Implementasi Algoritma *Merging Context Seeds* untuk *Plagiarism Detection*

Proposal Tugas Akhir

Kelas TA 1

Yusuf Anugrah Putra Aditama
1103120030

Program Studi Sarjana Informatika
Fakultas Informatika
Universitas Telkom
Bandung
2016

Lembar Persetujuan

Implementasi Algoritma *Merging Context Seeds* untuk *Plagiarism Detection*

Merging Context Seeds Implementation for Plagiarism Detection

Yusuf Anugrah Putra Aditama
NIM : 1103120030

Proposal ini diajukan sebagai usulan pembuatan tugas akhir pada
Program Studi S1 Teknik Informatika
Fakultas Informatika
Universitas Telkom

Bandung, 29 Maret 2016

Menyetujui,

Pembimbing 1,	Pembimbing 2,
Ir. Moch. Arif Bijaksana, Ph.D	Mohamad Syahrul Mubarok, S.T.
NIP. 03650312-4	NIP. 10830757-3

Daftar Isi

Abstrak

Plagiat merupakan tindakan berupa mengklaim suatu ide ataupun gagasan orang lain sebagai hasil karya atau miliknya sendiri. Tindak plagiat ini sangat merugikan terutama bagi penulis sesungguhnya baik di dunia pendidikan maupun di dunia seni.
Maka dari itu pada tugas akhir ini akan membahas mengenai pendeteksian plagiat pada suatu dokumen, dengan informasi awal berupa pasangan dokumen plagiat dan dokumen sumber. Dokumen yang ada akan diolah dengan metode utama yaitu *Merging Context Seed*. Pada akhirnya keluaran dari program ini adalah jenis tindak plagiat dan bagian yang terindikasi plagiat. Nantinya perfomansi sistem yang dibuat akan dihitung menggunakan *Precision, Recall* dan *F-Measure*. Diharapkan sistem yang dibuat mampu mengklasifikasi dokumen yang ada dengan akurasi yang tinggi.

Kata kunci: Plagiat, Merging Context Seed, Seeds, Source Retrieval

Abstract

Plagiarism is an act that claim an idea other people as their own. This plagiarism act is doing harm especially for original writer, either in education and art subject.
Therefore on this last project will be discussed about detecting plagiarism on document, by information from pair plagiarism document with origin document. Those document will be treated with main method : Merging Context Seed. At the end of the output of the program is, type of plagiarism and part of the document that was traced. Performance of system will be calculated with Precision, Recall, and F-Measure. The expectation from the system is, can afford classify document with high accuracy.

Keyword(s): Plagiarism, Merging Context Seed, Seeds, Source Retrieval

Bab 1. Pendahuluan

1.1. Latar Belakang

Plagiarism, Plagiat atau Penjiplakan merupakan tindakan berupa mengkalim suatu ide, pendapat maupun karangan orang lain sebagai hasil karya sendiri [1]. Tindak plagiat ini merupakan masalah yang sangat serius, dan sering ditemukan pada bidang sastra dan pendidikan.

Tindak plagiat ini juga sangat merugikan baik pihak penjiplak dan pihak yang dijiplak. Merugikan pihak penjiplak karena dapat berujung pada sanksi pidana dan secara tidak langsung membatasi kreatifitas si penjiplak itu sendiri. Dan merugikan pihak yang dijiplak karena usaha atau hasil karya yang ia hasilkan digunakan orang dengan seenaknya tanpa memberikan *credits* kepada pengarang aslinya.

Tindakan plagiat ini juga terdapat beberapa karakteristik, yaitu :

1. Mengklaim ide / gagasan orang lain miliknya sendiri.
2. Menggunakan tulisan orang lain pada sebagian / seluruh karyanya secara utuh.
3. Menggunakan tulisan orang lain pada sebagian / seluruh karyanya dengan menambahkan atau menghilangkan beberapa kata pada tulisannya.
4. Menggunakan tulisan orang lain pada sebagian / seluruh karyanya dengan mengubah tataan kalimatnya namun mempunyai makna yang sama.

Dengan karakteristik diatas, suatu tindak plagiat dapat diketahui dengan menggunakan pendekatan *Text Alignment* pada kasus *Plagiarism Detection* atau Deteksi Plagiat sesuai dengan *task* yang ada di PAN[2]. Hal ini dapat dilakukan karena dengan pendekatan *Text Alignment* dapat diketahui apakah suatu dokumen menjiplak dokumen lain berdasarkan *pairs* dari proses *Source Retrieval*, hal ini dengan membaca pola kalimat beserta kata-kata yang ada pada dua buah dokumen. Dan dari pendekatan *Text Alignment* terdapat metode yang bernama *Merging Context Seed* yang akan dibahas pada Tugas Akhir ini.

