

Sistem Informasi Geografis Risiko Kemunculan *Rip Current* Menggunakan *Decision Tree C4.5*

Made Leo Radhitya^{*1}, Agus Harjoko²

¹Program Studi S2 Ilmu Komputer, FMIPA UGM, Yogyakarta

²Jurusan Ilmu Komputer dan Elektronika, FMIPA UGM, Yogyakarta

e-mail: ^{*1}leoibel@gmail.com, ²aharjoko@ugm.ac.id

Abstrak

Salah satu bahaya yang terdapat di daerah pesisir pantai adalah *rip current*. *Rip current* merupakan bahaya yang signifikan untuk para pengunjung pantai. Penelitian ini berusaha untuk menyajikan informasi geografis mengenai risiko kemunculan *rip current* dengan menggunakan *decision tree* yang dibangun menggunakan algoritma *C4.5*. Hasil yang menjadi keluaran dari *decision tree* tersebut berupa risiko kemunculan *rip current*. Studi kasus pada penelitian ini adalah pantai yang terletak di Pulau Rote, Kabupaten Rote Ndao, Provinsi Nusa Tenggara Timur. Dari hasil evaluasi kinerja sistem, didapatkan nilai rata – rata untuk *accuracy* sebesar 0,84, nilai rata – rata untuk *precision* sebesar 0,61, nilai rata – rata untuk *recall* sebesar 0,68, dan nilai rata – rata *F-measure* yang didapat sebesar 0,59 dalam rentang nilai 0 hingga 1.

Kata kunci—SIG, *Rip current*, *Decision tree*, *C4.5*

Abstract

One of the dangers that occur at the beach is *rip current*. *Rip current* poses significant danger for beachgoers. This paper proposes a method to predict the *rip current's* occurrence risk by using *decision tree* generated using *C4.5* algorithm. The output from the *decision tree* is *rip current's* occurrence risk. The case study for this research is the beach located at Rote Island, Rote Ndao, Nusa Tenggara Timur. Evaluation result shows that the *accuracy* is 0.84, and the *precision* is 0.61. The average *recall* value is 0.68 and the average *F-measure* is 0.59 in the range 0 to 1.

Keywords—GIS, *Rip current*, *Decision tree*, *C4.5*

1. PENDAHULUAN

Pariwisata di Indonesia sudah sangat terkenal sejak dulu, terutama pantai-pantai yang sangat banyak dikarenakan Indonesia merupakan negara kepulauan. Pulau Rote merupakan salah satu tujuan pariwisata yang juga sedang dikembangkan. Informasi yang disediakan tentang pariwisata di Pulau Rote masih terbilang kurang, terutama informasi tentang keamanan pada setiap pantai di pulau tersebut. Pembuatan sistem informasi geografis berbasis web diperlukan untuk mendukung pembangunan pariwisata ini.

Salah satu bahaya yang terdapat di daerah pesisir pantai adalah *rip current*. Menurut [1], *rip current* merupakan bahaya yang signifikan untuk para pengunjung pantai karena kombinasi dari arus cepat, pendalaman saluran, dan potensi untuk menghancurkan gelombang di titik keluar dari robekan. Bahaya *rip current* ini telah memakan banyak korban di penjuru dunia. Pada penelitian yang dilakukan oleh [2], korban tenggelam yang diakibatkan *rip current*

mencapai 78%, bunuh diri dan kecelakaan kapal mencapai 5%, kecelakaan terjadi di pelabuhan sekitar 10% dan sisanya tidak diketahui penyebabnya.

Terdapat beberapa penelitian tentang algoritma C4.5 yang telah dimodifikasi sehingga performanya lebih meningkat. Penelitian yang dilakukan oleh [3] memodifikasi algoritma C4.5 dengan memadukan metode *Bagging* dari Breiman dan *Boosting* dari Schapire yang merupakan metode terbaru saat itu untuk meningkatkan kekuatan prediksi dari *classifier learning system*. Penelitian selanjutnya yang dilakukan oleh [4] di tahun yang sama adalah tentang peningkatan penggunaan *continuous attribute* dalam algoritma C4.5. Ternyata modifikasi ini jauh lebih mudah, mengingat kembali bahwa kriteria *gain ratio* yang membagi *gain* yang tampak dengan informasi yang tersedia dari *split*, sehingga *tree* menjadi lebih sederhana dan lebih akurat.

Penelitian tentang prediksi *rip current* telah dilakukan oleh beberapa peneliti. Awalnya, [5] telah mencari korelasi antara arah angin, kecepatan angin, tinggi gelombang dan waktu air surut untuk penyelamatan pantai di Florida tenggara dan kemudian membuat *Lushine Rip current Scale* (LURCS). kemudian, [6] melakukan penelitian di Central Florida bagian timur dan memodifikasi LURCS berdasarkan pengamatan bahwa kondisi gelombang lebih parah pada bagian pantai Florida tersebut dan periode gelombang memiliki peranan penting dan menamakannya East-Central Florida LURCS (ECFL LURCS). Arah gelombang selanjutnya menjadi peranan penting dalam ECFL LURCS. Hal ini ditemukan oleh [7] dalam penelitiannya di Pantai Daytona dan Pantai New Smyrna di Florida. Selanjutnya, [8] meneliti penggunaan LURCS yang telah dimodifikasi sebelumnya oleh [7] dengan menambahkan 2 pantai sebagai studi kasusnya. Pada penelitian yang dilakukan oleh [8], ditemukan hubungan antara kemunculan *rip current* dan keberadaan cuaca yang diikuti oleh angin kencang dan periode gelombang yang panjang. Menurut [9], ECFL LURCS masih kurang akurat, [9] menekankan bahwa kondisi lokal juga mendikte kemunculan *rip current* di pantai, dan teknik empiris seperti LURCS dan turunannya harus dikalibrasi secara lokal. *Rip current* di India telah diteliti oleh [2] dan menghilangkan faktor angin dalam LURCS karena kurangnya statistik dan bukti ilmiah yang mengaitkan angin dan kematian yang disebabkan *rip current*. Arah penyebaran gelombang dianggap mempunyai peranan penting dan menambahkannya ke dalam LURCS.

