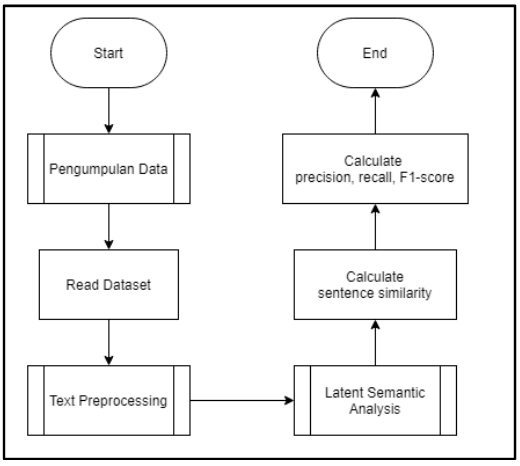
BAB 3

METODOLOGI PENELITIAN

3.1 Gambaran Umum Metodologi Penelitian

Gambaran umum metodologi penelitian dibuat untuk menjelaskan alur proses penelitian yang dilakukan dan speksifikasi *hardware* dan *software* yang digunakan. Tahap pertama pada penelitian adalah melakukan pengumpulan data berita difabel dari *dataset* IndoSum. Setelah mendapatkan data, dilakukan *text preprocessing* untuk mempersiapkan berita agar lebih mudah untuk diolah. Kemudian dilakukan peringkasan berita dengan menggunakan metode LSA. Setelah itu, dilakukan perhitungan untuk mencari nilai *cosine similarity* antara ringkasan yang telah dibuat oleh LSA dengan ringkasan yang ada pada *dataset* IndoSum. *Flowchart* gambaran umum metodologi penelitian dapat dilihat pada Gambar 3.1.



Gambar 3.1 *Flowchart* Gambaran Umum Metodologi Penelitian

Berikut adalah speksifikasi *hardware* dan *software* yang digunakan dalam pengimplementasian metode ini.

Spesifikasi *hardware* yang digunakan:

a. Processor Intel Core i5-4210U
b. RAM 8 GB DDR3
c. NVIDIA GeForce 920M

Spesifikasi *software* yang digunakan:

a. Anaconda 3
b. Flask
c. Jupyter Notebook
d. Python 3.7
e. Visual Studio Code
f. Google Chrome
g. Google Colaboratory
h. OS Windows 10 Pro

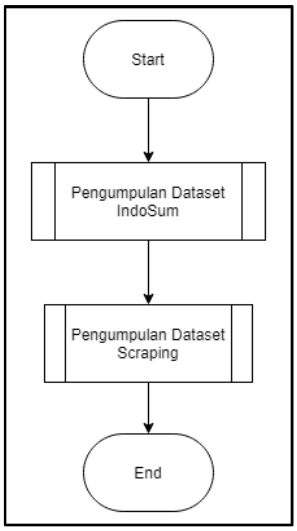
3.2 Telaah Literatur

Telaah literatur dilakukan dengan mencari, membaca, dan mempelajari teori-teori yang berhubungan dengan penelitian, seperti Disabilitas, Difabel, *Text Summarization*, LSA, dan Teknik Evaluasi.

3.3 Pengumpulan Data

Gambar 3.2 merupakan *flowchart* dari proses pengumpulan data. Pada penelitian ini digunakan dua *dataset*, yaitu *dataset* IndoSum dan *dataset* hasil

scraping dari situs liputan6.com, newsdifabel.com, difabel.tempo.co, kompas.com, cnnindonesia.com, kumparan.com, merdeka.com, juara.bolasport.com, dan antaranews.com. Proses pengumpulan berita dari kedua *dataset* adalah sebagai berikut.



Gambar 3.2 *Flowchart* Pengumpulan Data

3.3.1 Pengumpulan Dataset IndoSum

Pada *dataset* IndoSum, terdapat minimal 18.774 berita. Pada *dataset* ini, setiap berita mengandung informasi mengenai kategori, *gold labels*, sumber berita, alamat sumber berita, dan ringkasan. Data ringkasan yang ada pada *dataset* IndoSum ini dibuat secara manual oleh dua orang penutur asli Bahasa Indonesia. *Dataset* IndoSum sudah pernah digunakan oleh Kemal Kurniawan (2018) dalam penelitiannya yang berjudul "*INDOSUM: A New Benchmark Dataset for Indonesian Text Summarization*". Bentuk dari *dataset* IndoSum dapat dilihat pada Gambar 3.2.

	category	gold_labels	id	paragraphs	source	source_url	summary
0	tajuk utama	[[False, True], [True, True], [False, False, F...	1501893029-lula-kamal-dokter-ryan-thamrin-saki...	[[[Jakarta, ,, CNN, Indonesia, -, -, Dokter, R...	cnn indonesia	https://www.cnnindonesia.com/hiburan/201708041...	[[Dokter, Lula, Kamal, yang, merupakan, selebr...
1	teknologi	[[False, False, False, False], [False, True, T...	1509072914-dua-smartphone-zenfone-baru-tawarka...	[[[Selfie, ialah, salah, satu, tema, terpanas,...	dailysocial.id	https://dailysocial.id/post/dua-smartphone-zen...	[[Asus, memperkenalkan, , ZenFone, generasi, ...
2	hiburan	[[True], [True], [False, False], [False], [Fal...	1510613677-songsong-visit-2020-bengkulu-perkua...	[[[Jakarta, ,, CNN, Indonesia, -, -, Dinas, Pa...	cnn indonesia	https://www.cnnindonesia.com/gaya-hidup/201711...	[[Dinas, Pariwisata, Provinsi, Bengkulu, kemba...
3	tajuk utama	[[True, True], [False, False, False], [True], ...	1502706803-icw-ada-kejanggalan-atas-tewasnya-s...	[[[Merdeka.com, -, Indonesia, Corruption, Watc...	merdeka	https://www.merdeka.com/peristiwa/icw-merasa-a...	[[Indonesia, Corruption, Watch, (, ICW,), mem...
4	tajuk utama	[[False, True], [True, True, True], [False], [...	1503039338-pembagian-sepeda-usai-upacara-penur...	[[[Merdeka.com, -, Presiden, Joko, Widodo, (, ...	merdeka	https://www.merdeka.com/peristiwa/usai-upacara...	[[Jokowi, memimpin, upacara, penurunan, bender...