Diharapkan dengan metode yang diusulkan ini dapat dibangun sistem yang dapat mendeteksi tindak plagiat dengan akurat berdasarkan info *pairs* dari

proses *Source Retrieval*. Pada sistem yang dibangun juga akan digunakan *N-Gram* sebagai metode bantuan untuk mengekstraksi ciri dari dua dokumen yang ada.

1.2. Rumusan Masalah

Masalah yang dihadapi disini adalah bagaimana mendeteksi tindak plagiat pada suatu dokumen. Untuk menyelesaikan masalah tersebut pada dasarnya memerlukan 2 *task* utama yaitu *Source Retrieval* dan *Text Alignment*.

Namun pada tugas akhir ini hanya akan membahas proses *Text Alignment* untuk mendeteksi dua dokumen yang terindikasi plagiat. Dua dokumen itu adalah, dokumen yang mencurigakan, dan dokumen sumber. Dan untuk menentukan dua dokumen tadi dilakukan pada tahap *Source Retrieval* yang pada tugas akhir ini tidak dilakukan, dikarenakan *pairs* atau data berupa pasangan dokumen yang terindikasi dan sumbernya sudah diberikan sebelumnya.

Dan untuk menyelesaikan masalah diatas digunakan metode *Merging Context Seed* untuk memeriksa kebenaran indikasi plagiat dari *pairs* yang ada.

1.2.1 *Input*

Input pada kasus ini adalah :

1. Kumpulan dokumen yang terindikasi plagiat.
2. Kumpulan dokumen yang menjadi sumber plagiat dokumen diatas.
3. *Pairs*, yang berisikan informasi pasangan dokumen yang terindikasi plagiat beserta dokumen sumbernya.

1.2.2 *Output*

Output yang akan dihasilkan dari tugas akhir ini adalah klasifikasi berdasarkan *Pairs*. Kelas klasifikasi yang ada adalah sebagai berikut :

1. ***No-Plagiarism***
   Pasangan dokumen tidak tedardapat tindak plagiat sama sekali.
2. ***No-Obfuscation***
   Dokumen terindikasi melakukan tindak plagiat berupa *copy-paste*, yaitu mengunakan kalimat sumber secara utuh tanpa melakukan perubahan apapun.
3. ***Random-Obfuscation***
   Pada dokumen yang terindikasi terdapat tindak plagit berupa penghapusan ataupun penambahan kata pada kalimat yang sumber.

4. ***Summary-Obfuscation***
   Terdapat tindak plagiat berupa peringkasan kalimat sumber.

Selain kelas klasifikasi, pada tugas akhir ini juga akan menampilkan bagian pada dokumen yang terindikasi tindak plagiat.

1.3. Tujuan

1. Menggunakan *N-Gram* untuk ekstraksi ciri dari 2 buah dokumen.
2. Mendeteksi tindak plagiat dari *pairs* dengan metode *Merging Context Seed*.
3. Menentukan kelas tindak plagiat seusai jenisnya.

1.4. Batasan Masalah

Adapun yang menjadi batasan masalah dalam penelitian Tugas Akhir ini adalah sebagai berikut :

1. *Dataset* yang digunakan merupakan dokumen berbahasa inggris.
2. *Dataset* yang digunakan merupakan dokumen *plain text*.
3. Dokumen yang terindikasi melakukan tindak plagiat dibagi menjadi 4 kelas, yaitu :
   (a) *No-Plagiarism*
   (b) *No-Obfuscation*
   (c) *Random-Obfuscation*
   (d) *Summary-Obfuscation*

1.5. Hipotesa

Dengan pengimplementasian metode *Merging Context Seed* dapat mendeteksi tindak plagiat dari *pairs* yang ada dengan akurasi yang tinggi. Dan sistem mampu membuktikan apakah suatu dokumen menjiplak dokumen lain atau tidak dengan metode yang digunakan.

1.6. Rencana Kegiatan

Adapun metedeologi yang digunakan untuk memecahkan masalah yang ada yaitu sebagai berikut :

1. Studi Literatur
   Pada tahap ini dikumpulkan data dan informasi segala metode yang dibutuhkan untuk menyelesaikan masalah yang ada. Studi literatur yang digunakan diantaranya adalah sebagai berikut :

(a) *Plagiarism*.

(b) *Text Alignment*.

(c) *Merging Context Seed*.

(d) *N-Gram*.

(e) *K-Nearest Neighbor*.