2. METODE PENELITIAN

Sistem ini mempunyai tujuan untuk memberikan informasi keamanan pantai dalam hal risiko *rip current*. Pengambilan keputusannya menggunakan *decision tree* yang dibangkitkan dengan algoritma C4.5. Pembangunan sistem dimulai dari pengumpulan data parameter risiko kemunculan *rip current*. Data non spasial yang dikumpulkan merupakan data ketinggian gelombang, periode gelombang, arah gelombang, penyebaran gelombang, dan tahap pasang surut.

Pembangkitan *decision tree* dilakukan dengan menggunakan algoritma C4.5. Selanjutnya, *decision tree* yang telah dibangkitkan diletakkan pada *web client*. Pada proses pengambilan keputusan risiko *rip current*, data parameter yang sama diambil dan kemudian menjadi masukan pada *decision tree* yang telah dibangkitkan sebelumnya. Hasil keluaran dari *decision tree* merupakan risiko *rip current*. Studi kasus yang dipilih adalah kondisi pantai yang ada disekitar Pulau Rote. Setelah proses pengambilan keputusan selesai, sistem akan masuk ke tahap pengujian. Pengujian sistem yang dilakukan adalah data dari hasil *decision tree* dibandingkan dengan data aslinya. kemudian dilakukan perhitungan untuk mengetahui nilai *precision*, *accuracy*, *recall*, dan *F-measure* dari *decision tree* yang telah dibangkitkan menggunakan algoritma C4.5 sebelumnya.

Menurut [10], terdapat sejumlah kekurangan dalam algoritma C4.5. Pertama, C4.5 menggunakan strategi *divided*, karena itu hasil akhirnya masih kurang optimal meskipun mempunyai akurasi yang tinggi. Kedua, kedalaman pohon dan jumlah node tidak dipertimbangkan, padahal kedalaman pohon berpengaruh pada kecepatan dan jumlah node

menunjukkan ukuran dari *Decision Tree*. Ketiga, saat membangun *Decision Tree*, sangat sulit untuk mengatur ulang isi di dalam tree. Keempat, efisiensi ujinya rendah.

2.1 Analisis Kebutuhan Sistem

Kebutuhan fungsional sistem pada penelitian ini adalah sebagai berikut:

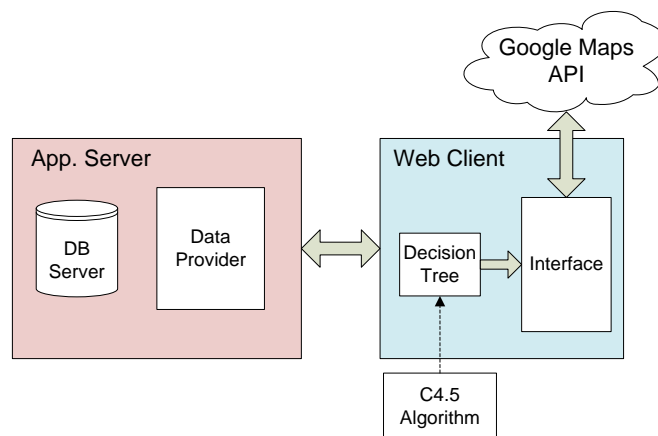
1. Sistem dapat menyusun data kondisi pantai yang berasal dari *database* ke dalam format JSON.
2. Sistem dapat memproses data parameter *rip current* dalam format JSON untuk membangkitkan *decision tree* melalui algoritma C4.5.
3. Sistem dapat melakukan *filtering* risiko *rip current* pada peta sesuai dengan keinginan pengguna.

Sistem dapat melakukan pencarian pantai dan menampilkan kondisi serta risiko *rip current* yang terjadi di pantai tersebut. Dokumen masukan yang diperlukan oleh sistem dalam tahap pembangkitan adalah data parameter risiko *rip current* selama 3 tahun. Data parameter risiko *rip current* yang dimaksud adalah: tinggi gelombang laut, periode gelombang laut, arah gelombang laut, arah penyebaran, pasang surut laut, dan risiko *rip current* yang didapat dari lapangan.

Pada proses pengambilan keputusan risiko *rip current*, dokumen masukan berupa data non spasial parameter risiko *rip current* dalam format JSON. Selanjutnya, data tersebut menjadi masukan *decision tree* yang telah dibangkitkan sebelumnya. Data spasialnya berupa peta yang didapat dari Google Maps dan juga kordinat pantai untuk *layering* pada peta.

