Komparasi Decision Tree dan Random Forest untuk Klasifikasi Kapal pada Citra Satelit di Wilayah Maritim

Nama Kelompok : The Return of Sultan

Anggota Kelompok

1. Sultan Muzahidin (1806169591)
2. Fauhan Handay Pugar (1806255355)
3. Rif'at Ahdi Ramadhani (1806269543)

Deskripsi proyek

Proyek ini akan membahas mengenai komparasi dua algoritma yaitu *decision tree* dan *random forest* pada klasifikasi antara kapal dan bukan kapal pada citra satelit. Sebagai representasi dari wilayah maritim kami menggunakan dataset yang diperoleh dari *Planet satellite imagery* yang berada pada area San Fransisco Bay dan San Pedro Bay sekitar di California. Dataset terdiri dari 4000 citra chip RGB, dimana setiap citra berukuran 80x80 piksel. Citra chip yang didapat dari visual frame *PlanetScope* secara penuh yang ter-ortoreksi ke ukuran piksel dengan jarak 3 meter. Data citra yang digunakan adalah png dimana penamaan gambarnya memiliki format khusus yaitu {label}_{scene_id}_{longitude}_{latitude}.png. Dataset ini juga menyediakan format teks JSON dengan nama shipsnet.json yang terdiri dari data, label, scene_ids dan daftar lokasi. Label terdiri nilai 1 dan 0, angka 1 merepresentasikan kelas "kapal" dan "bukan kapal". Kelas "kapal" (1000 citra) terdiri dari banyak bentuk kapal dengan berbagai ukuran dan bentuk. Sedangkan kelas "bukan kapal" (3000 citra) dimana sepertiganya adalah *random sampling* dari fitur tutupan lahan (*landcover features*) seperti air, vegetasi , tanah kosong, bangunan, dll. Sepertiga berikutnya adalah "kapal parsial" yang hanya berisi sebagaian kapal sehingga tidak memenuhi bagian kapal secara penuh. Sepertiga terakhir adalah gambar yang salah label oleh pembelajaran mesin, biasanya disebabkan oleh piksel yang cerah atau lain-lainnya. *Scene_id* adalah pengenal unik dari *PlanetScope* untuk setiap visual dari citra chip yang di ekstrak. *Longitude_latitude* merupakan koordinat dari citra pada titik tengah gambar dimana setiap nilainya dipisah dengan underscore. Setiap nilai piksel dari citra RGB berukuran 80x80 disimpan dalam bentuk list yang terdiri dari nilai integer 19200. Data pertama terdiri dari 6400 nilai pada channel R, selanjutnya 6400 nilai pada channel G, dan terakhir 6400 nilai pada channel B. Dataset tersedia pada situs Kaggle – *Ship in Satellite Imagery* [1].

Motivasi

Indonesia adalah negara kepulauan terbesar didunia dimana indonesia memiliki 17.499 pulau dari sabang sampai merauke. Sebagai negara kepulauan, Indonesia memiliki wilayah laut lebih luas dari daratan karena itu Indonesia di sebut sebagai negara maritim. Total luas wilayah Indonesia adalah 7,81 juta km^2 yang terdiri dari 3,25 juta km^2 lautan, 2,01 juta km^2 daratan dan 2,55 juta km^2 Zona Ekonomi Eksklusif (ZEE) [2]. Wilayah laut Indonesia memiliki potensi sumber kekayaan sangat besar yaitu sebagai pemasok ikan terbesar didunia. Potensi ekonomi sumber daya kelautan dan perikanan yang dapat dimanfaatkan untuk mendorong pertumbuhan ekonomi diperkirakan mencapai USD 82 miliar per tahun [3]. Dengan banyaknya potensi yang dimiliki Indonesia memerlukan perlindungan dan pengelolaan sumber daya perairan yang baik.

Salah satu permasalahan yang dihadapi negara maritim seperti Indonesia adalah praktik *illegal fishing*. *Illegal fishing* merupakan aktivitas pencurian ikan yang dilakukan oleh kapal asing yang melewati wilayah yurisdiksi suatu negara secara ilegal. Praktik ini jelas telah sangat merugikan negara setiap tahunnya. Menurut Menteri Kelautan dan Perikanan, Susi Pudjiastuti, kerugian negara telah mencapai Rp 240 triliun [4]. Selain itu, praktik *illegal fishing* juga menyebabkan kerugian lainnya, yakni kerusakan ekosistem laut.