2. Analisis Kebutuhan Sistem
   Dilakukan analisis terhadap kebutuhan sistem untuk mencapai tujuan pada tugas akhir ini.

3. Perancangan Sistem
   Merancang alur sistem yang akan dibangun untuk tugas akhir ini, dimulai dari *input* berupa *pairs* dan dokumen terindikasi berikut sumbernya kemudian implementasi metode hingga mengeluarkan *output*.

4. Implementasi
   Mengimplementasikan metode yang dipelajari kedalam sistem, disini segala proses mulai dari *reprocessing*, ekstraksi ciri menggunakan *N-Gram*, perhitungan menggunakan *Merging Context Seed* dan klasifikasi menggunakan *K-Nearest Neighbor*.

5. Pengujian Sistem
   Melakukan uji coba dengan menjalankan sistem yang telah dibuat dan melakukan analisis sementara menggunakan *f-measure*.

6. Analisis
   Menganalisis hasil *output* yang dikeluarkan dari sistem. Menghitung akurasi dengan *f-measure*.

7. Pembuatan Laporan
   Pembuatan laporan mengenai kegiatan dan sistem yang di bangun yang meliputi latar belakang, rumusan masalah, tujuan, implementasi sistem hingga hasil analisis yang dilakukan selama pengerjaan tugas akhir.

1.7. Jadwal Kegiatan

Tabel menunjukan *timeline* sesuai dengan rencana kegiatan pada bab sebelumnya.

Tabel 1.1: Jadwal kegiatan pengerjaan

Kegiatan	Bulan							
	Feb	Mar	Apr	Mei	Jun	Jul	Agu	Sep
Identifikasi masalah								
Studi literatur								
Analisis kebutuhan sistem								
Perancangan sistem								
Implementasi sistem								
Pengujian sistem								
Analisis hasil								
Pembuatan laporan								

Bab 2. Tinjauan Pustaka

2.1. *Plagiarism*

Plagiarism merupakan tindakan mengklaim suatu ide, gagasan ataupun tulisan orang lain sebagai miliknya sendiri. Gagasan atau tulisan yang di klaim dapat berupa jurnal, buku, ucapan ataupun hasil diskusi.

Tindak *plagiarism* ini dapat berupa menghilangkan atau menambahkan satu atau beberapa kata dari tulisan asli seorang penulis dan digunakan untuk karya / tulisan sendiri[3]. Contohnya adalah :

Teks asli : *Dengan menggunakan metode* ***k-Nearest Neighbor*** *kita dapat mengetahui derajat ketetangaan suatu node dengan node lainnya.*
Teks Plagiat : *Dengan metode* ***k-Nearest Neighbor*** *kita dapat mengetahui derajat ketetangaan suatu node dengan node lain disekitarnya.*

Selain itu tindak *plagiarism* dapat berupa *paraphrase*. Yaitu mengubah tataan suatu kalimat menjadi bentuk lain, namun masih memiliki makna yang sama. Contohnya adalah :

Teks asli : *Dengan menggunakan metode* ***k-Nearest Neighbor*** *kita dapat mengetahui derajat ketetangaan suatu node dengan node lainnya.*
Teks Plagiat : *Untuk menghitung derajat ketetanggaan suatu node dapat menggunakan metode* ***k-Nearest Neighbor****.*

Tetapi walaupun suatu teks diubah dengan *paraphrase* tindak *plagiarism* masih dapat di indetifikasi karena ada kemiripan makna antara dua buah kalimat.

2.2. *Plagiarism Detection*

Plagiarism Detection merupakan solusi yang ditawarkan untuk menangani kasus *plagiarism*. *Plagiarism Detection* ini dapat mengidentifikasi tindak plagiat dengan beberapa pendekatan. Salah satunya adalah *Text Alignment*. *Plagiarism Detection* ini terbagi menjadi 2 *task* yaitu ***Source Retrieval*** dan ***Text Alignment***[2].

2.2.1 *Source Retrieval*

Source Retrieval Merupakan *task* awal untuk *Plagiarism Detection*. Pada tahap ini suatu dokumen akan diuji dengan cara melakukan pencarian perkalimat dari dokumen yang diuji. Kaliamt yang diuji akan dimasukan kedalam *query* mesin pencarian yang berisikan jurnal atau dokumen sejenis. Kemudian apabila ada kemiripan akan dibuat file yang berisikan informasi pasangan dokumen terindikasi dengan dokumen sumber[4].

2.2.2 *Text Alignment*

Text Alignment merupakan pendekatan yang dapat menguji kebenaran hasil pemasangan dari tahap sebelumnya. Pada tahap ini hasil dari proses *Source Retrieval* akan diuji kebenarannya, apakah dokumen tersebut terbukti memplagiat dokumen sumber atau tidak dengan cara mengekstrasi ciri yang ada pada dokumen yang terindikasi dan dokumen sumber yang kemudian diolah dengan metode tertentu.