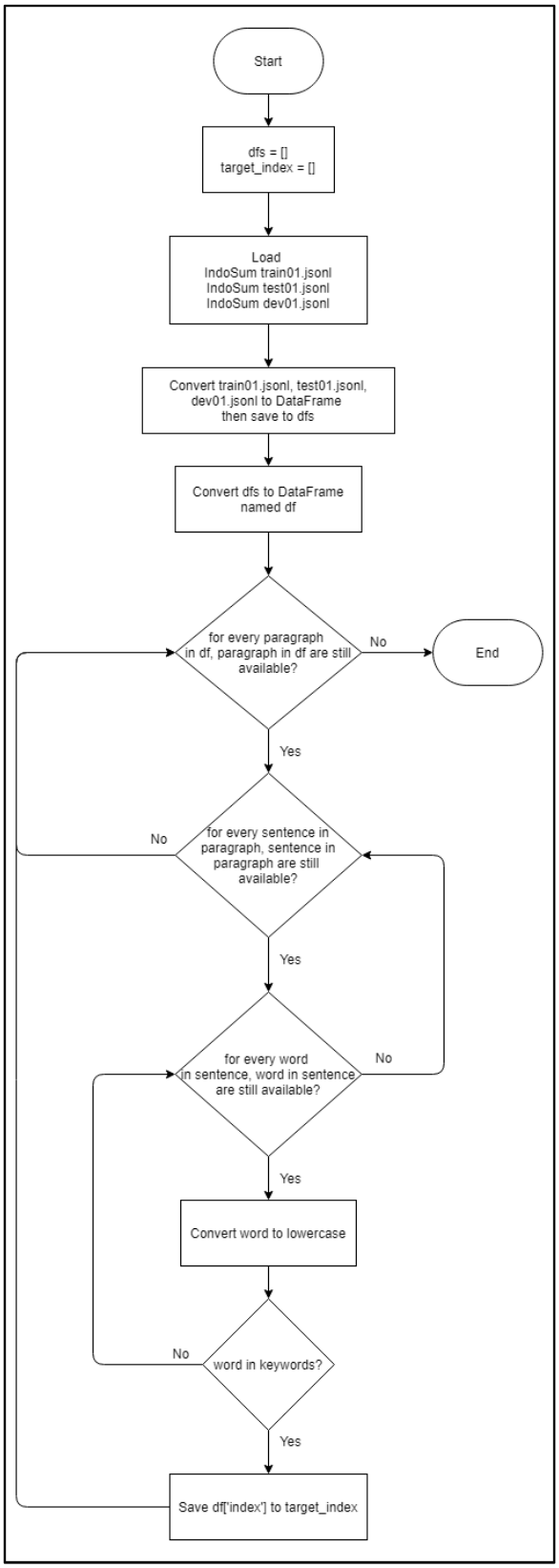
Gambar 3.3 Bentuk *Dataset* IndoSum

Pada *dataset* IndoSum terdapat 5 (lima) bentuk data yang terbagi menjadi 3 (tiga) *subset*, yaitu *training subset*, *test subset*, dan *development subset*. Setiap bentuk data mengandung 18.774 berita yang sama. Oleh karena itu, pada penelitian ini hanya akan digunakan satu bentuk data. Struktur *file* yang ada pada *dataset* IndoSum dapat dilihat pada Gambar 3.3.

Gambar 3.4 Struktur *File Dataset* IndoSum

Dataset IndoSum bukan *dataset* yang hanya mengandung berita difabel. Sehingga dilakukan proses penyaringan terhadap *dataset* IndoSum untuk mendapatkan berita difabel. Proses penyaringan dilakukan dengan menggunakan

