequent itemset yaitu *Hierarchical Frequent Term-based Clustering* (HFTC) [**4**]. Namun, berdasarkan penelitian Fung,dkk bahwa HFTC tidak scalable [**5**]. Sehingga untuk menghasilkan metode yang scalable, Fung dkk mengembangkan metode *Frequent Itemset Hierarchical Clustering* (FIHC) yang merupakan algoritma hasil pengembangan *frequent-itemset* yang berasal dari *association rule mining* untuk membangun *hierarchical tree* untuk topik *cluster*.

Penggabungan antara *fuzzy* dan *association rule mining* yaitu *Fuzzy Frequent Itemset-Based Hierarchical Clustering* (F2IHC) mampu meningkatkan tingkat akurasi serta menghasilkan *cluster* yang *overlapping* dalam *clustering* dokumen [**6**]. Beberapa penelitian HFTC [**4**], FIHC [**5**], dan F^2IHC dengan *fuzzy* set tipe-2 [**7**] masih menggunakan *term* yang berada dalam dokumen teks sebagai *label cluster*. Meskipun hal tersebut dibenarkan, namun pelabelan *cluster* yang lebih umum akan memudahkan melakukan analisis terutama dalam domain pengetahuan [**8**].

Penelitan dengan menggunakan semantic dalam mengektraksi kata kunci dengan fuzzy association rule mining dapat memperluas kesamaan arti dari suatu kata dalam dokumen [**2**] [**9**]. Pada penelitian [**9**], penggunaan semantic hipernim untuk mengekstraksi kata kunci dapat meningkatkan nilai akurasi, karena mampu mengelompokkan suatu dokumen dengan karakteristik yang sama. Selain semantic menggunakan hipernim, penggunaan sinonim dalam pengelompokan dokumen mampu mengurangi dimensional yang tinggi [**10**] [**11**]. Dengan menggunakan sinonim permasalahan mengenai dokumen yang terdiri dari beberapa term yang berbeda namun memiliki arti yang sama dapat terselesaikan[**10**].

Ambiguitas dan *uncertainties* yang terjadi di dalam aturan *fuzzy logic systems* (FLS) tipe-1 [12] dapat mengurangi tingkat akurasi dalam clustering dokumen. Fuzzy set tipe-2 mampu menutupi kelemahan yang terdapat dalam fuzzy tipe-1 [12]. Penelitian [7] [9] [12] [13] [14] [15] dengan menggunakan fuzzy set tipe-2, menghasilkan bahwa fuzzy tipe-2 mampu mengatasi kelemahan yang terjadi pada fuzzy tipe-1 serta hasil dari fuzzy tipe-2 lebih baik dibanding dengan menggunakan fuzzy tipe-1. Selain itu penggunaan sinonim untuk mendapatkan suatu cluster label dapat memperluas makna dari cluster label, sehingga dapat diperoleh suatu meaningful cluster label. Oleh karena itu, penelitian ini bertujuan membangun metode sinonim untuk ekstraksi kata kunci pada pengelompokan dokumen menggunakan fuzzy association rule mining.

2. Pembahasan

Rancangan sistem dalam penelitian ini terdiri atas empat bagian utama yaitu : (1) preprocessing dokumen, (2) ekstraksi *key term* dari sinonim, (3) ekstraksi *candidate cluster*, dan (4) konstruksi *cluster tree*.

(1) *Preprocessing* Dokumen

Terdapat beberapa tahap yang dilakukan dalam preprocessing dokumen, yaitu : ektraksi term, penghilangan stopwords, stemming, dan seleksi term. Pada tahap awal, hasil dari ekstraksi dokumen dikumpulkan dalam suatu koleksi single word $T_D = \{t_1, t_2, \dots, t_n\}$. T_D menyatakan koleksi term (t) dalam dokumen (D), n menyatakan jumlah term dalam T_D. Hasil yang didapatkan dari ekstraksi term T_D digunakan sebagai input untuk dilanjutkan dengan penghilangan stopwords dan proses stemming. Algoritma stemming yang digunakan dalam penelitian ini adalah Porter stemmer yang ditemukan oleh Martin Porter pada tahun 1980. Langkah terakhir yang dilakukan dalam preprocessing dokumen adalah seleksi term dengan menghitung bobot tfidf (1) setiap term dalam T_D.