2.3. *N-Gram*

N-Gram merupakan suatu metode yang digunakan untuk memotong-motong suatu kata menjadi beberapa bagian. Bagian yang dipotong dimulai dari panjang karaketernya sebanyak 1 hingga *k*. Sebagai contoh, kita menggunakan kata "Plagiat", maka *N-Gram* yang didapatkan adalah sebagai berikut :

Unigram : P, L, A, G, I, A, T

Bigram : _P, PL, LA, AG, GI, IA, AT, T_

Trigram : _PL, PLA, LAG, AGI, GIA, IAT, AT_, T__

Quad : _PLA, PLAG, LAGI, AGIA, GIAT, IAT_, T___

Karakter "_" melambangkan spasi didepan dan akhir kata.

2.3.1 *Learning*

Setelah proses diatas dilakukan, maka bagian yang dipecah akan di masukan ke proses *learning* yang tahapnnya adalah :

1. Fitur atau bagian yang dipotong di ubah ke betuk *N-Gram* dengan n=1,2,3,4 dan seterusnya.
2. Memasukan tiap-tiap *N-Gram* yang didapatkan ke *hash table*.
3. Jumlah *counter* akan ditambah apabila ditemukan pola yang sama pada ekstraksi ciri pada kata lainnya. Gambar 2.1 menunjukan contoh *table hash* dari ciri diatas.

N-Gram	***Counter***
PL	1
LA	1
AG	1
..	..

Tabel 2.1: Contoh Tabel Hash

4. Setelah dihitung urutkan *N-Gram* secara *descending*.

2.3.2 *Testing*

Testing pada *N-Gram* biasanya digunakan untuk menentukan kategori suatu dokumen berdasarkan kemiripan dengan dokumen lain yang sudah dikategorisasikan. Hal ini dilakukan dengan cara mengukur jarak dengan mekanisme *out-of-place measure*[5].

2.4. *Merging Context Seed*

Merging Context Seed adalah salah satu metode yang ditawarkan oleh pendekatan *Text Alignment*. Pada metode ini juga diterapkan tahapan yang umum digunakan oleh pendekatan *Text Alignment* yaitu[6, 7, 8] :

1. *Seed generation* : Terdapat dokumen yang terindikasi beranama dokumen X dan dokumen sumbernya bernama dokumen Y. Kemudian membagi kedua dokumen menjadi bagian-bagian kecil yang dapat diukur.
   Pada tahap ini *N-Gram* digunakan. Selain itu, pada tahap ini dokumen X akan diubah selhuruhnya ke huruf kecil dan menghilangkan *tab*, penghentian kata, dan seluruh karakter yang tidak termasuk *alphanumeric*. Dan dari ciri yang ada, dipilih ciri yang mempunyai makna. Sedangkan untuk dokumen Y, cirinya diesktrak berdasarkan ciri yang sudah diekstrak dari dokumen X.

2. *Merging* : Menggabungkan 2 bagian X dan Y yang mempunyai kemiripan. Tahap ini dilakukan hingga seluruh *case* selesai di *merge*[9].
   Penggabungan bagian X dan Y juga mempunyai karakteristik tertentu dan akan terus mencari irisan antara ciri dokumen X dan dokumen Y hingga tidak ada pasangan / *pairs* yang memiliki nilai kurang dari 0 atau konstanta yang ditentukan.

3. *Extraction* dan *Filtering* : Pada tahap ini, setiap bagian atau *passages* yang panjangnya kurang dari 15 kata akan dihilangkan. Dan kemudian tahap berikutnya adalah mengklasifikasikan hasil proses *merging* ke beberapa kelas yang ada.

Dan penyelesaian masalah yang akan diselesaikan dengan algoritma ini adalah bagaimana mendapatkan nilai *pladget(S,R)* yang tinggi, dimana *S merupakan kumpulan kasus plagiat* dan *R merupakan kumpulan deteksi yang dilakukan.*

2.5. *K-Nearest Neighbor*

K-Nearest Neighbor merupakan algoritma yang biasanya digunakan untuk mengklasifikasi suatu data berdasarkan pola yang ada. Algoritma ini mengklasifikasikan data berdasarkan label yang paling mirip dengan data yang ada di *training set*. Pada umumnya algoritma ini menggunakan *Euclidean Distance* untuk mengukur tingkat kesamaan data[10].
Algoritma ini dapat diadaptasikan ke berbagai masalah yang ada, sehingga *classifier* ini dipilih untuk pengerjaan tugas akhir ini.

