# PENDAHULUAN

## Latar Belakang

Pesatnya perkembangan internet menyebabkan semakin banyaknya informasi yang tersedia. Hal ini memudahkan seseorang dalam melakukan penjiplakan suatu karya. Penjiplakan menurut Kamus Besar Bahasa Indonesia (KBBI) berarti menggambar atau menulis garis-garis gambaran atau tulisan yang telah tersedia (dengan menempelkan kertas kosong pada gambar atau tulisan yang akan ditiru), mencontoh atau meniru tulisan atau pekerjaan orang lain, mencuri karangan orang lain dan mengakui sebagai karangan sendiri, mengutip karangan orang lain tanpa seizin penulisnya.

Tindakan plagiarisme tersebut dapat membuat seseorang menjadi malas berkreatifitas karena tidak mau berfikir. Untuk itu tindakan plagiarisme harus ditekan dan salah satu caranya adalah dengan mendeteksi plagiarisme dokumen teks.

Berdasarkan uraian di atas, sudah banyak dibuat sistem pendeteksi plagiarisme pada dokumen teks. Salah satunya menggunakan Algoritma SCAM untuk mendeteksi plagiarisme pada dokumen teks. Algoritma SCAM memecah dokumen menjadi potongan kata. Kemudian ditambahkan ke dalam urutan indeks dan digunakan untuk perbandingan dengan masukan dokumen yang baru. Menurut uraian di atas peneliti tertarik untuk mengadakan penelitian mendeteksi plagiarisme pada dokumen teks menggunakan algoritma SCAM.

## Rumusan Masalah

Berdasarkan latar belakang pemilihan judul, maka yang menjadi permasalahan adalah bagaimana mengimplementasikan algoritma SCAM untuk mendeteksi plagiarisme pada dokumen teks.

## Tujuan

Membangun sebuah sistem deteksi plagiat (*Plagiarism detection system*) dengan menggunakan algoritma SCAM.

## Ruang Lingkup

Agar pengerjaan tugas akhir ini menjadi lebih terarah dan mendapatkan hasil yang lebih spesifik, maka sistem yang dirancang dibatasi pada ruang lingkup pembahasan sebagai berikut:

1. Algoritma yang digunakan dalam pembuatan sistem ini menggunakan algoritma SCAM.
2. Sistem ini dirancang untuk menentukan seberapa besar kemiripan sebuah dokumen teks yang berformat \*pdf.

## Metodologi

Metodologi yang digunakan dalam menyelesaikan pembangunan perangkat lunak terdiri dari :

1. Pengumpulan Data

Pengumpulan data dengan studi kepustakaan dilakukan dengan cara mempelajari jurnal, buku-buku, dan sumber terkait lainnya yang menjadi acuan dalam membangun sistem deteksi plagiat menggunakan algoritma SCAM.

1. Analisis dan Perancangan Algoritma

Pada tahap ini dilakukan analisis cara kerja deteksi plagiarisme dengan algoritma SCAM terhadap dokumen text.

1. Implementasi

Perancangan sistem dibuat secara terstruktur, sesuai dengan tahapan-tahapan yang harus dilakukan. Sebuah sistem yang dapat mendeteksi plagiarisme pada dokumen teks dengan algoritma SCAM.

## Sistematika Penulisan

Secara garis besar materi laporan tugas akhir ini terbagi dalam beberapa bab yang tersusun sebagai berikut :

**BAB I PENDAHULUAN**

Pada bab ini akan menjelaskan mengenai latar belakang, permasalahan yang diteliti, tujuan, dan metodologi penyelesaian masalah.

**BAB II LANDASAN TEORI**

Pada bab ini akan menjelaskan mengenai dasar-dasar pengetahuan yang dapat membantu merancang dan memperkuat sistem.

**BAB III ANALISIS DAN PERANCANGAN**

Bab ini memuat tentang analisis kebutuhan serta perancangan aplikasi sistem deteksi plagiat yang akan dibuat.

**BAB IV IMPLEMENTASI DAN PENGUJIAN**

Bab ini berisi hasil-hasil penelitian dan pengujian sistem yang telah dibangun.Kemudian dijelaskan juga hal-hal yang dianalisis serta bagaimana hasil analisis dari pengujian yang dilakukan.