$$\mathrm{tf.idf}_{ij} = \frac{f_{ij}}{\sum_{j=1}^{m} f_{ij}} \times \log\left(\frac{|D|}{|\{d_i | t_j \in d_i, d_i \in D|\}|}\right), \quad (1)$$

dimana $\mathrm{tf.idf}_{ij}$ adalah bobot term t_j dalam dokumen d_i. Untuk mencegah bias dokumen yang panjang, bobot frekuensi term f_{ij} dinormalisasi dengan total frekuensi semua term dalam dokumen d_i. Variabel $|D|$ adalah jumlah seluruh dokumen dan $|\{d_i | t_j \in d_i, d_i \in D|\}|$ adalah jumlah dokumen yang memiliki term t_j.

(2) Ekstraksi *Key terms* dari Sinonim

Sinonim adalah suatu kata yang memiliki bentuk berbeda namun memiliki arti atau pengertian yang sama. Sinonim dari setiap term dilakukan pencarian berdasarkan dari Wordnet. Sehingga jika suatu term dan sinonimnya terdapat dalam satu dokumen yang sama, maka nilai frekuensinya akan dihitung sebagai satu kesatuan seperti yang dijabarkan dalam persamaan 3.

$$sf_{ij} = sf_{ij} + f_{ij}, \quad (3)$$

dimana f_{ij} adalah frekuensi term j dalam dokumen i, dan sf_{ij} adalah frekuensi sinonim dari term t_j dalam dokumen d_i.

(3) Ekstraksi Kandidat *Cluster*

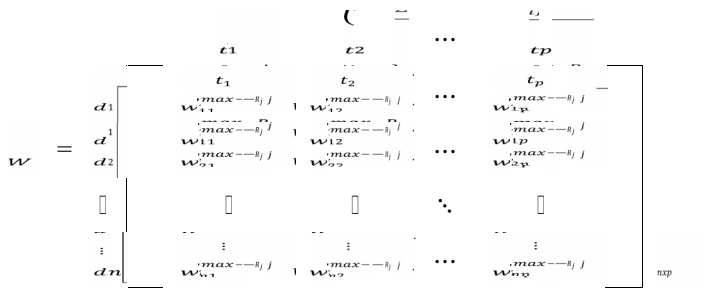
Terdapat empat proses yang harus dilalui untuk mendapatkan kandidat cluster, diantaranya : menghitung nilai membership function dengan fuzzy set tipe-2, menemukan candidate-1 itemset, menemukan candidate-2 itemset, dan seleksi kandidat cluster. Fuzzy set tipe-2 dalam penelitian ini menggunakan dua jenis tipe fungsi keanggotaan, yaitu : fungsi kenggotaan jenis triangular sebagai LMF (Lower Membership Function) dan fungsi keanggotaan jenis trapezoidal sebagai UMF (Upper Membership Function). Setiap term j dalam dokumen i dengan frekuensi f_{ij} memiliki bobot $w_{ij}^{r,z}$ yang menyatakan bobot atau fungsi keanggotaan term j dalam dokumen i yang terdapat dalam wilayah fungsi keanggotaan fuzzy set tipe-2. Variabel r dalam $w_{ij}^{r,z}$ merupakan variabel linguistik, yaitu : Low, Medium, dan High. Sementara z merepresentasikan LMF dan UMF.
Hasil bobot fuzzy tipe-2 dari setiap term selanjutnya akan digunakan untuk menentukan candidate 1-frequent itemset. Untuk menemukan term yang digunakan sebagai candidate 1-itemset, setiap term dilakukan perhitungan nilai support. Perhitungan nilai support didapatkan dari hasil perbandingan antara nilai bobot fuzzy dengan jumlah dokumen. Hasil term j yang diperoleh dari candidate 1-itemset akan diasosiasikan terhadap term yang lain untuk mendapatkan candidate 2-itemset. Setiap pasang term yang memiliki nilai support dan confidence lebih dari minimum support dan minimum confidence akan dijadikan sebagai candidate 2-itemset. Hasil dari candidat 1-itemset dan candidate 2-itemset dijadikan sebagai candidat cluster set $C_D = \{\tilde{c}_1^1, \dots, \tilde{c}_{l-1}^2, \tilde{c}_l^q, \dots, \tilde{c}_k^q\}$, dimana D merupakan koleksi dokumen, q merupakan jumlah q-itemset, dan k adalah jumlah semua kandidat cluster c yang didapatkan dari candidat 1-itemset dan candidate 2-itemset.