2.6. *Precision, Recall, F-Measure*

Precision, Recall, F-Measure merupakan metode pengukuran akurasi dari suatu data. Metode ini menggunakan tabel kebenaran[11]. Tabel 2.2 menunjukan sebuah tabel kebenaran yang akan digunakan.

Tabel 2.2: Tabel Kebenaran

	Correct	**Not Correct**
Selected	True Positive (TP)	False Positive (FP)
Not Selected	False Negative (FN)	True Negative (TN)

Sebagai contoh, pada kasus pengklasifikasian ke golongan A, nilai yang ada adalah sebagai berikut :

1. **True Positive** : Suatu data diklasifikasikan oleh sistem sebagai golongan A, dan tujuannya data tersebut memang seharusnya digolongkan ke golongan A.

2. **False Positive** : Suatu data diklasifikasikan oleh sistem sebagai golongan A, dan tujuannya data tersebut *tidak* seharusnya digolongkan ke golongan A.

3. **False Negative** : Suatu data diklasifikan oleh sistem sebagai **bukan** golongan A, tapi tujuan data tersebut seharusnya digolongkan ke golongan A.

4. **True Negative** : Suatu data diklasifikan oleh sistem sebagai **bukan** golongan A, dan tujuan data tersebut dipilih sebagai **bukan** golongan A.

Dimana nantinya akurasi akan dihitung dengan persamaan :

$$Akurasi = \frac{TP + TN}{TP + FP + FN + TN} \tag{2.1}$$

Sedangkan ***Precision*** merupakan persentase data yang di klasifikasikan secara benar (***Correct***).

$$Precision = \frac{TP}{TP + FN} \tag{2.2}$$

Sedangkan ***Recall*** merupakan jumlah data yang **benar** di klasifikasikan secara benar.

$$Recall = \frac{TP}{TP + FP} \tag{2.3}$$

Sedangkan ***F-Measure*** adalah metode untuk mencari nilai tengah antara *Precision* dan *Recall* untuk meingkatkan perfomansi klasifikasi agar sistem menghasilkan akurasi yang lebih baik, dibandingkan menggunakan *Precision* atau *Recall* saja. Adapun untuk mendapatkan nilai *F-Measure* ini adalah sebagai berikut :

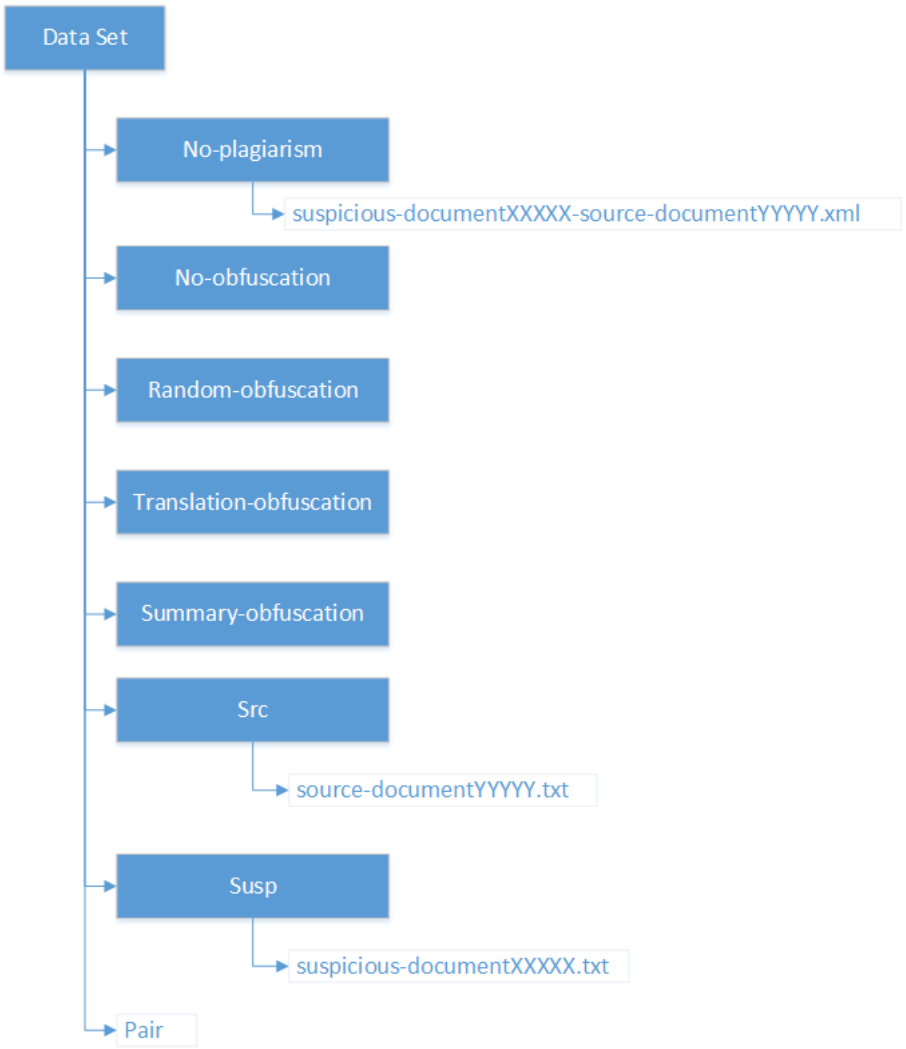
$$F = \frac{1}{\alpha\frac{1}{P} + (1-\alpha)\frac{1}{R}} = \frac{(\beta^2+1)PR}{\beta^2 P + R} \tag{2.4}$$

