Otomatisasi Peringkasan Dokumen Sebagai Pendukung Sistem Manajemen Surat

Ahmad Najibullah[1], Wang Mingyan[2]
[1,2]Fakultas Teknologi Informasi, Universitas Nanchang, Nanchang, Republik Rakyat Tiongkok

E-mail: [1]ahmednajibullah@gmail.com

Abstrak

Peringkasan dokumen adalah proses penyajian kembali dokumen dalam bentuk yang lebih singkat tanpa membuang informasi penting yang terdapat dalam dokumen tersebut. Dalam penelitian ini, peneliti menggunakan metode Naïve Bayes untuk menghasilkan ringkasan sebuah dokumen. Objek dalam penelitian ini berupa dokumen yang berbentuk surat. Dalam proses peringkasan dokumen, penghitungan probabilitas didasarkan pada fitur teks yang ada dalam surat, diantaranya adalah frekuensi kata, kata kunci, frase kunci, dan kata yang termasuk dalam kelas entitas atau numerik. Hasil uji coba menunjukkan bahwa tingkat kompresi adalah 53.67% dengan informasi penting yang tersedia dalam ringkasan mencapai 96.67% dari dokumen asli.
Kata kunci: otomatisasi, peringkasan, dokumen, surat, Naïve Bayes.

Abstract

Summirising document is a restatement process of document in a shorter form without discarding important information contained in the document. In this study, we used Naïve Bayes approach to generate a summary of a document. The object of this study is a document in the form of letter. In the process of summarising document, the probability is calculated based on the features of the existing text in the letter, including the frequency of words, keywords, key phrases and words which are included in the class of entities or numerical. Experimental results show that the compression rate is 53.67% with the important information provided in the summary reaches 96.67% of the original document
Key word: *automation, summarising, documents, letters, Naïve Bayes.*

1. Pendahuluan

Setiap organisasi mempunyai perumusan tujuan yang ingin dicapai. Semakin berkembang organisasi berimbas pula pada data yang harus dikelola oleh organisasi tersebut. Terdapat banyak data yang harus dikelola dalam organisasi secara baik. Misalnya data surat menyurat.

Fungsi surat dalam organisasi sangat penting, karena selain sebagai alat komunikasi surat juga bisa mendukung berjalannya roda organisasi sesuai visi dan misinya. Organisasi dengan aktifitas surat-menyurat yang banyak dan pengelolaan yang baik bisa menjadi parameter bahwa organisasi mempunyai aktifitas tinggi. Di sisi lain, walaupun organisasi dengan aktifitasnya tinggi, tetapi tidak memiliki metode pengelolaan surat yang baik maka organisasi tersebut secara administratif bisa dikatakan sebagai organisasi yang tidak efektif. Atas dasar pentingnya fungsi surat, maka dibutuhkan sistem yang bisa membantu organisasi dalam mengelola surat.

Di samping itu, semakin banyaknya surat dalam organisasi menjadi landasan dalam penelitian ini untuk membuat aplikasi yang dapat membantu pengelola organisasi dalam mencari surat dalam *database*. Dalam penelitian ini, penelitian ini menggunakan peringkasan dokumen secara otomatis untuk menampilkan *review* surat dalam hasil pencarian. Hal ini tentunya bisa membantu pengguna sehingga lebih cepat dalam mencari surat yang dimaksud.

Peringkasan dokumen, dalam penelitian ini surat organisasi sebagai objek peringkasan, adalah penulisan kembali sebuah dokumen dalam format yang lebih pendek dan merepresentasikan dokumen asli tanpa kehilangan informasi penting yang tersedia dalam dokumen asli. Manusia biasanya melakukan tugas ini setelah membaca dokumen dan memahaminya, kemudian memilih poin-poin yang penting dan merangkai kembali dalam bentuk singkat. Penggunaan komputer untuk meniru pekerjaan yang biasanya dikerjakan manusia ini disebut otomatisasi peringkasan dokumen (Binwahlan, 2011).

Tujuan dari penelitian ini adalah penyajian dokumen dalam format yang lebih singkat dengan tetap menjaga karakter dari dokumen asli, sehingga pembaca tidak kehilangan informasi penting dokumen asli. Secara garis besar peringkasan dokumen dibagi menjadi dua pendekatan, yaitu

pendekatan abstraktif dan ekstraktif. Sebagian besar peneliti menggunakan pendekatan ekstraksi untuk menghasilkan ringkasan dari dokumen tertentu. Penelitian ini juga menggunakan pendekatan ekstraksi.