**BAB V PENUTUP**

Bab ini berisi kesimpulan dan saran yang berkaitan dengan pembuatan sistem dan penulisan laporan berdasarkan pada yang telah diuraikan pada bab - bab sebelumnya**.**

# LANDASAN TEORI

## Plagiarisme

### Definisi Plagiarisme

Plagiat merupakan sebuah tindakan menjiplak ide atau karya orang lain. Dalam Kamus Besar Bahasa Indonesia (KBBI) plagiat (*plagiarism*) diartikan sebagai pengambilan karangan (pendapat dsb) orang lain dan menjadikannya seolah-olah karangan (pendapat dsb) sendiri, misal menerbitkan karya tulis orang lain atas nama dirinya sendiri; jiplakan [[1](#kbb)].

Plagiarisme sesuai dengan pendapat Ir. Balza Achamd, M.Sc.E adalah berbuat sesuatu seolah-olah karya orang lain tersebut adalah karya kita dan mengakui hasil karya tersebut adalah milik kita, sedangkan menurut Brotowidjoyo (1993:86) plagiarisme merupakan pembajakan berupa fakta, penjelasan, ungkapan, dan kalimat orang lain secara tidak sah [[2](#Aud09)]. Di Indonesia sendiri, perlindungan akan hal cipta telah diatur dalam Undang-Undang Republik Indonesia Nomor 19 Tahun 2002 Tentang Hak Cipta [[2](#Aud09)].

Tipe plagiarisme dapat dijelaskan melalui bagaimana cara melakukannya. Terdapat tiga jenis plagiarisme, yaitu *pathwriting*, *inappropriate praraphasing*, dan *summaries*. Titik poin dari ketiga jenis plagiarisme ini adalah apakah penulis menyalin, mengubah, dan menyingkat tulisan orang lain dengan menggunakan kaidah pengutipan dan parafrase secara baik [[3](#Muh19)].

### Bentuk – Bentuk Plagiarisme

Mengutip tulisan Siti Wachidah mengenai bentuk - bentuk plagiarisme yang didasari dari tulisan Roig, sebagai berikut [[4](#Sit13)] :

1. Menyalin pernyataan yang bukan merupakan pengetahuan umum dari orang lain, sebagian atau keseluruhan, dengan cara mengurangi dan/atau menambah kata-kata sendiri, atau menggantinya dengan kata yang hampir sama, tanpa menyebutkan sumbernya dengan lengkap dan benar.
2. Mengutip kalimat, frasa, dan kata tentang suatu konsep atau fakta yang belum menjadi pengetahuan umum persis dari sumbernya (verbatim) tanpa memberi tanda kutip meskipun menyebutkan sumber rujukan secara lengkap dan benar.
3. Mengutip kalimat, frasa, dan kata tentang suatu konsep atau fakta yang belum menjadi pengetahuan umum persis dari sumbernya tanpa menyebutkan sumber rujukan secara lengkap dan benar.
4. Mengutip dari sumbernya kalimat demi kalimat, atau paragraf demi paragraf tanpa menggunakan tanda petik meskipun menyebutkan sumbernya secara lengkap dan benar.
5. Mengambil sebagian pernyataan orang lain yang bukan merupakan pengetahuan umum, mengubah cara penyampaiannya secara keseluruhan dengan tata bahasa dan kata-kata yang tidak sama, tanpa menyebutkan sumbernya secara lengkap dan benar.
6. Mengutip kalimat, frasa, dan kata konsep persis dari sumbernya dengan menyebutkan sumber rujukan secara lengkap dan benar tetapi tanpa menggunakan tanda kutip.
7. Mengambil sebagian pernyataan orang lain, menyebutkan sumbernya dengan benar, tetapi hanya mengubah sedikit kata-kata atau tata bahasanya (misalnya, aktif ke pasif, ‘*cause*’ ke ‘*reason*’) meskipun dengan menyebutkan sumbernya secara lengkap dan benar.
8. Dalam meringkas atau memparafrasa pernyataan orang lain masih tetap menggunakan struktur kalimat persis dengan yang digunakan sumbernya, meskipun telah menyebutkan sumbernya dengan benar dan lengkap.
9. Mengambil atau menggunakan gagasan (a.l. penjelasan, pendapat, teori, kesimpulan, hipotesis, metafora) orang lain, sebagian atau keseluruhan, dengan kata-kata sendiri tanpa menyebutkan sumber secara lengkap dan benar.
10. Mengambil atau menggunakan gagasan yang disampaikan secara kasual (a.l. ngobrol), termasuk oleh orang biasa atau yang tidak memiliki kredibilitas keilmuan (a.l. teman, tukang sapu) tanpa menyebutkan sumber tersebut.