Mengacu pada Undang-Undang Nomor 31 tahun 2004 dan Undang-Undang Nomor 45 Tahun 2009, Pemerintah Indonesia telah melakukan kebijakan penanganan terhadap praktik *illegal fishing* dengan cara menerapkan kebijakan penenggelaman kapal

yang melakukan tindak pidana tersebut. Hingga tahun 2018, sebanyak 488 kapal illegal fishing telah ditenggelamkan dalam penerapan kebijakan tersebut [5]. Untuk menerapkan kebijakan tersebut, tentunya diperlukan pengawasan secara intensif. Namun pengawasan secara intensif masih memiliki tantangan besar dalam hal usaha dan biaya yang diperlukan.

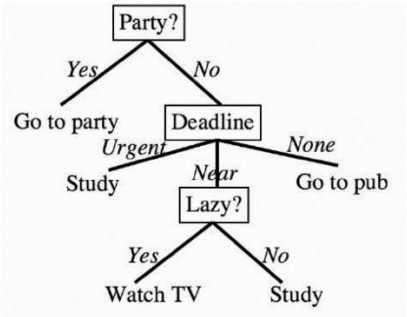
Peran *machine learning* diharapkan mampu memberikan solusi yang efisien dalam masalah pengawasan wilayah laut di Indonesia. Klasifikasi berperan untuk mengetahui pemetaan sebaran jumlah kapal di area tertentu. Identifikasi ini diperlukan agar pengawasan menjadi lebih terarah dan menjadi pondasi yang membantu dalam mengindentifikasi kapal yang terlibat dalam aktifitas *illegal fishing*. Oleh karena itu, pada penilitian ini kami membuat kerangka sistem machine learning untuk mengklasifikasikan objek kapal dan bukan kapal yang berada di suatu wilayah tertentu.

Metode

Metode yang kami gunakan adalah *decision tree* dan *random forest* pada klasifikasi kapal. Berikut adalah penjelasan mengenai kedua metode tersebut:

Decision Tree

Algoritma decision tree merupakan salah satu algoritma pembelajaran mesin yang memiliki struktur data dan performa komputasi yang baik. Secara umum, kebutuhan komputasi yang diperlukan dalam membuat struktur tree cukup rendah dan kebutuhan komputasi untuk keperluan klasifikasi yang dapat dinyatakan dengan notasi big-O $O(\log N)$. Hal ini menjadi poin penting dalam pembelajaran mesin, terutama dalam aspek kebutuhan pengolahan dan penanganan data dalam jumlah besar dan kebutuhan pemerolehan hasil yang cepat. Selain itu, decision tree juga memiliki keunggulan lain yaitu kemudahan untuk memahami secara langsung dan transparan alur logika yang dimodelkan pada decision tree jika dibandingkan dengan metode 'black box' seperti neural network. Hal inilah yang menjadi alasan klasifikasi dengan menggunakan decision tree menjadi cukup popular dalam pembelajaran mesin [6].



Gambar 1 Skema Decision Tree [6]

Ada beberapa jenis algoritma decision tree, namun semua varian yang ada memiliki prinsip yang sama yaitu proses pembuatan tree dilakukan secara rekursif dimulai dari bagian root dengan memilih fitur yang paling informatif berdasarkan nilai entropi tertentu. Nilai entropi dapat dihitung dengan persamaan:

$$Entropy(p) = -\sum_i p_i log_2 p_i \quad (1)$$

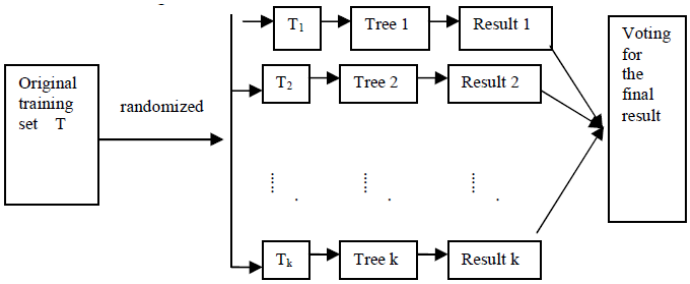
Random Forest

Random forest adalah kombinasi dari algoritma *machine learning.* Dimana kita mengkombinasikan serangkaian *tree classifier*, pada setiap *tree* dilakukan *voting* untuk mendapatkan kelas paling populer. Kemudian hasil dari kombinasi tersebut dilakukan *sorting*. *Random forest* memiliki akurasi yang tinggi, tahan terhadap noise dan juga tidak pernah mendapatkan *overfitting*.