Dalam penelitian ini digunakan metode Naïve Bayes untuk melakukan proses peringkasan dokumen berbahasa Indonesia. Untuk membangun graf dokumen, digunakan fitur-fitur teks. Perbedaan dengan penelitian-penelitian sebelumnya adalah penggunaan ekstraksi frase kunci dari sebuah kalimat dan penggunaan *part-of speech* (POS) sebagai salah satu fitur dari teks. Ektrasksi frase kunci ini dilakukan berdasarkan pola berdasarkan POS.

Penggunaan graf untuk proses peringkasan dokumen telah dilakukan oleh beberapa peneliti. Erkan et al. memperkenalkan *stochastic graph* untuk menghitung level informasi dokumen dalam *Natural Language Processing* (NLP) (Erkan & Radev, 2004). Pendekatan yang dilakukan adalah LexRank, metode ini digunakan untuk menghitung tingkat informasi suatu kalimat berbasis sentralitas *eigen vector* dalam graf. Mihalcea et al. melakukan peringkasan dokumen dengan cara ekstrasi dengan metode TextRank (Mihalcea, 2004). TekxRank adalah model peringkat berbasis graf.

Mirchev et al. menggunakan *extended graph* untuk menghasilkan ringkasan dengan objek multi-dokumen (Mirchev & Last, 2014). Dalam penelitiannya, Mirchev merepresentasikan dokumendokumen dalam sebuah graf, dengan koneksi antara node adalah bobot relasi antar kalimat.

Nandhini et al. menggunakan metode *supervised machine learning* untuk menghasilkan ringkasan ekstraktif pada dokumen dalam bidang sain dan pendidikan. Nandhini juga mempertimbangkan fitur-fitur teks untuk memprediksi kalimat yang akan diekstrak dari dokumen (Nandhini & S, Improving Readability through Extractive Summarization for Learners with Reading Difficulties, 2013). Dalam evaluasi hasil ektraksi yang dihasilkan sistem, Nandhini menggunakan *F-measure* dan tingkat pemahaman yang ditentukan oleh pembaca.

Beberapa peneliti menggunakan algoritma genetika untuk melakukan proses ini, diantaranya adalah Nandhini (Nandhini & S, Use of Genetic Algorithm f or Cohesive Summary Extraction to, 2013), Mine Berker (Berker, 2011), Khosravi (Dehkordi, Kumarci, & Khosravi, 2009), Qazvinian (Qazvinian, Hassanabadi, & Halavati, 2008), Aristoteles (Aristoteles, Widiarti, & Wibowo, 2014), dan lain sebagainya. Algoritma genetika pertama kali diperkenalkan oleh Goldberg (Goldberg, 1989). Siklus algoritma genetika terdiri dari beberapa bagian, yaitu populasi awal, evaluasi *fitness*, seleksi individu, *crossover*, mutasi, dan populasi baru.

Penelitian peringkasan dokumen otomatis dalam bahasa Indonesia sebelumnya pernah dilakukan. Tetapi hanya sedikit peneliti yang menjadikan bahasa Indonesia sebagai objek penelitiannya. Budhi et al. mengembangkan sistem untuk peringkasan dokumen berbahasa Indonesia berbasis graf (Budhi, Intan, R, & R, 2007). Metode yang digunakan adalah algoritma *exhaustive shortest path*. Dalam penelitiannya Budhi juga mempertimbangkan model paragraf deduktif dan induktif. Untuk menguji hasilnya, Budhi menggunakan metode interview kepada pembaca, pembaca diberikan dokumen asli, ringkasan, dan beberapa pertanyaan. Jika pertanyaan-pertanyaan yang disajikan kepada pembaca bisa terjawab berdasarkan hasil ringkasan, maka ringkasan yang dihasilkan sistem dinilai baik. Peneliti lain yang menjadikan bahasa Indonesia sebagai objek penelitiannya adalah (Nandhini & S, 2013). Aristoteles et al. menggunakan algoritma genetika dalam melakukan proses peringkasan dokumen berbahasa Indonesia.