2.2 Arsitektur Sistem

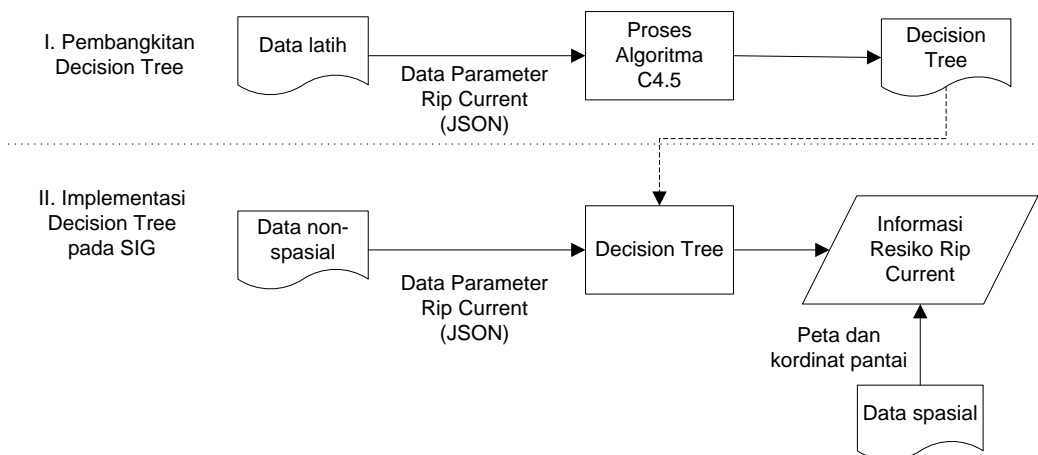
Pada arsitektur sistem, terdapat aplikasi di sisi server yang bertugas melayani permintaan data dari *web client*. Dalam *web client* terdapat data non-spasial yang diterima akan diproses oleh *decision tree* sebelum menuju *interface*. *Decision tree* yang ada di dalam *web client* didapat dari hasil pembangkitan oleh algoritma C4.5. Ilustrasi arsitektur sistem dapat dilihat pada Gambar 1.



Gambar 1 Arsitektur sistem secara umum

Gambaran umum dalam sistem ini dimulai dari tahap pembangkitan *decision tree*. Data parameter risiko *rip current* dalam format JSON menjadi masukan untuk proses pembangkitan *decision tree* menggunakan algoritma C4.5. Kemudian hasil keluaran yang berupa *decision tree* tersebut ditanam pada sistem informasi geografis risiko *rip current* yang berinteraksi langsung dengan pengguna. Gambaran umum proses sistem dapat dilihat pada Gambar 2.

Pada tahap implementasi *decision tree* ke dalam SIG, data parameter *rip current* beserta kordinat pantai yang telah disusun dalam format JSON diambil dan menjadi masukan untuk *decision tree* yang telah dibangkitkan dan ditanamkan. Data tersebut merupakan data non-spasial dari SIG.



Gambar 2 Gambaran umum proses keseluruhan sistem

Pada tahap implementasi *decision tree* ke dalam SIG, data parameter *rip current* beserta kordinat pantai yang telah disusun dalam format JSON diambil dan menjadi masukan untuk *decision tree* yang telah dibangkitkan dan ditanamkan. Data tersebut merupakan data non-spasial dari SIG. Peta beserta *polyline* yang merupakan data spasial dari SIG selanjutnya diambil. Data parameter *rip current* yang telah menjadi masukan *decision tree* akan diproses di dalamnya sehingga keluaran dari proses tersebut merupakan informasi risiko *rip current* yang akan ditampilkan pada antarmuka sistem. Dalam kurun waktu tertentu, evaluasi sistem dapat dilakukan dengan melakukan *update* pada data parameter *rip current* dan melakukan pembangkitan *decision tree* kembali berdasarkan data awal ditambah dengan data parameter terbaru.

2.3 Rancangan Proses Pembangkitan Decision Tree

Proses pembangkitan *decision tree* menggunakan algoritma C4.5 dimulai dengan mengumpulkan kasus-kasus berdasarkan rekam data yang tersedia. Setelah data terkumpul, jumlah kasus yang terjadi untuk setiap atribut yang ada mulai dihitung kemudian dituangkan ke dalam tabel. Contoh tabel kasus dapat dilihat pada Tabel 1.

Setelah tabel kasus didapat, perancangan pembangkitan *decision tree* menggunakan algoritma C4.5 dilakukan. Algoritma C4.5 dirancang secara rekursif. Awalnya, dilakukan penghitungan untuk mencari nilai entropi pada seluruh dan setiap kasus. Kemudian mulai menghitung nilai *gain* dan dilakukan perhitungan *split*. Setelah itu, nilai *gain ratio* pada setiap kategori (atribut) dicari. Nilai *gain ratio* yang didapat dari setiap atribut kemudian dibandingkan. Nilai terbesar akan menjadi dasar kasus untuk perhitungan selanjutnya. Dasar kasus yang kasusnya hanya terdapat pada satu kategori risiko *rip current* langsung menjadi *leaf node* atau keputusan. Selain itu, dilakukan kembali perhitungan yang sama seperti awal.

Pada persamaan (1), nilai *gain* dari setiap atribut $gain(C_j)$ didapat dari perhitungan entropi informasi total kasus ($info(T)$) dikurangi dengan entropi informasi setiap atribut yang bersangkutan ($info(T, C_j)$). Penghitungan entropi informasi dapat dilihat pada persamaan (2).

Penghitungan entropi informasi setiap atribut ($info(C_j, T)$) dapat dilihat pada persamaan (3). Setelah melakukan perhitungan-perhitungan tersebut, maka dilakukan perhitungan nilai *split*. Persamaan (4) menunjukkan penghitungan untuk mencari nilai *split*. Nilai *split* yang didapat akan menjadi masukan untuk mendapatkan rasio gain yang ditunjukkan pada persamaan (5). *Gain ratio* pada masing-masing atribut yang didapat akan menjadi perbandingan untuk mendapatkan dasar kasus untuk perhitungan selanjutnya. Rancangan diagram alir untuk proses pembangkitan *decision tree* menggunakan algoritma C4.5 dapat dilihat pada Gambar 3. Perhitungan ini akan terus berjalan hingga semuanya mendapatkan keluaran *leaf node*.