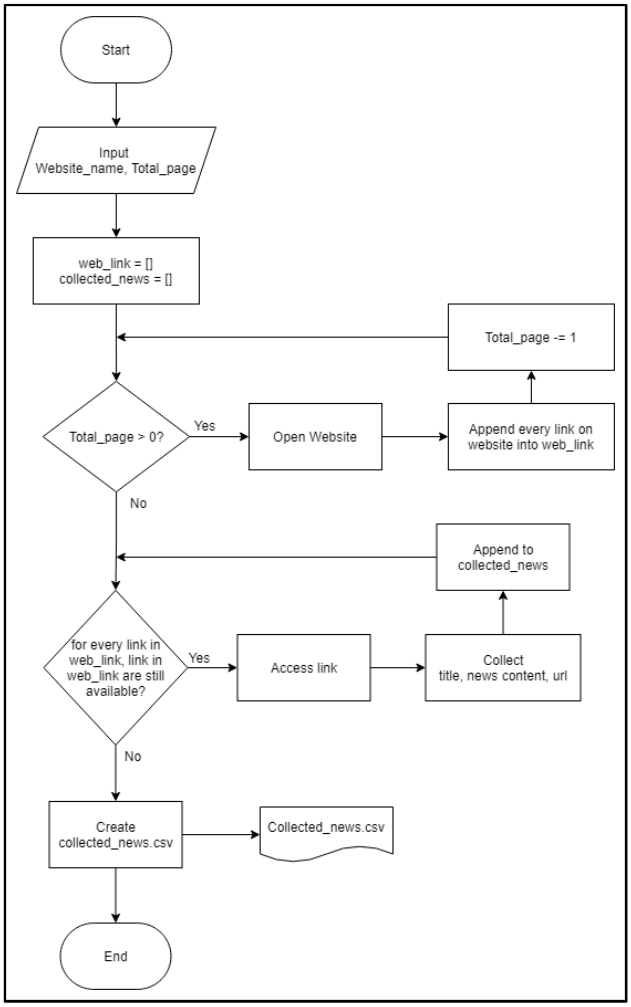
kata kunci, yaitu “autisme”, “autis”, “disabilitas”, “difabel”, “penyandang”, “tunarungu”, “rungu”, “tunanetra”, “netra”, “tunawicara”, “wicara”, “tunadaksa”, “daksa”, “tunalaras”, “tunagrahita”, “tunaganda”, dan ”paragames”. Terdapat 74 berita yang didapatkan setelah proses penyaringan. Kemudian dilakukan penyaringan secara manual untuk memastikan bahwa semua berita yang didapatkan merupakan berita difabel. Setelah dilakukan penyaringan jumlah berita yang didapatkan adalah sebanyak 66 berita. *Flowchart* pengumpulan *dataset* IndoSum dapat dilihat pada Gambar 3.4.



Gambar 3.5 *Flowchart* Pengumpulan *Dataset* IndoSum

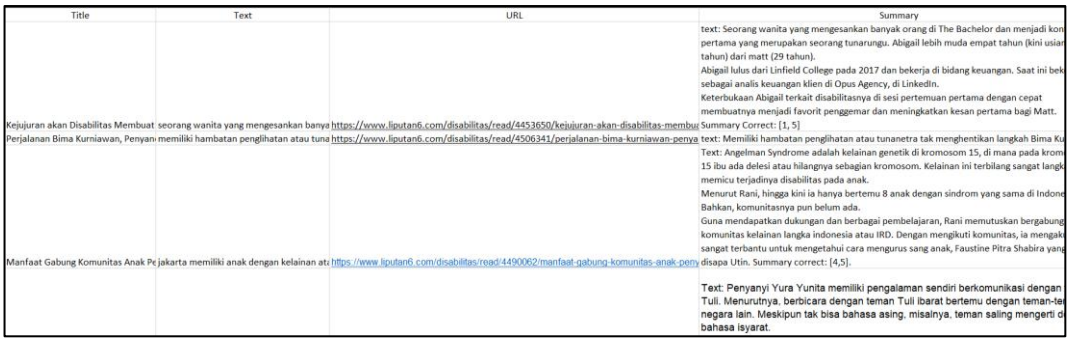
3.3.2 Pengumpulan Dataset Scraping

Gambar 3.6 merupakan *flowchart* dari proses pengumpulan *dataset scraping*. Proses pengumpulan *dataset scraping* dilakukan dengan menggunakan bahasa pemrograman Python dengan bantuan *library* beautifulsoup dan urlib. *Library* beutifulsoup digunakan untuk mengubah halaman situs menjadi bentuk teks dan membaca seluruh konten *Hyper Text Markup Language* (HTML) yang diakses menggunakan *library* urlib.



Gambar 3.6 *Flowchart* Pengumpulan *Dataset Scraping*

Gambar 3.7 merupakan bentuk *dataset scraping*. Pada pengumpulan *dataset* ini, terdapat sembilan situs yang menjadi target untuk *scraping* adalah liputan6.com, newsdifabel.com, difabel.tempo.co, kompas.com, cnnindonesia.com, kumparan.com, merdeka.com, juara.bolasport.com, dan antaranews.com. Setelah itu, ditentukan jumlah halaman untuk diproses. Kemudian dilakukan pengambilan seluruh *link* berita yang ada pada halaman situs dan mengambil informasi berupa judul, teks berita, dan alamat berita, dari setiap *link* yang telah didapatkan. Informasi yang telah didapat tersebut ke dalam *file comma-separated values* (csv) untuk setiap situsnya. Kemudian semua data yang ada pada setiap *file* cvs disatukan pada sebuah *file spreadsheet*. Terdapat 50 berita yang dihasilkan oleh proses ini yang kemudian akan diserahkan kepada pakar untuk diringkas secara manual.