2. Metode Penelitian

Tujuan dari penelitian ini adalah untuk menghasilkan ringkasan dokumen berbahasa Indonesia. Untuk menghasilkan ringkasan tersebut, penelitian ini menggunakan metode Naïve Bayes. Cara kerja metode ini adalah menghitung probabilitas setiap kalimat S dalam dokumen D. Penghitungan probabilitas ini didapatkan dari data pelatihan dan penghitungan setiap fitur teks yang terdapat kalimat S. Formula Naïve Bayes adalah seperti pada persamaan (1).

$$P(s \in S \mid F_{s1}, F_{s2}, \dots F_{sn}) = \frac{P(F_{s1}, F_{s2}, \dots F_{sn} \mid s \in S)\, P(s \in S)}{P(F_{s1}, F_{s2}, \dots F_{sn})} \quad (1)$$

Dengan asumsi bahwa setiap fitur teks memiliki probabilitas sendiri, maka penghitungan untuk menentukan kelas dari suatu kalimat adalah sebagai berikut:

$$P(s \in S) \mid F_{s1}, F_{s2}, \dots F_{sn} = \frac{\prod_{j=1}^{n} P(F_{sj} \mid s \in S) P(s \in S)}{\prod_{j=1}^{n} P(F_{sj})} \quad (2)$$

$P(s \in S)$ adalah konstan dan nilai $P(F_{sj}|s \in S)$ dan $P(F_{sj})$ dapat didapakan dari data pelatihan. Metode klasifikasi naïve bayes dihitung dengan menggunakan persamaan (2), nilai dari hasil ini akan menentukan apakah kalimat tersebut masuk dalam keluaran ringkasan atau tidak termasuk hasil ringkasan.

2.1. Fitur-fitur Teks

Ekstraksi fitur teks ini tergantung pada tujuan dan target sistem ringkasan yang akan dibuat. Dalam penelitian ini, penelitia ini menggunakan surat organisasi sebagai objek penelitian. Maka dari itu, teks fitur yang akan diekstrak sebagai penghitungan probabilitas disesuaikan dengan karakteristik dokumen surat yang akan diringkas. Terdapat lima (5) fitur yang digunakan dalam penghitungan karakateristik dokumen.

Fitur teks yang pertama adalah panjang suatu kalimat. Panjang suatu kalimat bisa mempengaruhi pembaca dalam memahami suatu dokumen. Diasumsikan bahwa kalimat yang terlalu panjang atau terlalu pendek itu lebih sulit dipahami. Jika kalimat terlalu panjang maka pembaca kesulitan dalam menemukan poin penting dalam kalimat, sebaliknya jika kalimat terlalu pendek maka dimunkingkan kalimat tersebut tidak memuat poin penting, perhatikan persamaan (3).

$$f_1 = \frac{Panjang(S) * \#(kalimat\ dalam\ dokumen)}{Panjang\ (dokumen)} \quad (3)$$

Selain fitur yang telah disebutkan di atas, juga terdapat beberapa fitur yang digolongkan berdasarkan tematik, yaitu frekuensi kata, kata kunci, frase kunci, dan kata yang termasuk dalam kelas entitas atau numerik.

Fitur rata-rata frekuensi kata (TF) ini dihitung berdasarkan frekuensi kata yang terdapat dalam satu kalimat dibandingkan dengan frekuensi kata tersebut dalam dokumen. Semakin banyak kemunculan kata tersebut, selain *stop-word*, mempunyai kemungkinan yang lebih besar untuk menjadi kata yang penting dalam dokumen, perhatikan persamaan (4).

$$f_2 = \frac{frekuensi\ kata(td)}{frekuensi\ kata(d)} \quad (4)$$

Ekstraksi frase kunci juga menentukan ringkasan yang dihasilkan. Frase kunci ini ditentukan oleh pola yang dibentuk dengan POS. Setiap kata dalam kalimat akan dicari kelas kata yang sesuai dengan kata tersebut, kemudian rangkaian kelas kata tersebut dicocokkan dengan pola yang tersedia, perhatikan persamaan (5).

$$f_3 = \frac{\#(frase\ pada\ kalimat)}{\#(frase\ pada\ dokumen)} \quad (5)$$

Biasanya kalimat yang mengandung nama entitas dan kata dalam bentuk numerik besar kemungkinan dimasukkan dalam ringkasan dokumen. Nama entitas dan numerik ini bisa didapatkan dari kelas kata yang dihasilkan dari proses POS *tagging*, perhatikan persamaan (6).