Pada Breiman's RF model, setiap *tree* dilakukan *training* menggunakan *random variable*, dimana variabel acak pada *tree* di notasikan dengan Θ_k, antara dua variabel acak saling *independent and identically distributed*. Hasil klasifikasi h(x, Θ_k) dimana x adalah masukan sebuah vektor. Setelah k dijalankan kita akan mendapatkan urutan klasifikasi { $h_1(x), h_2(x),\ldots h_k(x)$}, dan kemudian setelah mendapatkan model sistem yang lebih dari satu klasifikasi, hasil akhirnya dilakukan majority vote, fungsi dari pemilihannya adalah sebagai berikut :

$$H(x) = \arg\max_{Y} \sum_{i=1}^{k} I\left(h_i(x) = Y\right) \quad (2)$$

H(x) adalah kombinasi klasifikasi model, h_i adalah *single decision tree model*, Y adalah *output variable*, I adalah *indicator function*. Gambar jelasnya sebagai berikut [7]:



Gambar 2 Skema Random Forest

Hasil Eksperimen Sementara

Berdasarkan dari hasil penelitian yang dibuat, diperoleh akurasi, presisi, recall, F1-score dari masing-masing algoritma pembelajaran mesin yaitu decision tree dan random forest. Hasil uji coba yang dilakukan menggunakan kedua adalah sebagai berikut:

a. Decision Tree

   Diperoleh nilai akurasi pada data training dan data testing sebesar 1 dan 0,8425 dengan metrik perfoma seperti pada Tabel *2*.

b. Random Forest

   Diperoleh nilai akurasi pada data training dan data testing sebesar 1 dan 0,8575 dengan metrik perfoma seperti pada Tabel *3*.

Berdasarkan hasil yang diperoleh, kedua algoritma tersebut menunjukkan overfitting. Untuk mencegah hal tersebut kami menggunakan stratified k-fold cross validation dengan nilai k=5 dan k=10. Stratified k-fold digunakan untuk menyeimbangkan proposi data. Hal ini dikarenakan jumlah data antara kapal dan bukan kapal tidak seimbang yaitu 1000 dan 3000 dengan nilai prevelance data kapal adalah 25%. Hasil yang diperoleh adalah sebagai berikut:

a. Decision Tree dengan stratified k-fold (k=5)

   Hasil penelitian yang diperoleh menggunakan algoritma decision tree dengan stratified k-fold (k=5) diperlihatkan pada Tabel *4*. Rata-rata F-Score pada kelas 0 (bukan kapal) diperoleh sebesar 0,8986035543414882, rata-rata F-Score pada kelas 1 (kapal) diperoleh sebesar 0,69630214782103, dan rata-rata weighted_avg_f1_score sebesar 0,69630214782103.

b. Decision Tree dengan stratified k-fold (k=10)

Hasil penelitian yang diperoleh menggunakan algoritma decision tree dengan stratified k-fold (k=5) diperlihatkan pada Tabel *5*. Rata-rata F-Score pada kelas 0 (bukan kapal) diperoleh sebesar 0,9016775091881228, rata-rata F-Score pada kelas 1 (kapal) diperoleh sebesar 0,7037341337885067, dan rata-rata weighted_avg_f1_score sebesar 0,7037341337885067.

c. Random Forest dengan stratified k-fold (k=5)

   Hasil penelitian yang diperoleh menggunakan algoritma decision tree dengan stratified k-fold (k=5) diperlihatkan pada Tabel *6*. Rata-rata F-Score pada kelas 0 (bukan kapal) diperoleh sebesar 0,9471207075183564, rata-rata F-Score pada kelas 1 (kapal) diperoleh sebesar 0,8245840849812766, dan rata-rata weighted_avg_f1_score sebesar 0,8245840849812766.

d. Random Forest dengan stratified k-fold (k=10)

   Hasil penelitian yang diperoleh menggunakan algoritma decision tree dengan stratified k-fold (k=5) diperlihatkan pada Tabel *7*. Rata-rata F-Score pada kelas 0 (bukan kapal) diperoleh sebesar 0,9498340986735213, rata-rata F-Score pada kelas 1 (kapal) diperoleh sebesar 0,8342505009427372, dan rata-rata weighted_avg_f1_score sebesar 0,8342505009427372.