## SCAM

SCAM *(Stanford Copy Analysis Mechanism*) adalah sebuah algoritma untuk mendeteksi plagiarisme dengan cara membandingkan kemunculan frekuensi kata dari dokumen permintaan dan yang lainnya dimana metode ini sangat cepat dan dapat menemukan kalimat yang disalin sebagian dan efisien jika ada salinan signifikan. Jika hanya ada beberapa bagian yang disalin dari dokumen lain, metode SCAM menunjukkan hasil kurang presisi [[5](#LSi16)].

Model perhitungan SCAM sendiri menggunakan pendekatan secara *Information Retrieval*. Dasar metode pencocokan ini hampir sama dengan metode pencocokan yang sering digunakan dalam *Information Retrieval* yaitu VSM (*Vector Space Model*). Namun kelebihan dari SCAM dibandingkan dengan VSM adalah SCAM mampu mengatasi *overlap* yang terjadi terlepas dari perbedaan lenght antar dokumen. SCAM mampu secara tepat mendeteksi salinan (*copy*) *subset* atau *superset* [[6](#Anz11)].

Cara pendeteksian yang dilakukan oleh SCAM adalah dengan membagi teks kedalam potongan *term/ word chunk*. Setelah itu *term* tersebut akan dimasukan kedalam dokumen indeks untuk selanjutnya dapat dicocokan dengan dokumen yang lainnya.

### *Closeness set C* (R,T)

Dalam perhitungan tingkat kemiripan antar dokumen dibagi menjadi 3 tahapan, yaitu yang pertama adalah menentukan nilai *closeness set c*(R,T). Dimana R merupakan dokumen registered atau dokumen asli dan T merupakan dokumen test atau dokumen uji. *Closeness set c*(R,T) digunakan untuk menentukan himpunan *chunk* yang digunakan pada tingkat yang sama dalam 2 dokumen [[5](#LSi16)]. Adapun perhitungan closeness set c(R,T) adalah sebagai berikut [[6](#Anz11)] :

Keterangan :

*fi*(R) merupakan jumlah kemunculan chunk di dokumen registered (R)

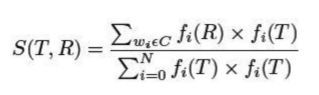
*fi*(T) merupakan jumlah kemunculan chunk di dokumen test (T)

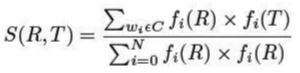
Nilai ɛ adalah nilai konstan yang lebih besar dari 2 (ɛ > 2).

Jika hasil *Closeness set C*(R,T) tidak terpenuhi (hasilnya tidak lebih dari 0) maka tidak bisa lanjut ke tahap perhitungan selanjutnya dikarenakan perbedaan *chunk* diantara kedua dokumen tersebut cukup jauh.

### Subset (R,T) dan (T,R)

Selanjutnya adalah mengukur subset dari dokumen R yang merupakan subset dari dokumen T juga sebaliknya. Dimana perhitungannya dilakukan dengan cara sebagai berikut :





Perhitungan ini bertujuan untuk mengukur sejauh mana tumpang tindih yang terjadi pada dokumen R dan T. Perhitungan ini menghitung subset antara pasangan dokumen T dan R yang hanya mempertimbangkan kata yang sama diantara kedua dokumen tersebut.

### *Similarity(*R,T)

Selanjutnya akan dihitung tingkat kemiripan antara dua dokumen yang ada,yaitu dengan perhitungan sebagai berikut :

*Similarity* = *max* [S(T,R), S(R,T)]

Dalam perhitungan nilai similarity antar dua dokumen ini, ditentukan dari nilai maksimum dari perhitungan S(T,R) dan S(R,T). Dimana nilai maksimum lah yang akan diambil sebagai nilai ukuran kemiripan antara dokumen R dan T. Jika nilai sim(R,T) > 1 maka nilai dari sim(R,T) akan menjadi 1. Hal ini karena tidak ada informasi tambahan yang dikumpulkan saat nilai sim(T,R) lebih besar dari 1, sehingga diatur bahwa nilai maksimum menjadi 1. Nilai kesamaan diekspresikan dengan rentang nilai antara 0 sampai 100% [[7](#Nar)].