Tabel 1 Contoh tabel kasus

		<i>Rip current Risk</i>				
		TOTAL KASUS	Very high	High	Moderate	Low
TOTAL KASUS		1092	17	452	605	18
Wave Height(m)						
	WH > 2.0	96	14	68	14	0
	1.5 < WH <= 2.0	171	3	104	64	0
	1.0 < WH <= 1.5	252	0	131	119	2
	0.5 < WH <= 1.0	439	0	129	297	13
	WH <= 0.5	134	0	20	111	3
Wave Period(s)						
	WP > 12	0	0	0	0	0
	9 < WP <= 12	50	5	42	3	0
	6 < WP <= 9	338	12	184	141	1
	4 < WP <= 6	449	0	196	244	9
	WP <= 4	255	0	30	217	8
Wave Direction(deg)						
	-15 <= WD <= 5	272	0	115	157	0
	(-25 <= WD < -15) or (5 < WD <= 10)	370	0	168	199	3
	(-30 <= WD < -25) or (10 < WD <= 15)	187	0	67	119	1
	(-35 <= WD < -30) or (15 < WD <= 20)	197	0	53	130	14
	WD < -35 or WD > 20	66	17	49	0	0
Directional spreading(deg)						
	DS < 10	0	0	0	0	0
	10 <= DS < 20	327	0	155	172	0
	20 <= DS < 30	509	7	167	320	15
	30 <= DS < 40	239	10	121	105	3
	DS >= 40	17	0	9	8	0
Tidal Stage (m)						
	TS <= 0.2	674	12	341	318	3
	TS > 0.2	418	5	111	287	15

Pada Gambar 3, proses dimulai dari mengumpulkan kasus-kasus kemunculan *rip current*. Dari kasus yang dikumpulkan tersebut, dihitung nilai entropi untuk total kasus dan setiap atributnya, kemudian menghitung nilai *gain*, *split*, sehingga didapat nilai *gain ratio* untuk setiap atribut. Setelah setiap atribut memiliki *gain ratio*, maka dicari atribut dengan nilai *gain ratio* terbesar. Atribut dengan nilai *gain ratio* terbesar akan menjadi *node tree*. Kemudian, atribut dengan *gain ratio* terbesar dilihat setiap sub atributnya. Jika ada sub atribut yang kasus. Jika tidak, maka proses diulang lagi tetapi dengan kasus yang berdasarkan sub atribut dari atribut dengan *gain ratio* terbesar. kemunculan *rip current*-nya hanya pada 1 kategori risiko

kemunculan saja, maka pada sub atribut tersebut memiliki keputusan pada kategori risiko tersebut. Jika ada sub atribut yang masih terdapat lebih dari 1 kategori, maka proses diulang lagi tetapi dengan kasus yang berdasarkan sub atribut dari atribut dengan gain ratio terbesar. Proses tersebut akan terus berulang hingga setiap node memiliki keputusan. Rumus untuk $gain(C_j)$

$$gain(C_j) = info(T) - info(T, C_j) \quad (1)$$

dimana rumus $info(T)$

$$info(T) = - \sum_{j=1}^{N_{Class}} \frac{freq(C_j, T)}{|T|} \times \log\left(\frac{freq(C_j, T)}{|T|}\right) \quad (2)$$

dan untuk rumus $info(T, C_j)$

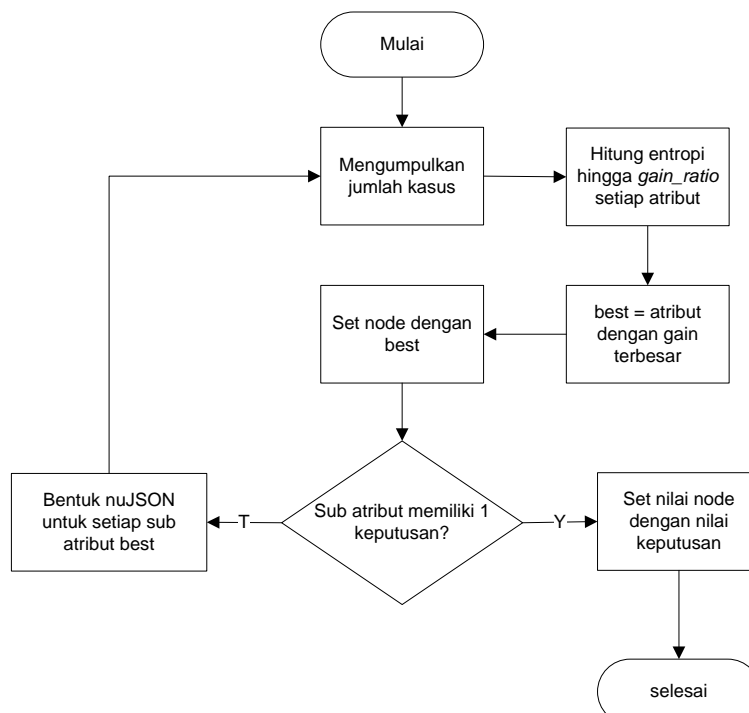
$$info(T, C_j) = \sum_{i=1}^s \frac{|T_i|}{|T|} info(T_i) \quad (3)$$