Hasil yang diperoleh menunjukkan overfitting pada kedua algoritma tersebut karena rata-rata akurasi dari data testing lebih kecil daripada akurasi dari data training. Dengan nilai akurasi pada data training adalah 1 untuk semua fold. Selain itu, berdasarkan nilai presisi dapat disimpulkan bahwa model masih belum mampu mengenali kapal dengan baik dengan rata-rata presisi decision tree dan random forest adalah 0,73 dan 0,71.

Eksperimen Selanjutnya

Setelah memperoleh hasil penelitian, eksperimen selanjutnya yang akan kami lakukan adalah untuk mengatasi masalah overfitting. Kami akan melakukan optimasi parameter dan augmentasi fitur untuk masalah ini. Optimasi parameter yang dilakukan adalah hyperparameter dari algoritma, hyperparameter dari fitur dasar yaitu HOG. Augmentasi fitur yang dilakukan adalah rotasi, scaling, flipping, noise addition. Kami juga akan memodelkan proses deteksi objek kapal pada citra satelit.

Kontribusi

Pada progress 2 ini kami mengimplementasikan algoritma decision tree, random forest, stratified k-fold dengan dua parameter k yaitu 5 dan 10 pada kedua algoritma tersebut yaitu decision tree dan random forest. Masing-masing algoritma diukur perfomansinya menggunakan akurasi, presisi, recall, dan F1-score. Adapun pekerjaan spesifik yang dilakukan setiap anggota adalah sebagai berikut:

1. Sultan Muzahidin : Decision tree dengan evaluasi perfoma
2. Fauhan Handay Pugar : Random forest dengan evaluasi perfoma
3. Rif’at Ahdi Ramadhani : Stratified k-fold pada decision tree dan random forest (k=5 dan 10) dengan evaluasi perfoma

Referensi

[1] Kaggle, “Ships in Satellite Imagery,” 2018. [Online]. Available: https://www.kaggle.com/rhammell/ships-in-satellite-imagery.

[2] B. P. S. INDONESIA, “STATISTIK SUMBER DAYA LAUT DAN PESISIR.” 2018.

[3] S. K. R. INDONESIA, “Potensi Besar Perikanan Tangkap Indonesia.” 2016.

[4] Detik, “Menteri Susi: Kerugian Akibat Illegal Fishing Rp 240 Triliun.” 2014.

[5] Katadata, “Cek Data: Benarkah 488 Kapal Illegal Fishing Sudah Ditenggelamkan?” 2019.

[6] S. Marsland, *Machine Learning: An Algorithmic Perspective, Second Edition*, 2nd ed. Chapman & Hall/CRC, 2014.

[7] Y. Liu, Y. Wang, and J. Zhang, “New Machine Learning Algorithm: Random Forest,” 2012, pp. 246–252.

[8] Y. Wang, X. Zhu, and B. Wu, “Automatic detection of individual oil palm trees from UAV images using HOG features and an SVM classifier,” vol. 1161, 2018.

[9] H. Zhou, Y. Zhuang, L. Chen, and H. Shi, “Ship Detection in Optical Satellite Images,” vol. 3, 2018.

Lampiran

Tabel

Tabel 1. Lini waktu penelitian

Rencana	Februari 2019	Maret 2019					April					Mei	
	5	1	2	3	4	5	1	2	3	4	5	1	2
Proposal													
Analisis Dataset													
Pre-processing													
Ekstraksi Fitur													
Progress 1													
Decision Tree													
Random Forest													
Progress 2													
Optimasi dan Evaluasi													
Pembuatan Laporan													

Akhir dan Poster													
Pengumpulan													

Keterangan :

Warna abu-abu: Aktifitas yang telah dilakukan

Warna Hijau: Akftitas yang akan dilakukan

Tabel 2. Metrik perfoma algorima decision tree

	Precision	Recall	F-Score	support
0	0.88	0.9	0.89	581
1	0.73	0.68	0.7	219
micro avg	0.84	0.84	0.84	800
macro avg	0.8	0.79	0.8	800
weighted avg	0.84	0.84	0.84	800