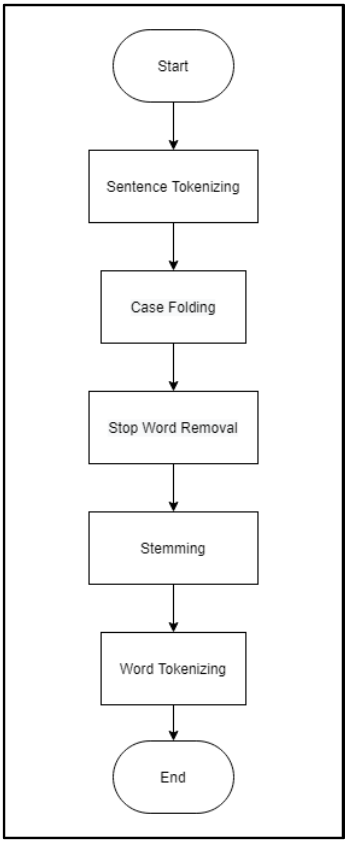
Title	Text	URL	Summary
Kejujuran akan Disabilitas Membuat	seorang wanita yang mengesankan banya	https://www.liputan6.com/disabilitas/read/4453650/kejujuran-akan-disabilitas-membu	text: Seorang wanita yang mengesankan banyak orang di The Bachelor dan menjadi kon pertama yang merupakan seorang tunarungu. Abigail lebih muda empat tahun (kini usia tahun) dari matt (29 tahun). Abigail lulus dari Linfield College pada 2017 dan bekerja di bidang keuangan. Saat ini bek sebagai analis keuangan klien di Opus Agency, di LinkedIn. Keterbukaan Abigail terkait disabilitasnya di sesi pertemuan pertama dengan cepat membuatnya menjadi favorit penggemar dan meningkatkan kesan pertama bagi Matt. Summary Correct: [1, 5]
Perjalanan Bima Kurniawan, Penyan	memiliki hambatan penglihatan atau tuna	https://www.liputan6.com/disabilitas/read/4506341/perjalanan-bima-kurniawan-penya	text: Memiliki hambatan penglihatan atau tunanetra tak menghentikan langkah Bima Ku
Manfaat Gabung Komunitas Anak Pe	jakarta memiliki anak dengan kelainan at	https://www.liputan6.com/disabilitas/read/4490062/manfaat-gabung-komunitas-anak-peny	Text: Angelman Syndrome adalah kelainan genetik di kromosom 15, di mana pada krom 15 ibu ada delesi atau hilangnya sebagian kromosom. Kelainan ini terbilang sangat langk memicu terjadinya disabilitas pada anak. Menurut Rani, hingga kini ia hanya bertemu 8 anak dengan sindrom yang sama di Indone Bahkan, komunitasnya pun belum ada. Guna mendapatkan dukungan dan berbagai pembelajaran, Rani memutuskan bergabung komunitas kelainan langka indonesia atau IRD. Dengan mengikuti komunitas, ia mengak sangat terbantu untuk mengetahui cara mengurus sang anak, Faustine Pitra Shabira yang disapa Utin. Summary correct: [4,5].
			Text: Penyanyi Yura Yunita memiliki pengalaman sendiri berkomunikasi dengan Tuli. Menurutnya, berbicara dengan teman Tuli ibarat bertemu dengan teman-te negara lain. Meskipun tak bisa bahasa asing, misalnya, teman saling mengerti d bahasa isyarat.

Gambar 3.7 Bentuk *Dataset Scraping*

3.4 Text Preprocessing

Gambar 3.8 merupakan *flowchart* dari tahapan *text preprocessing*. Tahap *text preprocessing* dilakukan untuk mempersiapkan data untuk diolah oleh metode LSA. Pada proses *text preprocessing*, pertama-tama dilakukan *sentence tokenizing* dengan menggunakan kelas sent_tokenize yang ada pada *library* nltk. Setelah itu,

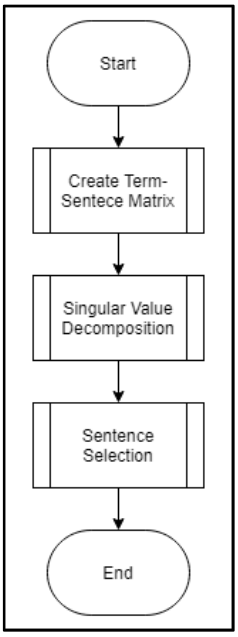
semua kalimat diubah menjadi huruf kecil (*case folding*) dengan menggunakan fungsi *lower*. Kemudian dilakukan *stop word removal* dengan menggunakan *library* Sastrawi. Kata-kata yang dihilangkan dipilih dari kamus *stop words* yang disediakan oleh *library* Sastrawi. Setelah itu, dilakukan *stemming* untuk mencari kata dasar dengan menghilangkan imbuhan dari kata-kata yang ada. Proses ini dilakukan dengan menggunakan fungsi *stem* yang ada pada *library* Sastrawi. Untuk menggunakan fungsi *stem* diperlukan sebuah *stemmer* yang dibuat dengan menggunakan kelas StemmerFactory dan kemudian menggunakan fungsi create_stemmer. Kemudian dilakukan *word tokenizing* dengan menggunakan kelas word_tokenize yang ada pada *library* nltk.



Gambar 3.8 *Flowchart Text Preprocessing*

3.5 Latent Semantic Analysis

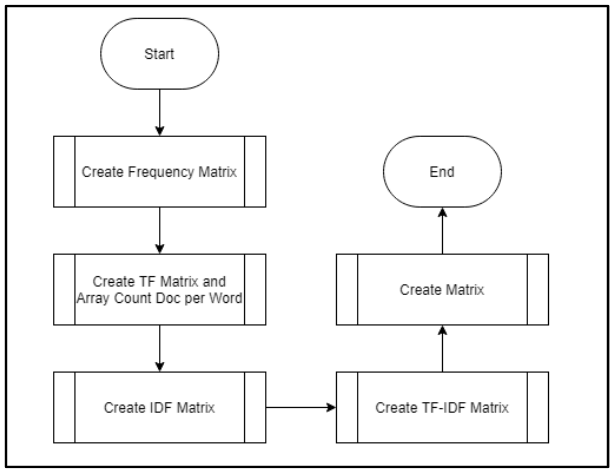
Setelah dilakukan tahap *text preprocessing*, dilakukanlah pembobotan dan pemilihan kalimat menggunakan metode LSA. Metode ini dibagi menjadi 3 (tiga) langkah utama, yaitu *create term-sentence matrix*, singular value decomposition, dan *sentence selection*. *Flowchart* dari LSA dapat dilihat pada Gambar 3.9.