$$f_4 = \frac{\#(nama\ entitas\ S_i\ \cup numerik\ S_i)}{Panjang(S_i)} \quad (6)$$

Fitur yang terakhir dalam penelitian ini diasumsikan dalam penelitian ini adalah, kata-kata kunci yang terdapat dalam suatu surat. Jika dalam kalimat tersebut terdapat kata kunci, maka probabilitas kalimat tersebut untuk menjadi ringkasan akan semakin tinggi, perhatikan persamaan (7).

$$f_5 = \frac{\#(kata\ kunci\ S_i)}{Panjang(S_i)} \quad (7)$$

2.2. Penghitungan Probabilitas

Untuk menentukan probabilitas kelas suatu kalimat, maka dihitung terlebih dahulu probabilitas setiap fitur yang terdapat dalam kalimat. Sesuai dengan teorema Naïve Bayes maka terdapat dua kelas suatu kalimat, dua kelas tersebut adalah termasuk ringkasan dan tidak termasuk ringkasan.

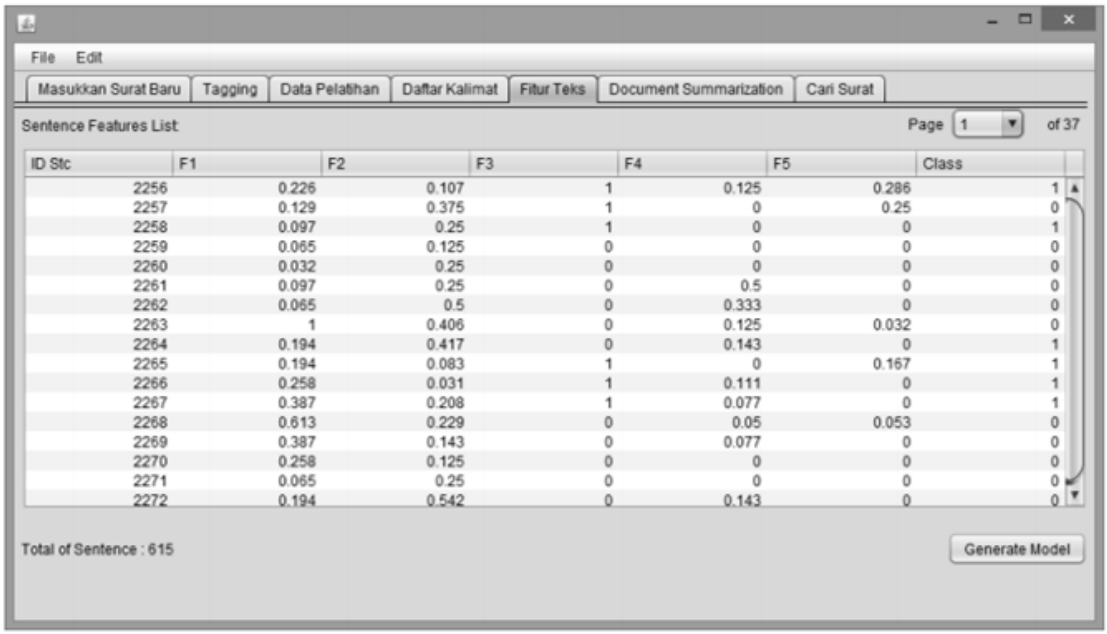
Representasi dokumen *D* dalam penelitian ini bisa dilihat pada Tabel 1. Di dalam dokumen D terdapat kumpulan kalimat, maka $D = \{S_1, S_2, S_3, S_n\}$. Setiap kalimat *S* mempunyai fitur $f_1, f_2, \dots, f_n$.

3. Hasil Penelitian dan Pembahasan

Untuk menguji sistem, dalam penelitian ini menggunakan 23 dokumen dalam bentuk surat yang diekstrak menjadi 615 kalimat, dan digunakan sebagai data pelatihan. Dokumen ini diperoleh dari arsip Organisasi Remaja Masjid Agung Jawa Tengah (Risma-JT). Untuk melakukan proses peringkasan, terlebih dahulu dihitung skor untuk masing-masing fitur. Contoh hasil penghitungan skor fitur dalam proses data pelatihan, bisa dilihat di Gambar 1.