(4) *Konstruksi cluster tree*

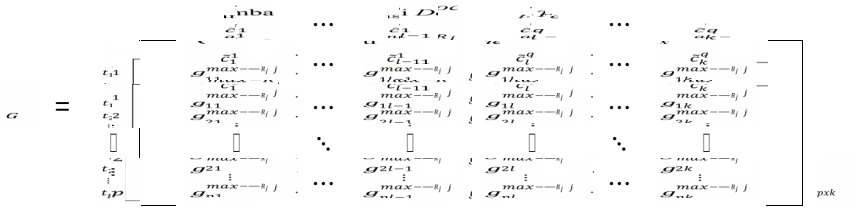
Untuk membentuk cluster tree dibutuhkan beberapa tahap, yaitu membentuk Document-Term Matrix (DTM), membentuk Term-Cluster Matrix (TCM), dan membentuk Document-Cluster Matrix (DCM). Document-Term Matrix (DTM) atau matriks $W = [w_{ij}^{\max - R_j}]$, dimana $w_{ij}^{\max - R_j}$ adalah bobot (nilai fungsi keanggotaan) dari term t_j dalam dokumen d_i. Matriks ini merupakan representasi dari kumpulan nilai maksimum fungsi keanggotaan dari tiap term t_j dalam dokumen d_i dengan ukuran $n \times p$, dengan n adalah jumlah dokumen d_i dalam koleksi dokumen D, dan p adalah jumlah key term t dari hasil ekstraksi candidate 1-itemset. Ilustrasi dari matriks DTM terdapat pada gambar 1.
Setelah terbentuk matriks DTM, selanjutnya adalah pembentukan Term-Cluster Matrix (TCM) atau matriks

$G = [g_{jl}^{max-R_j}]$ dengan ukuran $p \times k$, dimana p adalah jumlah key term t dari hasil ekstraksi candidate 1-itemset, dan k adalah jumlah kandidat cluster $\tilde{c}_l^q$ dari ekstraksi candidat 1-itemset dan candidate 2-itemset yang diilustrasikan dalam Gambar 2. Variabel $g_{jl}^{max-R_j}$ menyatakan derajat tingkat kepentingan suatu key *term* t_j dalam suatu candidat cluster $\tilde{c}_l^q$ yang dijabarkan dalam persamaan (4).

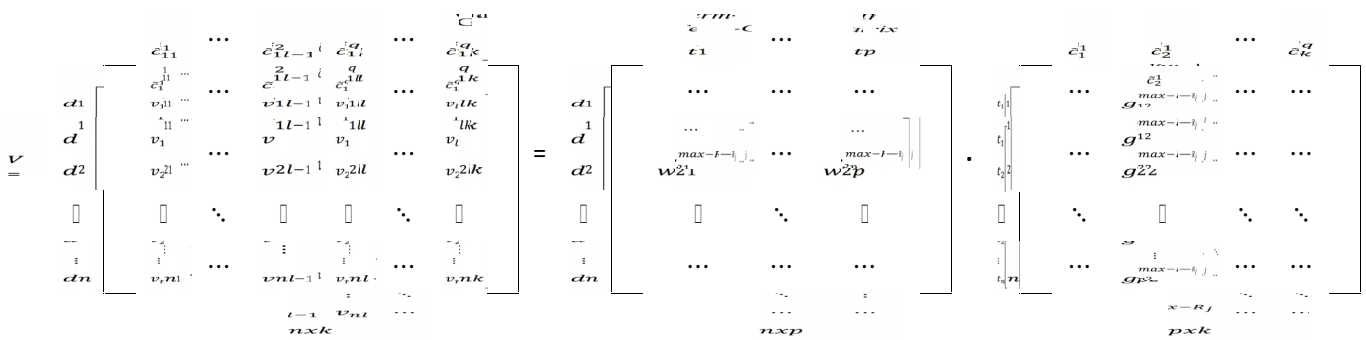
$$g_{jl}^{max-R_j} = \frac{score\,(\tilde{C}_l^q)}{\sum_{i=1}^{n} W_{ij}^{max-R_j}}, dimana,\ score\,(\tilde{C}_l^q) = \begin{cases} \sum_{d_i \in \tilde{c}_l^1, t_j \in L_1} W_{ij}^{max-R_j} \ if\ q = 1, \\ \frac{\sum_{d_i \in \tilde{c}_l^q, t_j \in L_1} W_{ij}^{max-R_j}}{\lambda}, else \end{cases} \quad (4)$$