Namun biasanya untuk perhitungan, digunakan *blanced F1-Measure*. Dimana nilai $\beta = 1$ dan $\alpha = \frac{1}{2}$. Sehingga persamaan perhitungan *F-Measure* menjadi :

$$F = \frac{2PR}{P + R} \tag{2.5}$$

2.7. Dataset

Dataset yang digunakan pada tugas akhir ini diambil di *web* resmi PAN[2]. Gambar 2.1 menunjukan hierarki dari *dataset* yang digunakan.



Gambar 2.1: Dataset

Dataset diatas mempunyai bagian-bagian umum sebagai berikut :

1. **Susp**
   Kumpulan dokumen yang terindikasi plagiat melalui proses *Source Retrieval*.

2. **Src**
   Merupakan dokumen sumber di plagiat oleh dokumen Susp melalui proses *Source Retrieval*.

3. **Pair**
   Merupakan informasi pasangan dokumen yang terindikasi, formatnya adalah *suspicious-documentXXXXX.txt source-documentYYYYY.txt*. Dimana *suspicious-documentXXXXX.txt* merujuk pada dokumen yang ada di Susp, dan *source-documentYYYYY.txt* merujuk pada dokumen yang ada di Src.

4. ***No-Plagiarism, No-Obfuscation, Random-Obfuscation, -Obfuscation, Summary-Obfuscation***
   Bagian ini merupakan kelas hasil klasifikasi tindak plagiat dari file Pair.

   (a) ***No-Plagiarism*** : Tidak terdeketsi tindak plagiat.

   (b) ***No-Obfuscation*** : Tindak plagiat berupa *copy-paste*.

   (c) ***Random-Obfuscation*** : Tindak plagiat berupa menghilang / menambahkan kata pada kalimat yang digunakan.

   (d) ***Translation-Obfuscation*** : Tindak plagiat berupa menerjemahkan kata.

   (e) ***Summary-Obfuscation*** : Tindak plagiat berupa merangkum suatu kalimat/paragraf.

Sebagai contoh, pada file *pairs* terdapat informasi sebagai berikut :

$$suspicious - document00005.txt source - document01496.txt \tag{2.6}$$

Berarti, menurut proses proses *Source Retrieval* terdapat indikasi plagiat dari dokumen *suspicious-document00005.txt* bersumber dari dokumen *source-document01496.txt*. Dan menurut data yang ada, tindak plagiat pada dokumen tadi termasuk ke dalam *No-Obfuscation*, dimana tindak plagiat yang dilakukan berupa *copy-paste*. Dan setelah diteliti, tindak plagiat terbukti dikarenakan pada kedua dokumen terdapat kalimat yang sama.

Berikut merupakan potongan dokumen *suspicious-document00005.txt* :

*.....WAYS TO SEND YOUR DOCUMENTATION Fax to 304-724-0909 Scan and email to DSA@apus.edu Mail to APUS ATTN : Disability Accomodations 10110 Battleview Parkway Suite 114 Manassas, VA 20109 x ED502 * required berfore student may register for courses Prepare a short essay of approximately 300 words (one page) describing why you are interested in this particular degree program. Your sample should preferably be written in Word and double-spaced. The Education degree coordinator will use your writing sample to assess your written communications skills.*
** Writing Sample Two character references are required from people who can attest to your moral and ethical character. Example of such people include supervisors, religious leaders, military commanders, school officials, or others who know you well and can provide credible information about you. He served at the Pentagon as par of Joint Staff in support of Noble Eagle and Enduring Freedom.....*

Berikut merupakan potongan dokumen *source-document01496.txt* :

*x ED502 * required berfore student may register for courses Prepare a short essay of approximately 300 words (one page) describing why you are interested*

in this particular degree program. Your sample should preferably be written in Word and double-spaced. The Education degree ccordinator will use your writing sample to assess your written communications skills.
** Writing Sample Two character references are required from people who can attest to your moral and ethical character. Example of such people include supervisors, religious leaders, military commanders, school officials, or others who know you well and can provide credible information about you. Forms will be provided to you by your admissions representative. Once forms are completed and signed by the references, send them to APUS following the document submission instructions below.....*

Bagian yang di *highlight* merupakan bagian yang berhasil diindikasi adanya tindak plagiat.