Tabel 1 Representasi fitur-fitur dalam dokumen

Kalimat	Fitur Teks					Kelas
	f_1	f_2	f_3	....	f_n	
S_1	x_{11}	x_{12}	x_{13}	...	x_{1m}	y_0
S_2	x_{21}	x_{22}	x_{23}	...	x_{2m}	y_1
...	...	...	...	...	...	...
S_n	x_{n1}	x_{n2}	x_{n3}	...	x_{nm}	y_0



ID Stc	F1	F2	F3	F4	F5	Class
2256	0.226	0.107	1	0.125	0.286	1
2257	0.129	0.375	1	0	0.25	0
2258	0.097	0.25	1	0	0	1
2259	0.065	0.125	0	0	0	0
2260	0.032	0.25	0	0	0	0
2261	0.097	0.25	0	0.5	0	0
2262	0.065	0.5	0	0.333	0	0
2263	1	0.406	0	0.125	0.032	0
2264	0.194	0.417	0	0.143	0	1
2265	0.194	0.083	1	0	0.167	1
2266	0.258	0.031	1	0.111	0	1
2267	0.387	0.208	1	0.077	0	1
2268	0.613	0.229	0	0.05	0.053	0
2269	0.387	0.143	0	0.077	0	0
2270	0.258	0.125	0	0	0	0
2271	0.065	0.25	0	0	0	0
2272	0.194	0.542	0	0.143	0	0

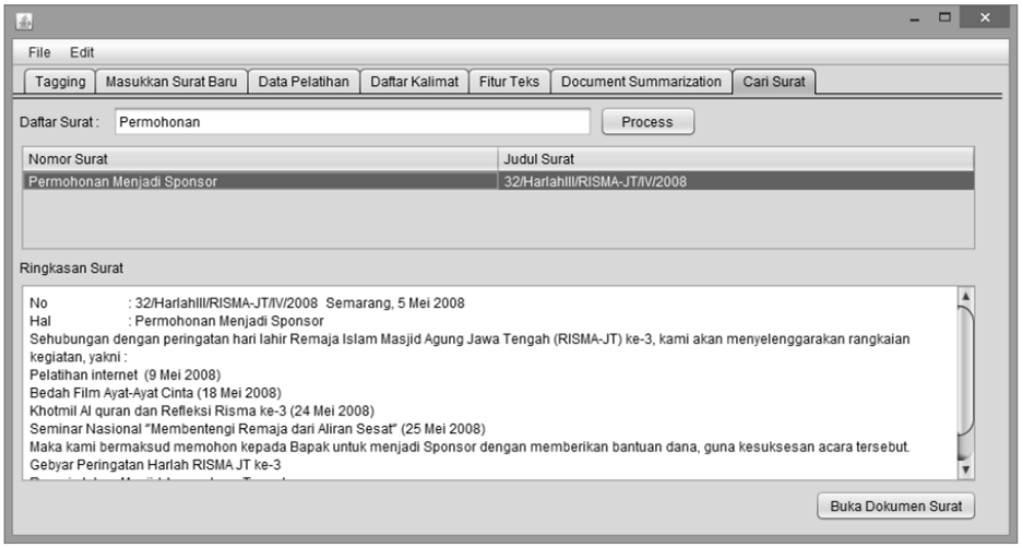
Gambar 1 Skor fitur teks dalam suatu kalimat

Untuk memudahkan pengguna dalam melakukan pencarian surat, sistem akan menampilkan ringkasan sebuah surat, sehingga memudahkan pengguna untuk memahami surat yang dicari tanpa harus membuka seluruh isi surat. Sebagaimana ditunjukkan dalam Gambar 2.

Selain itu digunakan data uji untuk mengetahui kinerja sistem peringkasan surat. Dalam pengujian ini, penelitian ini menggunakan data yang berbeda dengan data latih. Di Tabel 2 bisa dilihat hasil presentasi kompresi dokumen ringkasan dibandingkan dengan dokumen asli. Diuji juga tingkat kemudahan ringkasan untuk dipahami oleh pengguna. Hasil ringkasan yang baik mengandung informasi penting sebagaimana dokumen asli. Diasumsikan ringkasan yang baik adalah ringkasan yang mengandung nomor surat, tanggal surat, dan konten utama surat. Berdasarkan hasil uji coba, tingkat kesuksesan ringkasan dalam menyediakan informasi penting mencapai 96.67% dan rata-rata kompresi mencapai 53.67%. Hal ini memenuhi kriteria bahwa hasil ringkasan adalah tidak lebih dari setengah dari dokumen asli.