Gambar 1. Ilustrasi *Document-Term Matrix*



Gambar 2. Ilustrasi *Term-Cluster Matrix*



Gambar 3. Ilustrasi Document-Cluster Matrix

Pada persamaan (3), $w_{ij}^{max-R_j}$ adalah bobot (nilai fungsi keanggotaan) dari term t_j dalam dokumen d_i, dengan λ merupakan minimum confidence. Hasil dari terbentuknya matriks DTM dan matriks TCM digunakan untuk membangun matriks Document-Cluster Matrix (DCM). DCM memiliki ukuran matriks $n \times k$ yang merupakan turunan dari hasil perkalian antara matriks DTM dan matriks TCM. Matriks DCM secara keseluruhan dapat diilustrasikan dalam Gambar 3.

Setelah ditemukan matriks DCM, maka langkah selanjutnya adalah melakukan tree pruning. Tree pruning adalah melakukan suatu kegiatan untuk mengganti suatu subtree dengan suatu leaf. Tree pruning dalam Clustering dokumen bertujuan untuk menggabungkan beberapa cluster sejenis dan memilki kemiripan sama yang berada di level 1 , sehingga akan menghasilkan cluster yang lebih baik. Setiap pasang cluster pada level 1 dapat dihitung nilai kemiripannya dengan menggunakan ukuran similiarity yaitu inter_sim. Pasangan cluster yang memiliki nilai inter_sim tertinggi akan di gabung hingga nilai dari inter_sim dari seluruh

pasangan cluster pada level 1 kurang dari nilai minimum threshold dari inter_sim. Pengukuran kemiripan antara cluster (c_x^1) dengan cluster (c_y^1) menggunakan inter_sim dapat didefinisikan dalam persamaan (5).

$$\text{inter}_{\text{sim}}(c_x^1, c_y^1) = \frac{\sum_{d_i \in c_x^1, c_y^1}^{n} v_{ix} \times v_{iy}}{\sqrt{\sum_{d_i \in c_x^1}^{n} (v_{ix})^2 \times \sum_{d_1 \in c_y^1}^{n} (v_{iy})^2}} \quad (5)$$

dimana v_{ix} dan v_{iy} adalah nilai yang diperoleh dari hasil perhitungan DCM (Document Cluster Matrix). Variabel x merupakan term pertama dan y merupakan term kedua. Nilai dari inter_sim memiliki rentang antara [0,1] yang didapat dari penjumlahan hasil perkalian antara v_{ix} dan v_{iy} sebanyak n dokumen dimana cluster (c_x^1) dan cluster (c_y^1) merupakan kandidat cluster dari dokumen d_i. Hasil penjumlahan tersebut akan dibagi dengan akar kuadrat dari penjumlahan kuadrat v_{ix} sebanyak n dokumen yang dikalikan dengan penjumlahan kuadrat v_{iy} sebanyak n dokumen.