Gambar 3.9 *Flowchart Latent Semantic Analysis*

3.5.1 Create Term-Sentence Matrix

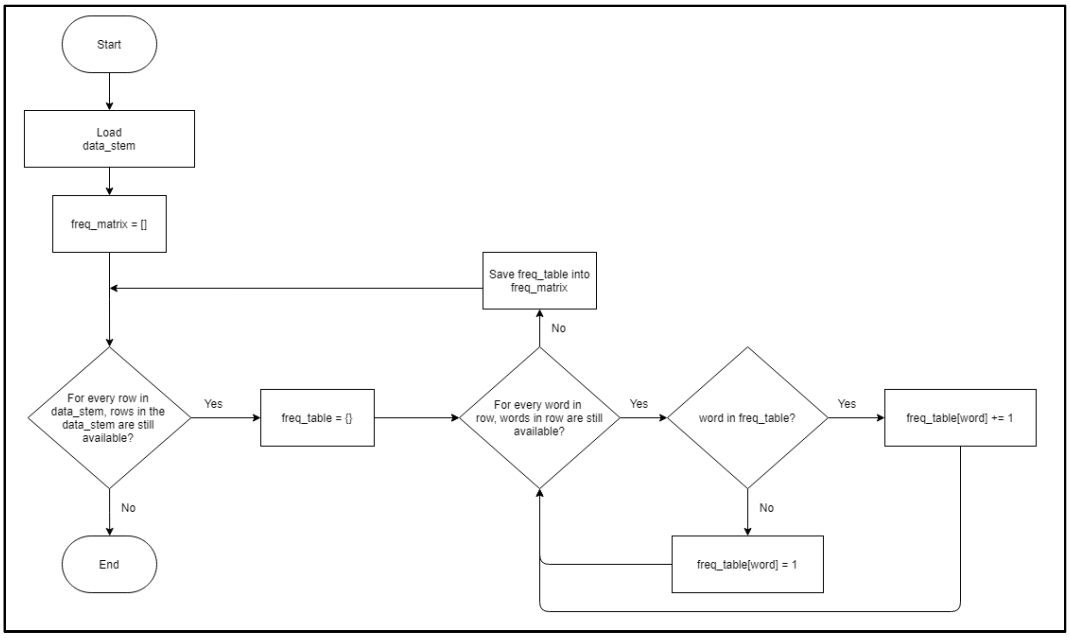
Gambar 3.10 merupakan *flowchart* dari tahapan *create term-sentence matrix*. Pada tahap ini dilakukan pembobotan terhadap data yang telah diproses sebelumnya dan kemudian hasil pembobotan tersebut diubah menjadi matriks. Pembobotan dilakukan dengan menggunakan persamaan TF-IDF. Tahap *create term-sentence matrix* dibagi menjadi lima bagian utama, yaitu *create frequency matrix*, *create* TF *matrix* dan *array count doc per word*, *create* IDF *matrix*, *create* TF-IDF *matrix*, dan *create matrix*.



Gambar 3.10 *Flowchart Create Term-Sentence Matrix*

A. Create Frequency Matrix

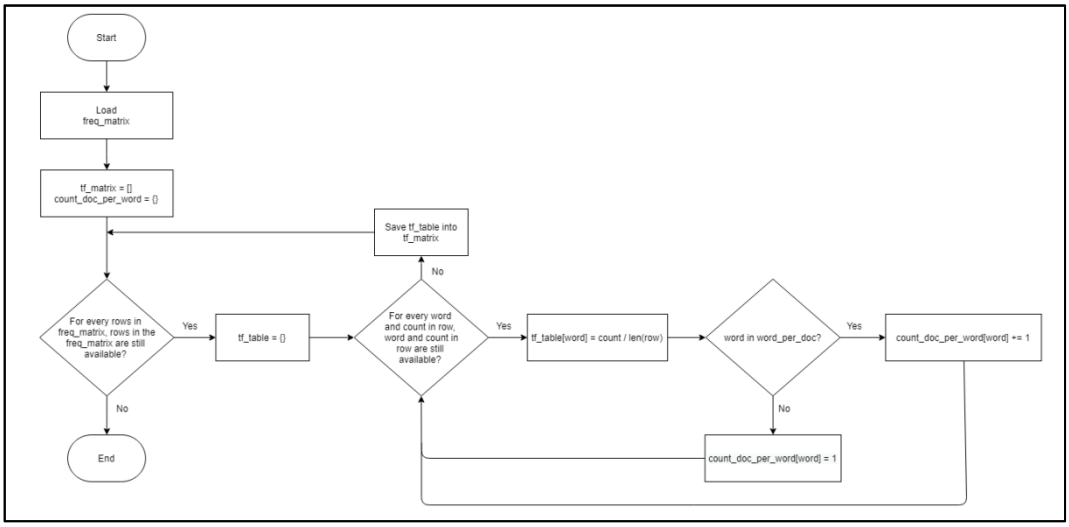
Pada bagian ini dibuat matriks *frequency* yang setiap barisnya berisikan jumlah suatu kata dalam satu kalimat. *Flowchart* dari pembuatan matriks frekuensi dapat dilihat pada Gambar 3.11.