4. Kesimpulan

Otomatisasi peringkasan dokumen bisa diterapkan sebagai penunjang sistem manajemen surat menyurat dalam suatu organisasi. Hal ini tentunya memudahkan pengguna dalam mengelola data yang berbentuk surat. Berdasarkan hasil penelitian, metode Naïve Bayes juga bisa diterapkan dalam otomatisasi peringkasan dokumen.



Gambar 2 Skor fitur teks dalam suatu kalimat

Tabel 2 Representasi fitur-fitur dalam dokumen

No	Dokumen Asli		Ringkasan		Informasi Ringkasan			CR
	Judul Surat	Jumlah Kata	Jumlah Kata	Kompresi (%)	Nomor Surat	Tanggal Surat	Konten Utama	
1	Surat permohonan delegasi	182	86	52,75	√	√	√	√
2	Surat permohonan publikasi	208	92	55,77	√	√	√	√
3	Revisi surat dan formulir	202	97	51,98	√	√	√	√
4	Surat permohonan bank soal	190	88	53,68	√	√	√	√
5	Surat permohonan bantuan	180	85	52,78	√	√	√	√
6	Surat pemberitahuan kegiatan	157	75	52,23	√	√	√	√
7	Surat undangan Rapat	97	32	67,01	×	√	√	√
8	Surat permohonan	161	78	51,55	√	√	√	√
9	Surat permohonan ceramah	138	69	50,00	√	√	√	√
10	Surat Pemberitahuan Libur	153	78	49,02	√	√	√	×

5. Referensi

Aristoteles, Widiarti, & Wibowo, E. D. (2014). Text Feature Weighting for Summarization of Documents. *International Journal of Computer Science and Telecommunications, 5*(7), 29-33.

Berker, M. (2011). *Using Genetic Algorithms With Lexical Chains For Automatic Text Summarization.* Istanbul: Bogazici University.

Binwahlan, M. S. (2011). *Fuzzy Swarm Diversity Based Text Summarization.* Johor Bahru: Universiti Teknologi Malaysia.

Budhi, G. S., Intan, R., R, S., & R, S. R. (2007). Indonesian Automated Text Summarization. *Proceeding ICSIIT*.

Dehkordi, P. K., Kumarci, F., & Khosravi, H. (2009). Text Summarization Based on Genetic Programming. *International Journal of Computing and ICT Research, 3*(1), 57-64.

Erkan, G., & Radev, D. R. (2004). LexRank: Graph-based Lexical Centrality as Salience in. *Journal of Artificial Intelligence Research (JAIR), 22*(1), 457-479.

Goldberg, D. E. (1989). Genetic Algorithms In Search, Optimization, And Machine Learning.

Mihalcea, R. (2004). Graph-based Ranking Algorithms for Sentence Extraction, Applied to Text Summarization. *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions.* Stroudsburg (PA).

Mirchev, U., & Last, M. (2014). Multi-document Summarization by Extended Graph Text Representation and Importance Refinement. *ulti-document Summarization by Extended Graph Text Representation.* Hershey (PA): IGI Global.

Nandhini, K., & S, R. B. (2013). Improving readability through extractive summarization for learners with reading difficulties. *Egyptian Informatics Journal, 14*(3), 195–204.

Nandhini, K., & S, R. B. (2013). Use of G enetic Algorithm f or Cohesive Summary Extraction to. *Applied Computational Intelligence and Soft Computing, 2013*(8), 1-11.

Prasetyo, B., Uliniansyah, T., & Riandi, O. (2009). Indonesian Automated Text Summarization. *International Conference on Rural Information and Communication Technology* , 26-27.

Qazvinian, V., Hassanabadi, L. S., & Halavati, R. (2008). Summarising Text With A Genetic Algorithm-Based Sentence Extraction. *Int. J. Knowledge Management Studies, 2*(4), 426-444.