Evaluasi

F-measure merupakan salah satu perhitungan evaluasi dalam sistem temu kembali informasi yang mengkombinasikan antara recall dan precision. Ukuran yang menampilkan timbal balik antara recall dan precision adalah f-measure yang merupakan bobot harmonic mean dari recall dan precision. Recall merupakan dokumen yang ditemukan dan relevan dengan query yang dimasukkan oleh user dalam suatu sistem temu balik informasi. Recall terkait dengan kemampuan suatu sistem untuk mendapatkan dokumen yang relevan. Sementara, precision merupakan jumlah kelompok dokumen yang relevan dari total jumlah dokumen yang ditemukan oleh sistem. Precision terkait dengan tingkat efektivitas sistem temu balik informasi. F(C) dapat didefinisikan dalam persamaan 6.

$$F(C) = \sum_{l_j \in L} \frac{|l_j|}{|D|} \max_{c_i \in C}\{F\},$$

$$\text{dimana } F = \frac{2PR}{P+R}, P = \frac{|c_i \cap l_j|}{|c_i|}, R = \frac{|c_i \cap l_j|}{|l_i|}, \quad (6)$$

dimana |D| merupakan jumlah dokumen dalam dataset D. Variabel C merupakan cluster yang terdapat dalam sistem. Variabel L merupakan label kelas yang diperoleh dari dataset. $|c_i|$ merupakan jumlah dokumen dalam cluster C ($c_i \in C$). $|l_j|$ merupakan jumlah dokumen dalam class L ($l_j \in L$). Persamaan $|c_i \cap l_j|$ menyatakan jumlah dokumen yang berada tepat dalam kedua cluster c_i dan l_j.

Dataset

Tiga jenis dataset yang berbeda digunakan dalam penelitian ini. Dataset tersebut adalah sebagai berikut :

(1) Classic : merupakan dataset dari abstract jurnal ilmiah yang terdiri atas kombinasi empat kelas CACM, CISI, CRANFIELD, dan MEDICAL. Jumlah data yang digunakan dalam dateset classic ini berjumlah 1000 data, dimana setiap kelas, yaitu : CACM, CISI, CRANFIELD dan MEDICAL berjumlah 250 data. CACM merupakan jurnal dengan topik akademis, CISI merupakan jurnal dengan topik informatian retreival, CRAN merupakan jurnal dengan topik sistem penerbangan, dan MED merupakan jurnal dengan topik medis. (2) Reuters : merupakan dataset yang berasal dari koleksi Reuters newswire. Dalam dataset ini terdapat beberapa kelas, diantaranya reut2-001, reut2-002, reut2-003, dan reut2-004. Masing-masing kelas terdiri dari 250 data, sehingga total keseluruhan data adalah 1000 data, dan (3) 20 Newsgroup : merupakan kumpulan dari dokumen newsgroup yang terbagi kurang lebih 20 kelas berbeda kelas yang digunakan dalam dataset 20 Newsgroup adalah 4 kelas yang terdiri dari : *comp.sys.mac.hardware, rec.sport.baseball, sci.space,* dan *talk.politics.mideast.* Masing − masing kelas terdiri dari 150 data, sehingga total terdapat 600 data.

Pengujian

Pengujian terhadap metode yang diusulkan dilakukan dengan tiga skenario berbeda, yaitu : (1) Pengujian tanpa menggunakan hipernim dan sinonim dalam ekstraksi kata kunci (non semantic), (2) pengujian dengan menggunakan hipernim, dan (3) pengujian dengan menggunakan sinonim. Setiap skenario pengujian dilakukan untuk mengetahui pengaruh jumlah dataset terhadap nilai *overall f-measures*. Jumlah dataset yang digunakan adalah 200, 400. 600. 800, dan 1000. Pengujian terhadap skenario 1 dan skenario 2 yaitu pengujian non semantic dan penggunaan hipernim menggunakan hasil dari penelitian sebelumnya [**9**] dengan dataset yang sama. Hasil dari pengujian ini untuk dataset Classic terdapat pada Tabel 1 dan Gambar 2, untuk dataset Reuters terdapat pada Tabel 2 dan Gambar 3, dan untuk dataset 20 Newsgroup terdapat pada Tabel 3 dan Gambar 4.