Tabel 3. Metrik perfoma algorima random forest

	Precision	Recall	F-Score	support
0	0.9	0.91	0.91	605
1	0.71	0.7	0.7	195
micro avg	0.86	0.6	0.86	800
macro avg	0.81	0.8	0.81	800
weighted avg	0.86	0.86	0.86	800

Tabel 4. Metrik perfoma algorima decision tree dengan k-fold (k=5)

Fold-	Accuracy Training	Accuracy Testing		Precision	Recall	F-Score	support
1	1	0.83375	0	0.89	0.88	0.89	600
			1	0.66	0.69	0.67	200
			micro avg	0.83	0.83	0.83	800
			macro avg	0.78	0.78	0.78	800
			weighted avg	0.84	0.83	0.83	800
2	1	0.84625	0	0.89	0.8	0.9	600
			1	0.7	0.68	0.69	200
			micro avg	0.85	0.85	0.85	800
			macro avg	0.8	0.79	0.79	800
			weighted avg	0.85	0.85	0.85	800
3	1	0.8725	0	0.92	0.91	0.91	600
			1	0.74	0.76	0.75	200
			micro avg	0.87	0.87	0.87	800
			macro avg	0.83	0.83	0.83	800
			weighted avg	0.87	0.87	0.87	800
4	1	0.85375	0	0.89	0.92	0.9	600
			1	0.73	0.67	0.69	200
			micro avg	0.85	0.85	0.85	800
			macro avg	0.81	0.79	0.8	800
			weighted avg	0.85	0.85	0.85	800

5	1	0.83375	0	0.9	0.88	0.89	600
			1	0.66	0.69	0.68	200
			micro avg	0.83	0.83	0.83	800
			macro avg	0.78	0.79	0.78	800
			weighted avg	0.84	0.83	0.84	800

Tabel 5. Metrik perfoma algorima decision tree dengan k-fold (k=10)

Fold-	Accuracy Training	Accuracy Testing		Precision	Recall	F-Score	support
1	1	0.8675	0	0.89	0.94	0.91	300
			1	0.78	0.65	0.71	100
			micro avg	0.87	0.87	0.87	400
			macro avg	0.84	0.79	0.81	400
			weighted avg	0.86	0.87	0.86	400
2	1	0.8525	0	0.91	0.89	0.9	300
			1	0.7	0.73	0.71	100
			micro avg	0.85	0.85	0.85	400
			macro avg	0.8	0.81	0.81	400
			weighted avg	0.86	0.85	0.85	400
3	1	0.835	0	0.9	0.88	0.89	300
			1	0.65	0.7	0.68	100
			micro avg	0.83	0.83	0.83	400
			macro avg	0.78	0.79	0.78	400
			weighted avg	0.84	0.83	0.83	400
4	1	0.86	0	0.9	0.92	0.91	300
			1	0.74	0.68	0.71	100
			micro avg	0.86	0.86	0.86	400
			macro avg	0.82	0.8	0.81	400
			weighted avg	0.86	0.86	0.86	400
5	1	0.8425	0	0.92	0.86	0.89	300
			1	0.66	0.78	0.71	100
			micro avg	0.84	0.84	0.84	400
			macro avg	0.79	0.82	0.8	400
			weighted avg	0.86	0.84	0.85	400
6	1	0.8775	0	0.92	0.92	0.92	300
			1	0.76	0.75	0.75	100
			micro avg	0.88	0.88	0.88	400
			macro avg	0.84	0.83	0.84	400
			weighted avg	0.88	0.88	0.88	400
7	1	0.85	0	0.88	0.93	0.9	300
			1	0.74	0.61	0.67	100
			micro avg	0.85	0.85	0.85	400
			macro avg	0.81	0.77	0.79	400
			weighted avg	0.84	0.85	0.84	400
8	1	0.865	0	0.91	0.91	0.91	300
			1	0.73	0.74	0.73	100
			micro avg	0.86	0.86	0.86	400
			macro avg	0.82	0.82	0.82	400
			weighted avg	0.87	0.86	0.87	400
9	1	0.8575	0	0.91	0.89	0.9	300

			1	0.7	0.75	0.72	100
			micro avg	0.86	0.86	0.86	400
			macro avg	0.81	0.82	0.81	400
			weighted avg	0.86	0.86	0.86	400
10	1	0.82	0	0.88	0.88	0.88	300
			1	0.64	0.63	0.64	100
			micro avg	0.82	0.82	0.82	400
			macro avg	0.76	0.76	0.76	400
			weighted avg	0.82	0.82	0.82	400