Gambar 3.11 *Flowchart Create Frequency Matrix*

B. Create TF Matrix dan Array Count Doc per Word

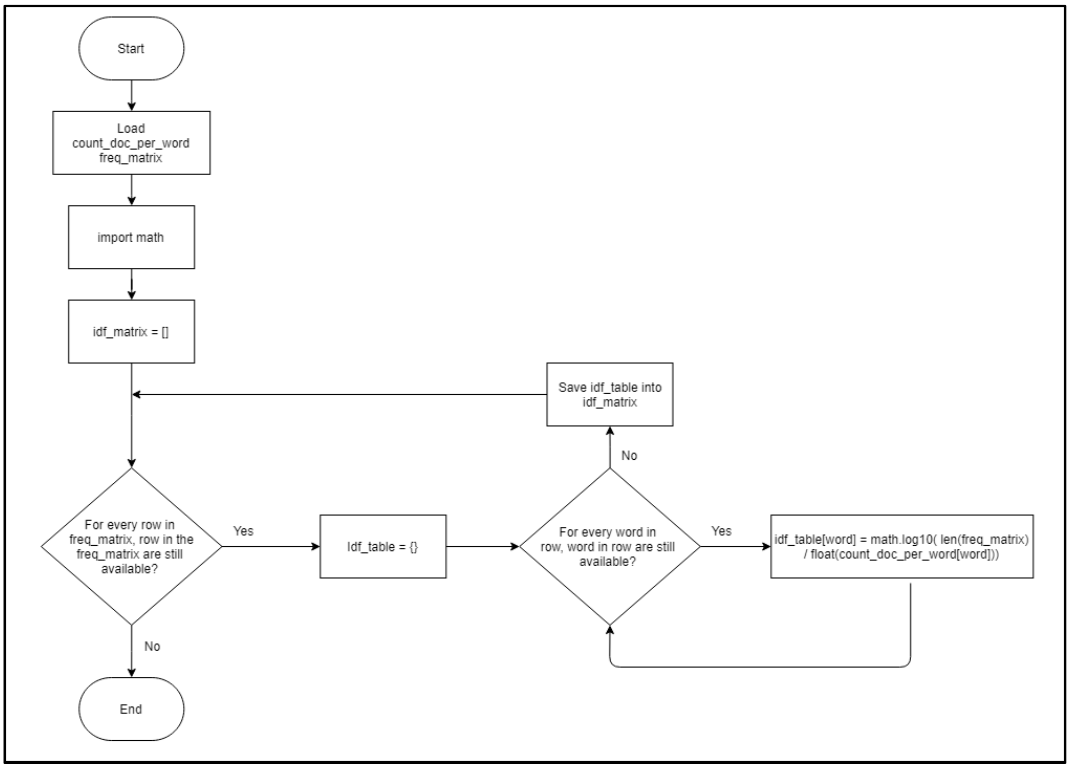
Pada bagian ini dibuat matriks TF menggunakan persamaan 2.1 dan *array count doc per word* yang merupakan *array* yang menampung jumlah kalimat yang mengandung kata ke-i. *Array* ini akan digunakan pada perhitungan IDF. *Flowchart create* TF *matrix* dan *array count doc per word* dapat dilihat pada Gambar 3.12.



Gambar 3.12 *Flowchart Create* TF *Matrix* dan *Array Count Doc per Word*

C. Create IDF Matrix

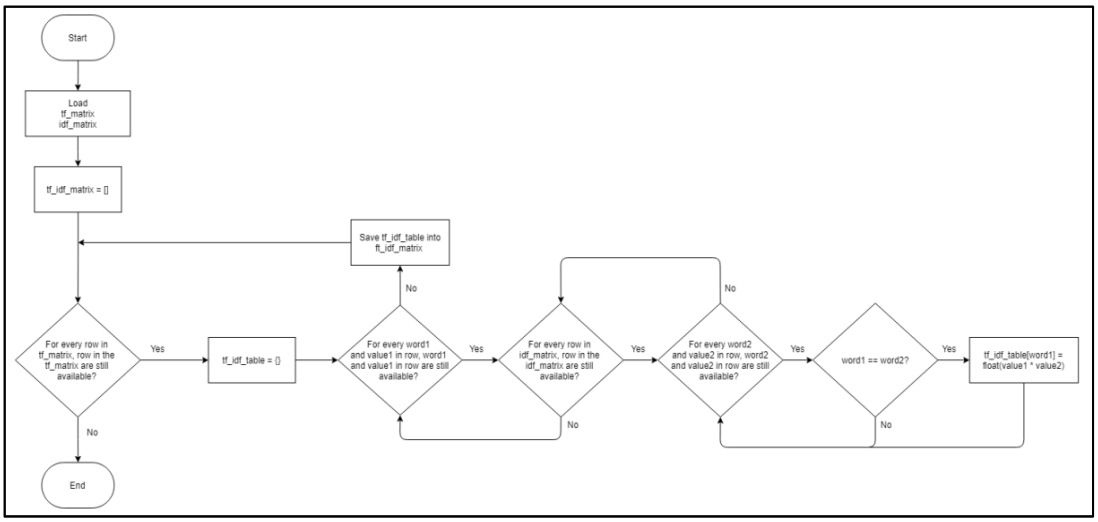
Pada bagian ini dilakukan pembuatan matriks IDF dengan menggunakan persamaan 2.2 serta bantuan dari *library math*. *Flowchart create* IDF *matrix* dapat dilihat pada Gambar 3.13.