Tabel 1. Hasil Pengaruh Jumlah Data Terhadap Overall f-measure pada Data Classic

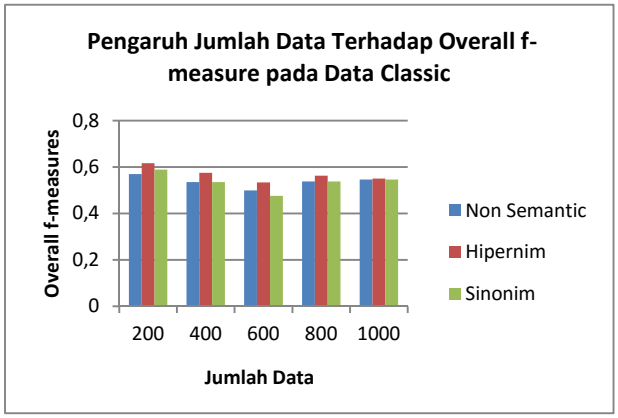
Jumlah Data	*Overall f-measure*		
	Non Semantic	**Hipernim**	**Sinonim**
200	0.5694	0.6174	0.5890
400	0.5358	0.5759	0.5358
600	0.4992	0.5335	0.4768
800	0.5382	0.5627	0.5382
1000	0.5461	0.5510	0.5460

Tabel 2. Hasil Pengaruh Jumlah Data Terhadap Overall f-measure pada Data Reuters

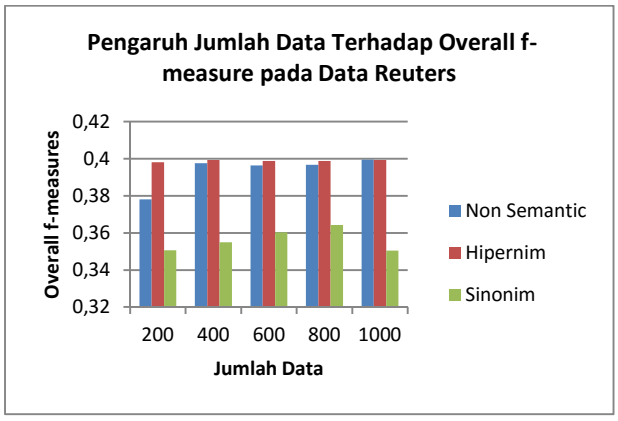
Jumlah Data	*Overall f-measure*		
	Non Semantic	Hipernim	Sinonim
200	0.3780	0.3980	0.3507
400	0.3975	0.3992	0.3550
600	0.3964	0.3988	0.3602
800	0.3966	0.3988	0.3642
1000	0.3994	0.3993	0.3505

Tabel 3. Hasil Pengaruh Jumlah Data Terhadap Overall f-measure pada Data 20 Newsgroup

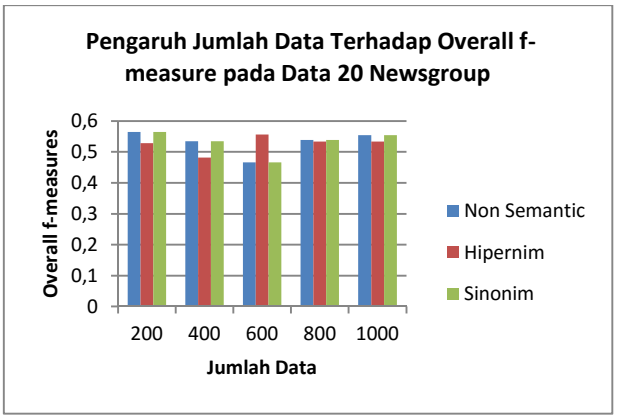
Jumlah Data	*Overall f-measure*		
	Non Semantic	Hipernim	Sinonim
200	0.5651	0.5286	0.5651
400	0.5343	0.4823	0.5343
600	0.4658	0.5565	0.4658
800	0.5387	0.5336	0.5387
1000	0.5542	0.5335	0.5542



Gambar 4. Grafik Pengaruh Jumlah Data Terhadap Overall f-measure pada Data Classic



Gambar 5. Grafik Pengaruh Jumlah Data Terhadap Overall f-measure pada Data Reuters



Gambar 6. Grafik Pengaruh Jumlah Data Terhadap Overall f-measure pada Data 20 Newsgroup