Tabel 6. Metrik perfoma algorima random forest dengan k-fold (k=5)

Fold-	Accuracy Training	Accuracy Testing		Precision	Recall	F-Score	support
1	1	0.92	0	0.92	0.98	0.95	600
			1	0.91	0.75	0.82	200
			micro avg	0.92	0.92	0.92	800
			macro avg	0.92	0.86	0.89	800
			weighted avg	0.92	0.92	0.92	800
2	1	0.92375	0	0.93	0.97	0.95	600
			1	0.91	0.78	0.84	200
			micro avg	0.92	0.92	0.92	800
			macro avg	0.92	0.87	0.89	800
			weighted avg	0.92	0.92	0.92	800
3	1	0.92	0	0.94	0.96	0.95	600
			1	0.87	0.81	0.83	200
			micro avg	0.92	0.92	0.92	800
			macro avg	0.9	0.88	0.89	800
			weighted avg	0.92	0.92	0.92	800
4	1	0.9275	0	0.92	0.98	0.95	600
			1	0.94	0.76	0.84	200
			micro avg	0.93	0.93	0.93	800
			macro avg	0.93	0.87	0.9	800
			weighted avg	0.93	0.93	0.92	800
5	1	0.9025	0	0.91	0.96	0.94	600
			1	0.86	0.73	0.79	200
			micro avg	0.9	0.9	0.9	800
			macro avg	0.89	0.84	0.86	800
			weighted avg	0.9	0.9	0.9	800

Tabel 7. Metrik perfoma algorima random forest dengan k-fold (k=10)

Fold-	Accuracy Training	Accuracy Testing		Precision	Recall	F-Score	support
1	1	0.93	0	0.93	0.89	0.95	300
			1	0.94	0.77	0.85	100
			micro avg	0.93	0.93	0.93	400
			macro avg	0.93	0.88	0.9	400
			weighted avg	0.93	0.93	0.93	400
2	1	0.9125	0	0.92	0.97	0.94	300
			1	0.88	0.75	0.81	100
			micro avg	0.91	0.91	0.91	400
			macro avg	0.9	0.86	0.88	400

			weighted avg	0.91	0.91	0.91	400
3	1	0.9225	0	0.93	0.97	0.95	300
			1	0.91	0.77	0.83	100
			micro avg	0.92	0.92	0.92	400
			macro avg	0.92	0.87	0.89	400
			weighted avg	0.92	0.92	0.92	400
4	1	0.935	0	0.94	0.98	0.96	300
			1	0.92	0.81	0.86	100
			micro avg	0.94	0.94	0.94	400
			macro avg	0.93	0.89	0.91	400
			weighted avg	0.93	0.94	0.93	400
5	1	0.9225	0	0.94	0.95	0.95	300
			1	0.86	0.83	0.84	100
			micro avg	0.92	0.92	0.92	400
			macro avg	0.9	0.89	0.9	400
			weighted avg	0.92	0.92	0.92	400
6	1	0.9325	0	0.94	0.97	0.96	300
			1	0.91	0.81	0.86	100
			micro avg	0.93	0.93	0.93	400
			macro avg	0.92	0.89	0.91	400
			weighted avg	0.93	0.93	0.93	400
7	1	0.925	0	0.92	0.99	0.95	300
			1	0.96	0.73	0.83	100
			micro avg	0.93	0.93	0.93	400
			macro avg	0.94	0.86	0.89	400
			weighted avg	0.93	0.83	0.92	400
8	1	0.935	0	0.94	0.98	0.96	300
			1	0.93	0.8	0.86	100
			micro avg	0.94	0.94	0.94	400
			macro avg	0.93	0.89	0.91	400
			weighted avg	0.93	0.94	0.93	400
9	1	0.9075	0	0.93	0.95	0.94	300
			1	0.85	0.77	0.81	100
			micro avg	0.91	0.91	0.91	400
			macro avg	0.89	0.86	0.87	400
			weighted avg	0.91	0.91	0.91	400
10	1	0.9075	0	0.91	0.97	0.94	300
			1	0.89	0.72	0.8	100
			micro avg	0.91	0.91	0.91	400
			macro avg	0.9	0.84	0.87	400
			weighted avg	0.91	0.91	0.9	400