kemudian dilakukan perhitungan $split(C_j)$

$$split(C_j) = - \sum_{i=1}^s \frac{|T_i|}{|T|} \log \frac{|T_i|}{|T|} \quad (4)$$

untuk mencari rasio dari gain,

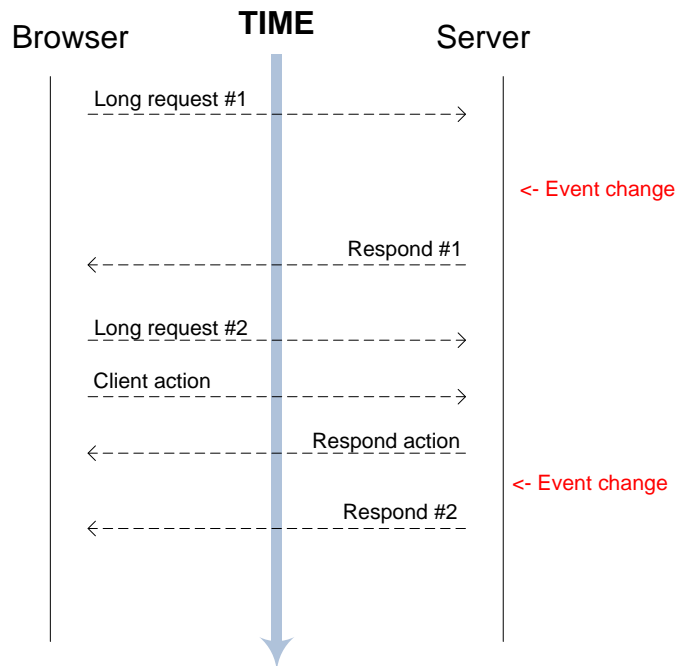
$$gain_ratio(C_j) = \frac{gain(C_j)}{split(C_j)} \quad (5)$$



Gambar 3 Diagram alir untuk pembangkitan decision tree

2.4 Rancangan SIG

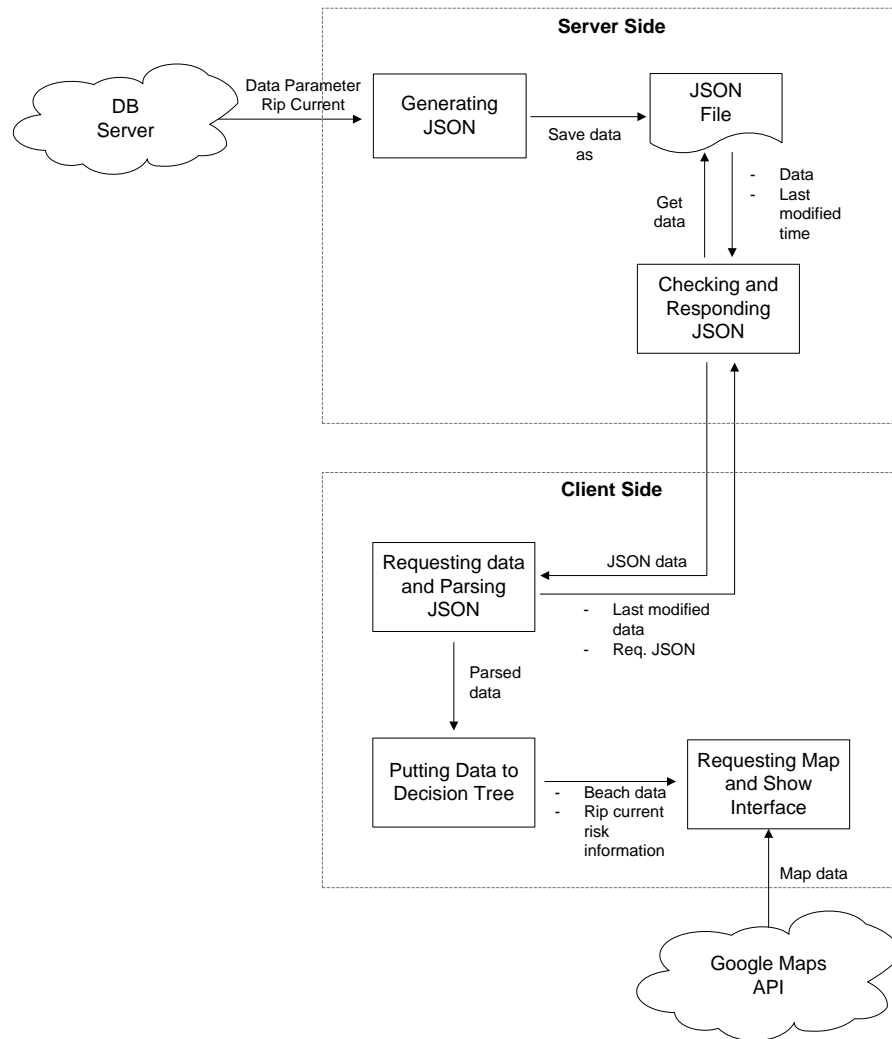
Pada rancangan SIG, dilakukan pendekatan konsep *reverse ajax* dengan metode *long polling*. Tujuan penggunaan pendekatan ini adalah melakukan *update* pada *interface web* ketika terdapat perubahan atau *update data* pada server dalam jeda waktu yang mendekati *real-time*. Metode *long polling* adalah *ajax* yang melakukan *request* dan *respond* seperti *ajax* lainnya. Hal yang membedakan *long polling* dengan metode *ajax* lainnya adalah *respond* yang diberikan dari server ke *client* berdasarkan perubahan data yang terjadi yang artinya, server tidak akan mengirimkan *respond* sebelum ada perubahan data. Model ilustrasi *long polling* dapat dilihat pada Gambar 4.