Berdasarkan Tabel 1, Tabel 2, dan Tabel 3 serta Gambar 4, Gambar 5, dan Gambar 6 diketahui bahwa setiap pada dataset memiliki hasil overall f-measures yang berbeda. Pada dataset Classic sinonim pada metode yang diusulkan memiliki nilai overall f-measure yang lebih rendah dibandingkan pada metode yang menggunakan hipernim dan metode yang tidak menggunakan semantic (non semantic). Metode dengan menggunakan sinonim memiliki rata-rata overall f-measure lebih rendah dibanding metode dengan hipernim dengan nilai rata-rata overall f-measure sebesar 0.5372. Sinonim pada dataset Classic hanya menambah jumlah term saja, namun term dari sinonim tidak mampu untuk mengelompokkan dokumen dengan karateristik yang sama. Berbeda dengan penggunaan hipernim pada dataset Classic mampu memperluas makna dari term sehingga dokumen-dokumen dengan karateristik yang sama namun tidak memiliki term yang sama dapat dikelompokkan menjadi satu kelompok yang sama karena memiliki term yang sama terhadap hipernim. Hal ini dikarenakan penambahan sinonim pada beberapa dokumen hanya akan menambah jumlah term yang akan dianggap sebagai noise. Namun pada pengujian dataset classic ditemukan bahwa pada jumlah data 200 nilai overall f- measures dari metode dengan sinonim lebih baik dibanding metode dengan non semantic, hal ini terjadi karena pada dataset 200 sinonim mampu mengelompokkan dokumen dengan karateristik yang sama.

Pada dataset Reuters memiliki hasil yang hampir sama dengan dataset Classic, yaitu penggunaan sinonim pada metode yang diusulkan memiliki nilai overall f-measure yang lebih rendah dibandingkan pada metode yang menggunakan hipernim dan metode yang tidak menggunakan semantic (non semantic). Metode dengan menggunakan sinonim memiliki rata-rata overall f-measure lebih rendah dibanding metode dengan hipernim dengan nilai rata-rata overall f-measure sebesar 0.3561. Sinonim pada dataset Reuters hanya menambah jumlah term saja, namun term dari sinonim tidak mampu untuk mengelompokkan dokumen dengan karateristik

yang sama. Berbeda dengan penggunaan hipernim pada dataset Reuters mampu memperluas makna dari term sehingga dokumen-dokumen dengan karateristik yang sama namun tidak memiliki term yang sama dapat dikelompokkan menjadi satu kelompok yang sama karena memiliki term yang sama terhadap hipernim. Hal ini dikarenakan penambahan sinonim pada beberapa dokumen hanya akan menambah jumlah term yang akan dianggap sebagai noise.

Sementara pada dataset 20 Newsgroup pengunaan hipernim pada metode dengan hipernim tidak memberikan nilai overall f-measures yang lebih rendah dibandingkan pada metode yang menggunakan hipernim dan metode yang tidak menggunakan semantic (non semantic). Metode dengan menggunakan sinonim memiliki rata-rata overall f-measure lebih rendah dibanding metode dengan hipernim dengan nilai rata-rata overall f-measure sebesar 0.5316. Sinonim pada dataset 20 Newsgroup hanya menambah jumlah term saja, namun term dari sinonim tidak mampu untuk mengelompokkan dokumen dengan karateristik yang sama. Berbeda dengan penggunaan hipernim pada dataset 20 Newsgroup mampu memperluas makna dari term sehingga dokumen-dokumen dengan karateristik yang sama namun tidak memiliki term yang sama dapat dikelompokkan menjadi satu kelompok yang sama karena memiliki term yang sama terhadap hipernim. Hal ini dikarenakan penambahan sinonim pada beberapa dokumen hanya akan menambah jumlah term yang akan dianggap sebagai noise.

3. Kesimpulan

Berdasarkan serangkaian pengujian dengan metode yang diusulkan dapat diambil kesimpulan bahwa penggunaan sinonim dalam ektraksi *key terms* memiliki akurasi yang lebih rendah dibandingkan dengan menggunakan hipernim. Hal ini disebabkan dalam ekstraksi kata kunci menggunakan sinonim hanya akan menambah jumlah *term* tanpa mampu mengelompokkan dokumen yang memiliki karakteristik yang sama sehingga dianggap sebagai *noise* dan mengurangi akurasi.