Bab 3. Metodologi dan Desain Sistem

3.1. Analisis Kebutuhan

Analisis kebutuhan mencakup dari penentuan spesifikasi perangkat yang akan digunakan, baik perangkat lunak maupun perangkat keras.

3.1.1 Spesifikasi Perangkat Keras

Perangkat keras yang akan digunakan untuk tugas akhir ini adalah sebagai berikut :

1. Prosesor Intel i5-3470S Quad Core.
2. RAM 6144 MB.
3. Harddisk 300 GB.

3.1.2 Spesifikasi Perangkat Lunak

Perangkat lunak yang akan digunakan untuk tugas akhir ini adalah sebagai berikut :

1. Sistem Operasi Windows 10 Pro 64-bit.
2. Python, sebagai bahasa pemrograman utama implementasi sistem.
3. Weka, sebagai alat bantu klasifikasi

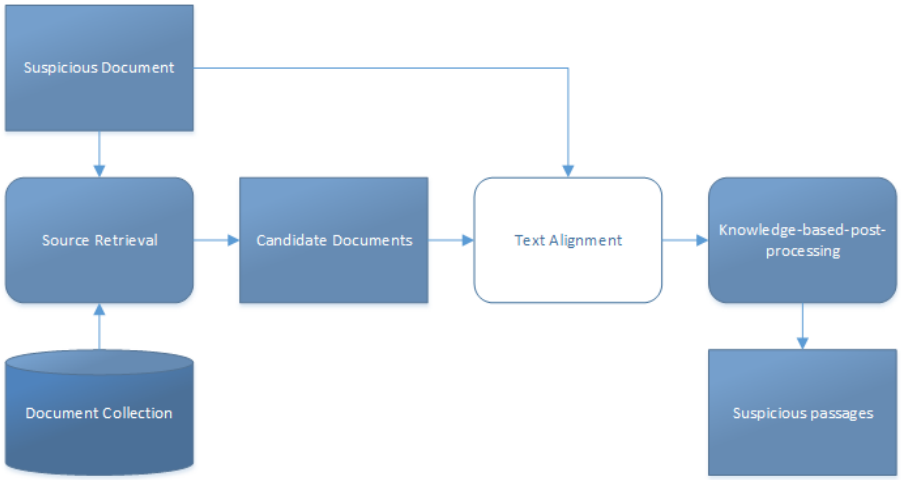
3.2. Perancangan Sistem

3.2.1 Gambaran Umum Sistem

Gambaran umum sistem yang akan dibuat adalah. Sudah didapat informasi mengenai dokumen yang terindikasi X plagiat beserta sumbernya Y. Kedua dokumen akan di ekstrak dan diambil cirinya dengan metode *N-Gram* lalu membuat *seed* yang akan di bandingkan di proses *merging*. Dari proses *merging* akan dilanjutkan ke proses *filtering* dan mengklasifikasi hasil yang ada kedalam klasifier untuk menentukan jenis tindak plagiat.

3.2.2 Alur Sistem

Pada kasus sebenarnya, untuk menyusun sistem *Plagiarism Detection* ini membutuhkan 2 *tasks* utama, yaitu ***Source Retrieval*** dan ***Text Alignment***. Sehingga sistem yang seharusnya dibangun ditunjukan oleh gambar 3.1.



Gambar 3.1: Alur Sistem Keseluruhan

Namun, pada tugas akhir ini *Candidate Documents* yang ada pada gambar 3.1 sudah didapatkan sebelumnya. Sehingga pada tugas akhir ini akan fokus membahas mengenai *task Text Alignment*. Untuk membuktikan kebenaran dari *Candidate Documents* hasil dari *task Source Retrieval* sebelumnya.