Gambar 3.13 *Flowchart Create* IDF *Matrix*

D. Create TF-IDF Matrix

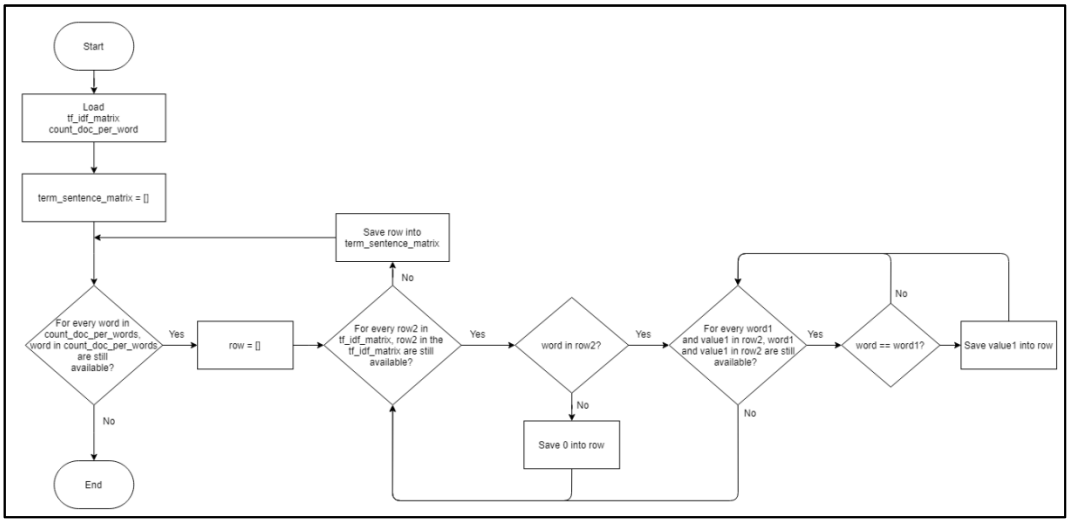
Pada bagian ini dibuat matriks TF-IDF dengan menggunakan persamaan 2.3. *Flowchart create* TF-IDF *matrix* dapat dilihat pada Gambar 3.14.



Gambar 3.14 *Flowchart Create* TF-IDF *Matrix*

E. Create Matrix

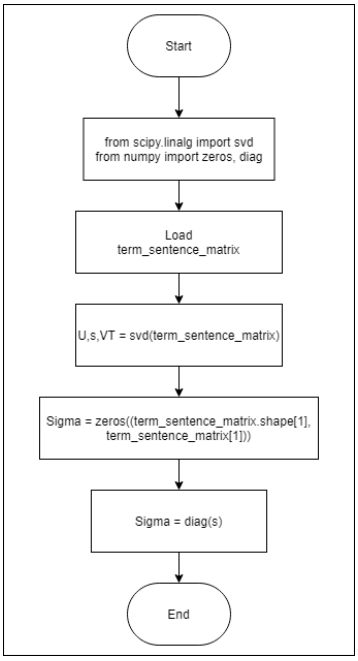
Pada bagian ini dibuat sebuah matriks berdasarkan pembobotan dari matriks TF-IDF. Bobot yang didapatkan menunjukkan seberapa besar kata tersebut mewakili kalimatnya. Bobot tersebut diubah menjadi sebuah matriks yang di mana barisnya merupakan *term* dan kolomnya merupakan *sentence*. *Flowchart create matrix* dapat dilihat pada Gambar 3.15.



Gambar 3.15 *Flowchart Create Matrix*

3.5.2 Singular Value Decomposition

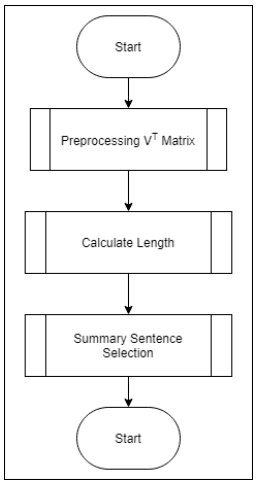
Tahap ini dilakukan untuk mendapatkan nilai dari matriks $\sum$ (sigma) dan V^T untuk digunakan pada tahap *sentence selection*. Nilai matriks tersebut didapatkan dengan menggunakan kelas svd yang ada pada *library* scipy. Kelas svd digunakan untuk menghasilkan nilai U, s, dan V^T. Matriks $\sum$ merupakan matriks diagonal dari nilai s yang didapatkan dari kelas svd dengan bantuan kelas zeros dan diag yang ada pada *library* numpy. *Flowchart singular value decomposition* dapat dilihat pada Gambar 3.16.