Daftar Pustaka

[1] Congnan Luo, Yanjun Li, and Soon M. Chung, "Text document clustering based on neighbors," *Data & Knowledge Engineering*, vol. 68, no. 1, pp. 1271-1288, Juli 2009.

[2] Chun Lieng Chien, Frank S.C Tseng, and Tyne Liang, "An Integration of WordNet and fuzzy association rule mining for multi-label document clustering," *Data & Knowledge Engineering*, vol. 69, no. 1, pp. 1208-1226, September 2010.

[3] Ridvan Saracoglu, Kemal Tutuncu, and Novruz Allahverdi, "A new approach on search for similiar documents with multiple categories using fuzzy clustering," *Expert Systems with Applications*, pp. 2545-2554, 2008.

[4] Florian Beil, Martin Ester, and Xiaowei Xu, "Frequent Term-Based Text Clustering," *Proc. of Int'l Conf. on knowledge Discovery and Data Mining*, pp. 436-442, 2002.

[5] B.C.M Fung, K. Wang, and M. Ester, "Hierarchical document clustering using frequent itemset," Simon Fraser University, 2002.

[6] Ling Chun Chen, Frank S.C Tseng, and Tyne Liang, "Mining fuzzy frequent itemset for hierarchical document clustering," *Information Processing and Management*, vol. 46, pp. 193-211, Oktober 2010.

[7] Susiana Sari, "Clustering berbasis dokumen secara hierarki barbasis fuzzy set tipe-2 trapezoidal dan triangular dari frequent itemset," Institut Teknologi Sepuluh Nopember, 2012.

[8] Yuen Hsien Tseng, "Generic title labeling for clustered documents," *Expert Systems with Applications*, vol. 37, pp. 2247-2254, 2010.

[9] Fahrur Rozi, Chastine Fatichah, and Diana Purwitasari, "Ektraksi Kata Kunci Berdasarkan Hipernim dengan Inisialisasi Klaster Menggunakan Fuzzy Association Rule Mining pada Pengelompokan Dokumen ," *Jurnal Teknologi Informasi (JUTI)*, vol. 13, no. 2, pp. 190-197, July 2015.

[10] Rekha Baghel and Renu Dhir, "A Frequent Concepts Based Document Clustering Algorithm," *International Journal of Computer Applications*, vol. 4, no. 5, pp. 6-12, July 2010.

[11] G. Bharathi and D. Venkatesan, "Improving Information Retreival Using Document Cluster and Semantic Synoym Extraction," *Journal of Theoretical and Applied Information Technology*, vol. 36, no. 2, pp. 167-173, February 2012.

[12] Jerry M. Mendel and Robert I. Bob John, "Type-2 Fuzzy Sets Made Simple," *IEEE Transactions on Fuzzy System*, pp. 117-127, 2002.

[13] Janusz T. Starczewski, "Centroid of triangular and Gaussian type-2 fuzzy sets," *Information Sciences*, vol. 280, pp. 289-306, Mei 2014.

[14] Cengiz Kahraman, Basar Oztaysi, Irem Ucal Sari, and Ebru Turanoglu, "Fuzzy analytic hierarchy process with interval type-2 fuzzy sets," *Knowledge-Based Systems*, vol. 59, pp. 48-57, Februari 2014.

[15] Janusz T. Starczewski, "Efficient triangular type-2 fuzzy logic systems," *International Journal of Approximate Reasoning*, pp. 799-811, 2009.

Biodata Penulis

Fahrur Rozi, S.Kom., M.Kom., memperoleh gelar Sarjana Komputer (S.Kom), Universitas Brawijaya. Memperoleh gelar Magister Komputer (M.Kom) Jurusan Teknik Informatika, Fakultas Teknologi Informasi, ITS Surabaya pada tahun 2015. Saat ini menjadi dosen di STKP PGRI Tulungagung.

Rikie Kartadie, S.T., M.Kom., Mendapatkan gelar Sarjana Teknik (S.T.) pada tahun 2001 dari Universitas Pembangunan Nasional "Veteran" Yogyakarta, mendapatkan gelar Master Komputer (M.Kom.) pada Mei 2014 Konsentrasi Sistem Informasi dari STMIK AMIKOM Yogyakarta, dan menjadi dosen di STKP PGRI Tulungagung.