Gambar 4 Model ilustrasi *long polling*

Rancangan SIG dimulai dengan pembangunan proses *long polling* dari sisi server dan *client*, memasang *decision tree* yang telah dibangkitkan pada proses pembangkitan yang menggunakan algoritma C4.5 pada sisi *client*, dan mengambil peta dari server. Data non-spasial untuk SIG berupa data tinggi gelombang laut, periode gelombang laut, arah gelombang laut, arah penyebaran dan pasang surut laut. Data-data tersebut disimpan didalam database yang kemudian diambil dan direpresentasikan dalam bentuk dokumen JSON. Dokumen ini kemudian masuk ke proses pengambilan keputusan yang ada di dalam *decision tree* yang telah dibangun sebelumnya. Sedangkan untuk data kordinat pantai akan menjadi representasi dari polyline garis-garis pantai pada peta. Hasil keluarannya berupa risiko *rip current* yang sesuai dengan kondisi tersebut dan ditampilkan dilayar beserta peta lokasi pantai tersebut.

Pada sisi *client*, terdapat proses yang melakukan *request* data spasial ke Google API. Data tersebut akan disesuaikan dengan data pantai dan risiko *rip current*. Desain proses tampilan dapat ditunjukkan pada Gambar 5.



Gambar 5 Desain proses SIG

Saat data pantai dan risiko *rip current* sudah diterima, dilakukan konfigurasi untuk pemetaan. Kemudian, peta yang telah dikonfigurasi dimunculkan ke layar. Pengguna sistem dapat melakukan interaksi seperti mencari lokasi pantai dan peta yang ada pada layar akan langsung menunjukkan pantai yang dicari beserta informasi dari pantai tersebut. Selain melakukan pencarian, pengguna juga bisa melakukan *filtering* risiko *rip current*. Peta akan memperlihatkan pantai mana saja yang mempunyai risiko *rip current* sesuai dengan *filtering* yang dilakukan sebelumnya.

2.5 Rancangan Evaluasi Pengujian

Evaluasi pengujian dilakukan dengan cara menguji informasi *rip current* yang diprediksi oleh sistem terhadap informasi *rip current* yang terjadi di lapangan. Pengujian tersebut meliputi pencarian nilai *Accuracy*, *Precision*, *Recall*, dan *F-measure* pada sistem. Terdapat beberapa modifikasi untuk menyesuaikan jumlah risiko *rip current* yang kategorinya terbagi menjadi 4. Persamaan tersebut dapat dilihat pada persamaan (6), (7), (8). Persamaan untuk menghitung *accuracy* menjadi:

$$\text{Accuracy} = \frac{1}{4} \sum_{i=1}^4 \frac{TP_i + TN_i}{Total_i} \quad (6)$$

Persamaan untuk mencari nilai *precision* dan *recall* juga mengalami modifikasi menjadi:

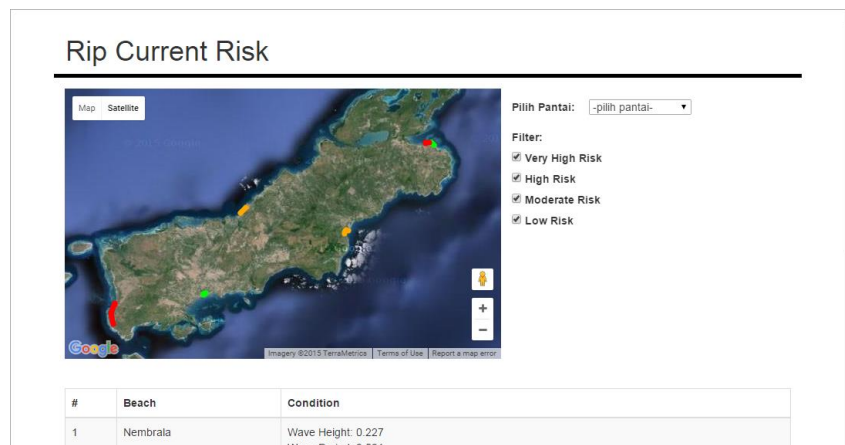
$$\text{Precision} = \frac{1}{4} \sum_{i=1}^4 \frac{TP_i}{PP_i} \quad (7)$$

$$\text{Recall} = \frac{1}{4} \sum_{i=1}^4 \frac{TP_i}{TP_i + FN_i} \quad (8)$$

3. HASIL DAN PEMBAHASAN

3.1 Tampilan sistem

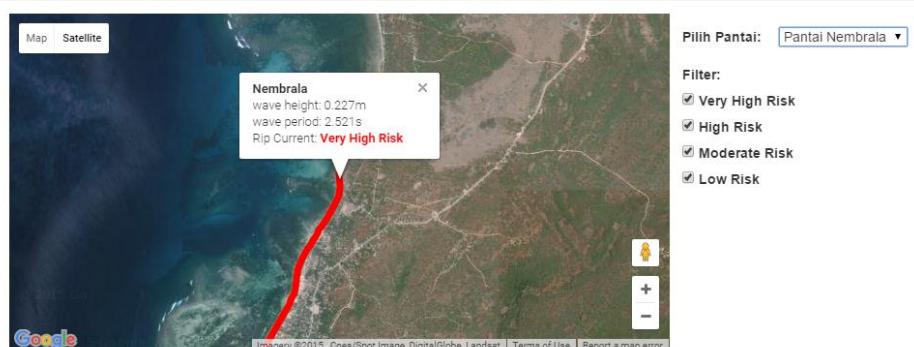
Terdapat beberapa fitur dalam sistem yang telah dibangun. Gambar 6 memperlihatkan tampilan sistem saat pertama kali dibuka.