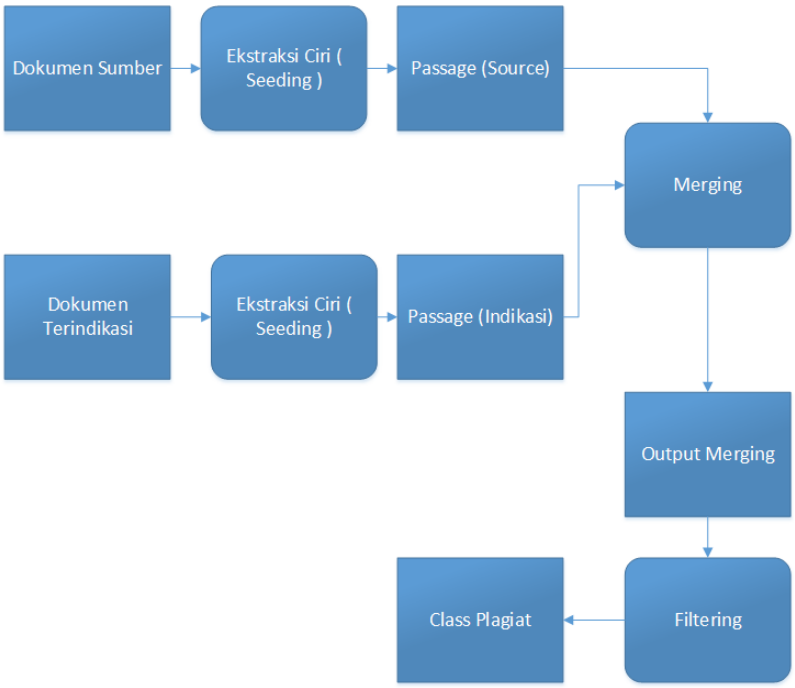
Adapun sistem yang akan dibangun ditunjukan oleh gambar 3.2.

3.3. Pengujian

Pengujian sistem yang dibangun dilakukan menggunakan *Tira[12]* yang mengacu pada paper acuan[6, 13]. Dimana nilai perfomansi dihitung berdasarkan akurasi pengklasifikasian oleh sistem. Kelas klasifikasi yang ada adalah :

1. ***No-Plagiarism***
2. ***No-Obfuscation***
3. ***Random-Obfuscation***
4. ***Summary-Obfuscation***

Dimana karakteristik dari kelas yang ada sudah dijelaskan pada Bab 2. Nilai akurasi didapatkan dengan menggunakan *Precision, Recall, F-Measure*. Dan pada akhir sistem, data yang akan di*outputkan* ditunjukan oleh tabel 3.3.



Gambar 3.2: Alur Sistem

Tabel 3.1: Tabel Perfomansi Sistem

Corpus	Pairs	PlagDet	Precision	Recall	Granularity
pan14-training-corpus					

Daftar Pustaka

[1] Kamus besar bahasa indonesia. [Online]. Available: http://kbbi.web.id/plagiat

[2] PAN. Plagiarism detection. [Online]. Available: http://pan.webis.de/clef14/pan14-web/plagiarism-detection.html

[3] M. P. NASIONAL", "Peraturan menteri pendidikan nasional republik indonesia nomor 17 tahun 2010 tentang pencegahan dan penanggulangan plagiat di perguruan tinggi."

[4] V. Rijsbergen, *INFORMATION RETRIEVAL.*

[5] A. Sukma, B. P. Santoso, D. Ramadhan, N. M. A. K. Wiraswari, and T. R. Sari, "Klasifikasi dokumen bahasa jawa menggunkan metode n-gram."

[6] P. Gross and P. Modaresi, "Plagiarism alignment detection by merging context."

[7] M. Potthast, M. Hagen, T. Gollub, M. Tippmann, J. Kiesel, P. Rosso, E. Stamatatos, and B. Stein, "Overview of the 5th international competition on plagiarism detection."

[8] M. Potthast, M. Hagen, A. Beyer, M. Busse, M. Tippmann, P. Rosso, and B. Stein1, "Overview of the 6th international competition on plagiarism detection."

[9] F. ALlvi, M. Stevenson, and P. Clogh, "Hashing and merging heuristics for text reuse detection."

[10] K. Weinberger, J. Blitzer, and L. Saul, "Distance metric learning for large margin nearest neighbor classification."

[11] G. Hripcsaka and A. S. Rothschild, *Agreement, the F-Measure, and Reliability in Information Retrieval.*

[12] M. Potthast, M. Hagen, T. Gollub, M. Tippmann, J. Kiesel, P. Rosso, E. Stamatatos, and B. Stein, *Information Access Evaluation. Multilinguality, Multimodality, and Visualization, Lecture Notes in Computer Science.*

[13] M. Potthast, B. Stein, A. Barron-Cedeno, and P. Rosso, "An evaluation framework for plagiarism detection."