## *Preprocessing*

*Preprocessing* merupakan tahap awal yang perlu dilakukan dalam pemrosesan teks untuk membentuk struktur data yang baik. Tujuan dari *preprocessing* adalah untuk mendapatkan indeks *term* dari dokumen teks agar dapat diproses untuk tahap berikutnya [[8](#Deset)]. Langkah-langkah yang dilakukan dalam tahap *preprocessing* adalah *cleaning*, *tokenisasi*, *stopword* dan *stemming*.

### *Cleaning*

Tahapan ini dapat dikatakan adalah tahap pembersihan *noise* dalam teks. *Noise* merupakan suatu bentuk data yang nantinya mungkin akan mengganggu proses pengolahan data tersebut. *Noise* yang dimaksud dalam penelitian ini adalah karakter selain huruf alfabet pada teks [[8](#Deset)].

### Tokenisasi

Secara garis besar tokenisasi adalah tahap memecah sekumpulan karakter dalam suatu teks kedalam satuan kata. Sekumpulan karakter tersebut dapat berupa karakter *whitespace*, seperti *enter*, tabulasi, spasi. Namun untuk karakter petik tunggal („), titik (.), semikolon (;), titik dua (:) atau lainnya, juga dapat memiliki peran yang cukup banyak sebagai pemisah kata. Sebuah titik (.) biasanya untuk tanda akhir kalimat, tapi dapat juga muncul dalam singkatan, inisial orang, alamat internet,dll. Kemudian tanda hypen (-) biasanya muncul untuk menggabungkan dua token yang berbeda untuk membentuk token tunggal. Tapi dapat pula ditemukan untuk menyatakan rentang nilai, kata berulang, dsb. Atau karakter slash (/) sebagai pemisah file atau direktori atau url ataupun untuk menyatakan “dan atau” [[9](#Abd)].

### Stopword

Pada tahap ini adalah proses pembuangan stopword yang dimaksudkan untuk mengetahui suatu kata masuk kedalam stopword atau tidak. Pembuangan stopword adalah proses kata yang tidak diperlukan-penghubung-tidak memiliki makna penting [[10](#Hid18)].

### Stemming

Stemming merupakan tahapan untuk mengubah kata yang merupakan hasil dari proses *filtering* menjadi kata dasar semuanya. Tahap ini dilakukan dengan menghapus semua imbuhan baik imbuhan pada awalan kata maupun pada akhiran kata. Keluaran dari tahap ini nantinya akan dijadikan *term* [[8](#Deset)]. Dalam bahasa Indonesia sendiri telah dikembangkan beberapa jenis algoritma stemming dimana salah satunya adalah algoritma Nazief & Andriani. Proses stemming pada teks berbahasa Indonesia lebih rumit/kompleks karena terdapat variasi imbuhan yang harus dibuang untuk mendapatkan *root word* (kata dasar) dari sebuah kata. Algoritma yang dibuat oleh Bobby Nazief dan Mirna Adriani ini memiliki tahap-tahap sebagai berikut [[11](#Agu09)]:

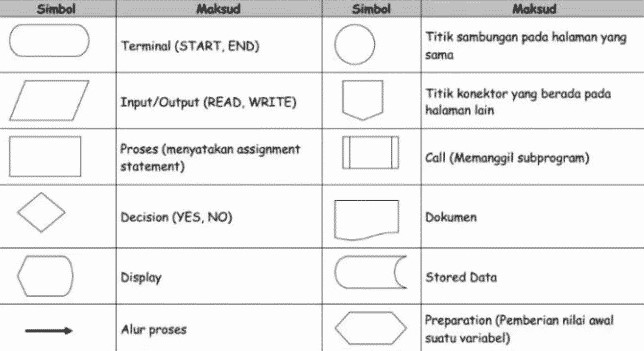
1. Cari kata yang akan distem dalam kamus. Jika ditemukan maka diasumsikan bahwa kata tesebut adalah *root word*. Maka algoritma berhenti.
2. Inflection Suffixes (“-lah”, “-kah”, “-ku”, “-mu”, atau “-nya”) dibuang. Jika berupa particles (“-lah”, “-kah”, “-tah” atau “-pun”) maka langkah ini diulangi lagi untuk menghapus Possesive Pronouns (“-ku”, “-mu”, atau “-nya”), jika ada.
3. Hapus Derivation Suffixes (“-i”, “-an” atau “-kan”). Jika kata ditemukan di kamus, maka algoritma berhenti. Jika tidak maka ke langkah 3a
4. Jika “-an” telah dihapus dan huruf terakhir dari kata tersebut adalah “-k”, maka “ k” juga ikut dihapus. Jika kata tersebut ditemukan dalam kamus maka algoritma berhenti. Jika tidak ditemukan maka lakukan langkah 3b.
5. Akhiran yang dihapus (“-i”, “-an” atau “-kan”) dikembalikan, lanjut ke langkah 4.
6. Hapus Derivation Prefix. Jika pada langkah 3 ada sufiks yang dihapus maka pergi ke langkah 4a, jika tidak pergi ke langkah 4b.
   1. Periksa tabel kombinasi awalan-akhiran yang tidak diijinkan. Jika ditemukan maka algoritma berhenti, jika tidak pergi ke langkah 4b.
   2. For i = 1 to 3, tentukan tipe awalan kemudian hapus awalan. Jika root word belum juga ditemukan lakukan langkah 5, jika sudah maka algoritma berhenti. Catatan: jika awalan kedua sama dengan awalan pertama algoritma berhenti.
7. Melakukan Recoding.
8. Jika semua langkah telah selesai tetapi tidak juga berhasil maka kata awal diasumsikan sebagai root word. Proses selesai.

## Flowchart

Flowchart adalah representasi secara simbolik dari suatu algoritma atau prosedur untuk menyelesaikan suatu masalah, dengan menggunakan flowchart akan memudahkan pengguna melakukan pengecekan bagian-bagian yang terlupakan dalam analisis masalah, disamping itu flowchart juga berguna sebagai fasilitas untuk berkomunikasi antara pemrogram yang bekerja dalam tim suatu proyek. Flowchart membantu memahami urutan-urutan logika yang rumit dan panjang. Flowchart membantu mengkomunikasikan jalannya program ke orang lain (bukan pemrogram) akan lebih mudah . Berikut pedoman-pedoman dalam flowchart [[12](#San17)] [[13](#Wid)] :

1. Bagan alir sebaiknya digambar dari atas ke bawah dan mulai dari bagian kiri suatu halaman.
2. Kegiatan di dalam bagan alir harus ditunjukkan dengan jelas.
3. Harus ditunjukkan dari mana kegiatan akan dimulai dan dimana akan berakhirnya
4. Masing-masing kegiatan di dalam bagan alir sebaiknya digunakan kata yang mewakili suatu pekerjaan.
5. Masing-masing kegiatan di dalam bagan alir harus di dalam urutan yang semestinya.
6. Kegiatan yang terpotong dan akan disambung di tempat lain harus ditunjukkan dengan jelas menggunakan simbol penghubung.
7. Gunakan simbol-simbol bagan alir standar.

Berikut gambar simbol-simbol flowchart :



# ANALISIS DAN PERANCANGAN

## Analisis Proses Berjalan

Proses antara perhitungan dokumen terdaftar dengan dokumen uji berlangsung bersamaan. Proses menerima inputan dokumen pdf yang akan dicek, dimana nantinya dokumen tersebut akan di test dengan dokumen - dokumen yang sudah di inputkan sebelumnya ke dalam sistem . Adapun langkah-langkah yang dilakukan dalam proses adalah sebagai berikut :

1. Normalisasi dokumen :

Normalisasi yang dimaksud adalah, dokumen akan di *convert* menjadi string menggunakan library PdfToText dengan maksud agar dokumen tersebut bisa diproses ketahap selanjutnya.

1. Preprocessing :

Pada tahap preprocessing terdiri dari beberapa langkah, yaitu cleaning, tokenisasi, stopword, stemming.

1. Proses pengukuran tingkat kemiripan dengan SCAM

Pada tahapan ini dilakukan proses perhitungan tingkat kemiripan antar dokumen uji dengan dokumen terdaftar menggunakan perhitungan SCAM.

## Gambaran Sistem