Gambar 6 Halaman sistem ketika pertama kali dibuka

Pada Gambar 6, dapat dilihat terdapat peta pada sisi kiri. Sisi kanan dari tampilan tersebut berisi input untuk *filtering* dan pencarian pantai. Pada peta, garis-garis pantai diberi *polyline* berwarna untuk menunjukkan posisi pantai. Warna merah menunjukkan risiko yang sangat tinggi (*very high risk*), oranye untuk risiko tinggi (*high risk*), hijau untuk risiko sedang (*moderate risk*), dan biru untuk risiko rendah (*low risk*). Gambar 7 menunjukkan input pantai yang telah dipilih.

Rip Current Risk



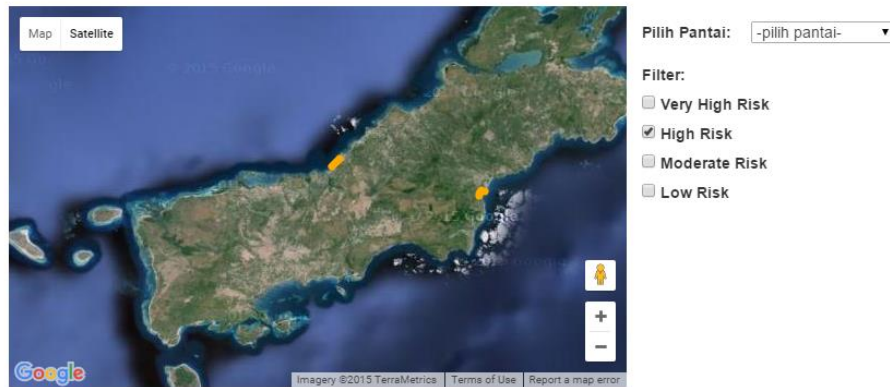
Gambar 7 Contoh menggunakan pencarian pantai

Pencarian pantai dapat dilakukan dengan memilih daftar pantai yang terdapat pada *dropdown box* dengan label “Pilih Pantai”. Ketika input pantai dipilih, peta akan langsung mengarah ke pantai tersebut dan memunculkan informasi dari pantai tersebut.

Proses *filtering* pantai dapat dilihat pada Gambar 8. Sistem akan memunculkan pantai mana saja yang sesuai dengan risiko yang dipilih. *Filtering* pada pantai dilakukan dengan memberikan centang pada input. Pantai lainnya yang tidak dicentang, tidak akan dimunculkan

pada peta. Pada Gambar 8, risiko yang dipilih adalah *high risk*, maka pantai-pantai yang memiliki risiko tersebut yang akan dimunculkan pada peta.

Rip Current Risk



Gambar 8 Contoh penggunaan fitur *filtering* risiko *rip current*

3.2 Hasil pengujian *accuracy*, *precision*, *recall*, dan *F-measure*

Pengujian dilakukan dengan membuat tabel confusion matrix untuk masing - masing risiko *rip current*. Kemudian mulai dilakukan penghitungan *accuracy*, *precision*, *recall*, dan *F-measure*. Nilai *accuracy* dari masing – masing kategori dan rata – rata keseluruhan kategori ditunjukkan pada Tabel 2.

Tabel 2 Nilai *accuracy* dari hasil *decision tree* C4.5

Risiko	Accuracy
Very high	0,69
High	0,71
Moderate	0,95
Low	0,99
Rata-rata	0,84

Nilai *precision* dari masing – masing kategori dan rata – rata keseluruhan kategori ditunjukkan pada Tabel 3.

Tabel 3 Nilai *precision* dari hasil *decision tree* C4.5

Risiko	Precision
Very high	0,070
High	0,98
Moderate	0,89
Low	0,5
Rata-rata	0,61

Nilai *recall* dari masing – masing kategori dan rata – rata keseluruhan kategori ditunjukkan pada Tabel 4.

Tabel 4 Nilai *recall* dari hasil *decision tree* C4.5

Risiko	Recall
Very high	0,53
High	0,59
Moderate	0,91
Low	0,75
Rata-rata	0,69

Setelah mendapatkan nilai *accuracy*, *precision*, dan *recall*, nilai *precision* dan *recall* dapat digunakan secara bersama – sama dengan menggabungkannya dalam *F-measure*. Nilai *F-measure* menunjukkan rata – rata harmonis antara *precision* dan *recall*.

Nilai *F-measure* dari masing – masing kategori dan rata – rata keseluruhan kategori ditunjukkan pada Tabel 5.

Tabel 5 Nilai *F-measure* pada masing – masing kategori

Risiko	F-measure
Very high	0,12
High	0,73
Moderate	0,90
Low	0,6
Rata-rata	0,59

Dari Tabel 5 dapat dilihat nilai F-measure pada kategori *very high* merupakan nilai yang terendah daripada kategori lainnya. Hal ini disebabkan oleh nilai *precision* pada kategori *very high* relatif kecil. Namun, nilai rata – rata *accuracy* hasil dari *decision tree* C4.5 memiliki nilai yang cukup tinggi yaitu 0,84 dari rentang nilai 0 hingga 1. Semakin besar nilai *accuracy* tersebut, maka sistem semakin akurat.

3.3 Pengujian waktu proses

3.2.1 Waktu proses pembangkitan *decision tree*

Pengujian waktu proses pembangkitan *decision tree* menggunakan algoritma C4.5 dilakukan secara *offline*. Dari eksperimen didapat waktu proses pembangkitan *decision tree* menggunakan algoritma C4.5 dari awal hingga menghasilkan *decision tree* memerlukan waktu 253ms dengan data kasus yang diproses sebanyak 1092 kasus.

3.2.1 Waktu proses memuat *website SIG*

Pengujian waktu proses memuat *website* dilakukan secara *online*. Layanan *web server* yang digunakan adalah *Google App Engine*. Waktu proses memuat *website* yang telah ditanamkan *decision tree* C4.5, dari pemanggilan *website* hingga penampilan informasi risiko kemunculan *rip current*, memerlukan waktu 6,04s. Waktu proses ini bisa berubah-ubah karena tergantung dengan koneksi internet yang tersedia.

4. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa:

1. Sistem Informasi Geografis Risiko Kemunculan *Rip current* dapat dibangun dengan menggunakan *decision tree* yang dibangkitkan dengan menggunakan algoritma C4.5 berdasarkan data parameter ketinggian gelombang laut, periode gelombang laut, arah gelombang laut, arah penyebaran, dan pasang surut laut untuk mendapatkan informasi risiko kemunculan *rip current*.
2. Dengan data parameter yang digunakan untuk membangkitkan *decision tree* menggunakan algoritma C4.5 dan menguji *decision tree* tersebut, didapatkan nilai rata – rata untuk *accuracy* sebesar 0,84, nilai rata – rata untuk *precision* sebesar 0,61, nilai rata – rata untuk *recall* sebesar 0,69, dan nilai rata – rata *F-measure* yang didapat sebesar 0,59. Nilai – nilai tersebut pada kategori *very high* masih cenderung kecil, yang artinya

keakuratan decision tree C4.5 masih kurang akurat. Hal ini disebabkan oleh kurangnya kasus kemunculan *rip current* pada kategori *very high*.

3. Waktu proses pembangkitan *decision tree* menggunakan algoritma C4.5 mulai dari menjalankan program hingga menampilkan *decision tree* diproses dalam waktu 253ms dengan data parameter yang digunakan sebanyak 1092 kasus. Sedangkan, waktu proses memuat *website* mulai dari pemanggilan *website* hingga penampilan informasi secara relatif diproses dalam waktu 6,04s.

5. SARAN

Dari penelitian yang dilakukan, adapun saran untuk perbaikan penelitian lebih lanjut antara lain:

1. Sistem Informasi Geografis Pemantauan Risiko Kemunculan *Rip current* pada penelitian ini perlu ditingkatkan lagi nilai *precision* dan *recall*-nya. Peningkatan nilai – nilai tersebut berhubungan dengan data parameter yang digunakan agar lebih disesuaikan lagi. Jumlah kasus juga perlu ditambah agar akurasi sistem meningkat.
2. Kemunculan *rip current* juga dipengaruhi oleh kondisi geografi sekitar pantai dimana sistem akan memerlukan kombinasi dengan metode lain yang berhubungan dengan struktur tanah dan kondisi geografi lainnya.

DAFTAR PUSTAKA

- [1] Short, A.D., dan Hogan, C.L., 1994, Rip Currents and Beach Hazards: Their Impact on Public Safety and Implications for Coastal Management, *J Coast Res S112*, 197–209.
- [2] Kumar, A., dan Prasad, 2014, Rip current-related fatalities in India: a new predictive risk scale for forecasting rip currents, *Nat Hazards* 70, 313–335
- [3] Quinlan, J.R., 2006, Bagging, Boosting, and C4.5, <http://www.cs.ecu.edu/~dingq/CSCI6905/readings/BaggingBoosting.pdf>, diakses tanggal 24 Desember 2012, 10:38
- [4] Quinlan, J.R., 2006, Improved Use of Continuous Attributes in C4.5, <http://arxiv.org/pdf/cs/9603103.pdf>, diakses tanggal 24 Desember 2012, 10:36.
- [5] Lushine, J., 1991 Rip Currents: Human Impact and Forecastability. In: Proceedings of the coastal zone, *ASCE*, New York, 3558–3569.
- [6] Lascody, R., 1998, East Central Florida Rip Current Program, *National Weather Dig* 22, 25–30.
- [7] Engle, J., 2003, Formulation of A Rip Current Forecasting Technique Through Statistical Analysis of Rip Currentrelated Rescues, *Tesis*, University of Florida, Gainesville.
- [8] Schrader, M., 2004, Evaluation of The Modified ECFL LURCS Rip Current Forecasting Scale and Conditions of Selected Rip Current Events in Florida, *Tesis*, University of Florida, Gainesville.
- [9] Nelko, V. dan Dalrymple R.A., 2008, Rip Currents: Mechanisms and Observations, *Smith JM (Ed) Proceedings Of The 31st International Conference On Coastal Engineering*, Singapore, September 5.
- [10] Rui, L., Xian-mei, W., dan Xue-wei, Y., 2009, The Improvement of C4.5 Algorithm and Case Study, *IEEE*, p.190-192, <http://origin-www.computer.org/plugins/dl/pdf/proceedings/iscid/2009/3865/02/3865b190.pdf>, diakses tanggal 01 Desember 2012, 15.24