Gambar 3.16 *Flowchart Singular Value Decomposition* (SVD)

3.5.3 Sentence Selection

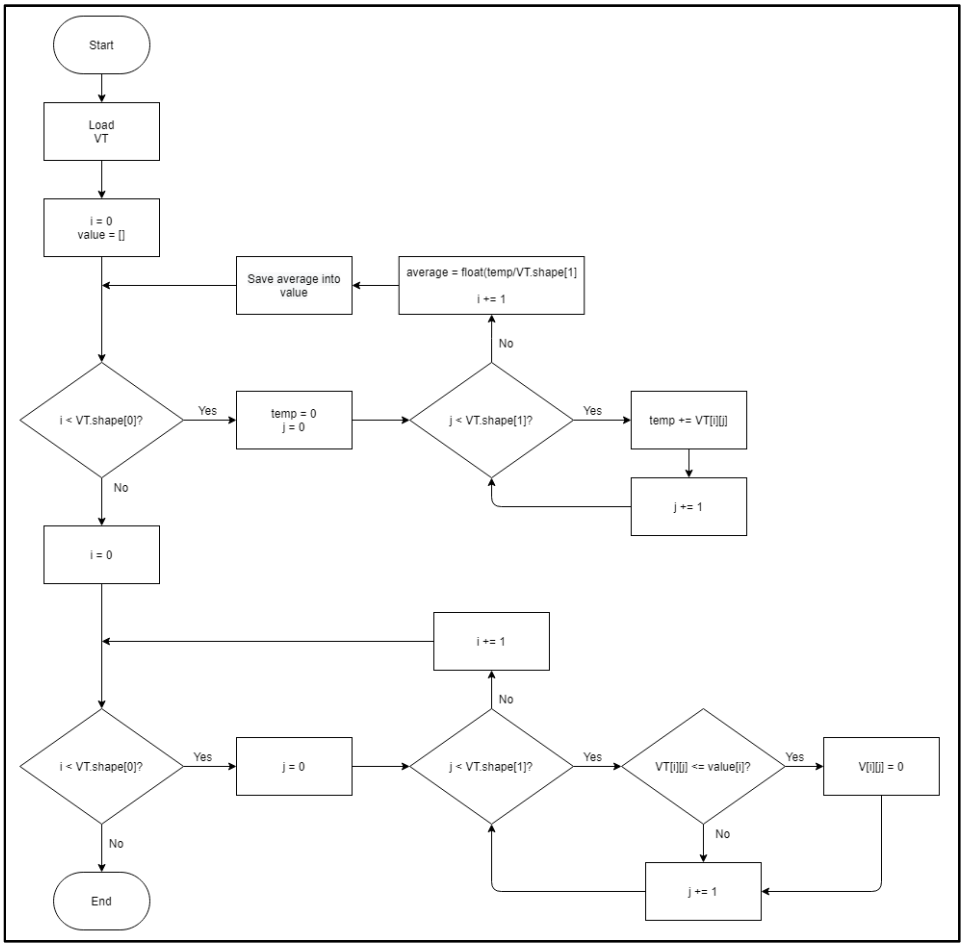
Gambar 3.17 merupakan *flowchart* dari *sentence selection*. Tahap ini dilakukan untuk memilih kalimat-kalimat penting untuk menghasilkan ringkasan. Proses tahap ini dibagi menjadi tiga bagian utama yaitu *preprocessing* V^T *matrix*, *calculate length*, dan *summary sentence selection*.



Gambar 3.17 *Flowchart Sentence Selection*

A. Preprocessing V^T Matrix

Proses preprocessing ini dilakukan dengan menghitung rata-rata dari setiap baris pada matriks V^T dan mengisis nilai dari sel V^T dengan nol (0), jika nilai tersebut kurang dari atau sama dengan (≤) rata-rata yang telah dihitung. Proses ini dilakukan untuk menghilangkan *noise* untuk setiap konsepnya. *Flowchart preporcessing* V^T *matrix* dapat dilihat pada Gambar 3.18.

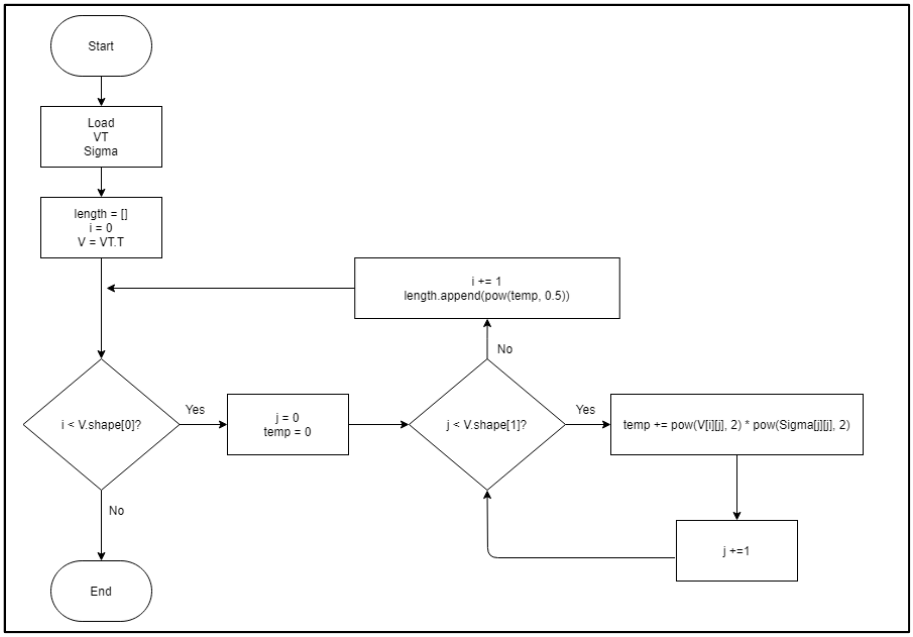


Gambar 3.18 *Flowchart Preprocessing* V^T *Matrix*

B. Calculate Length

Proses *calculate length* dilakukan untuk menemukan nilai dari setiap kalimat agar dapat terlihat kalimat mana yang lebih penting. Pada proses ini

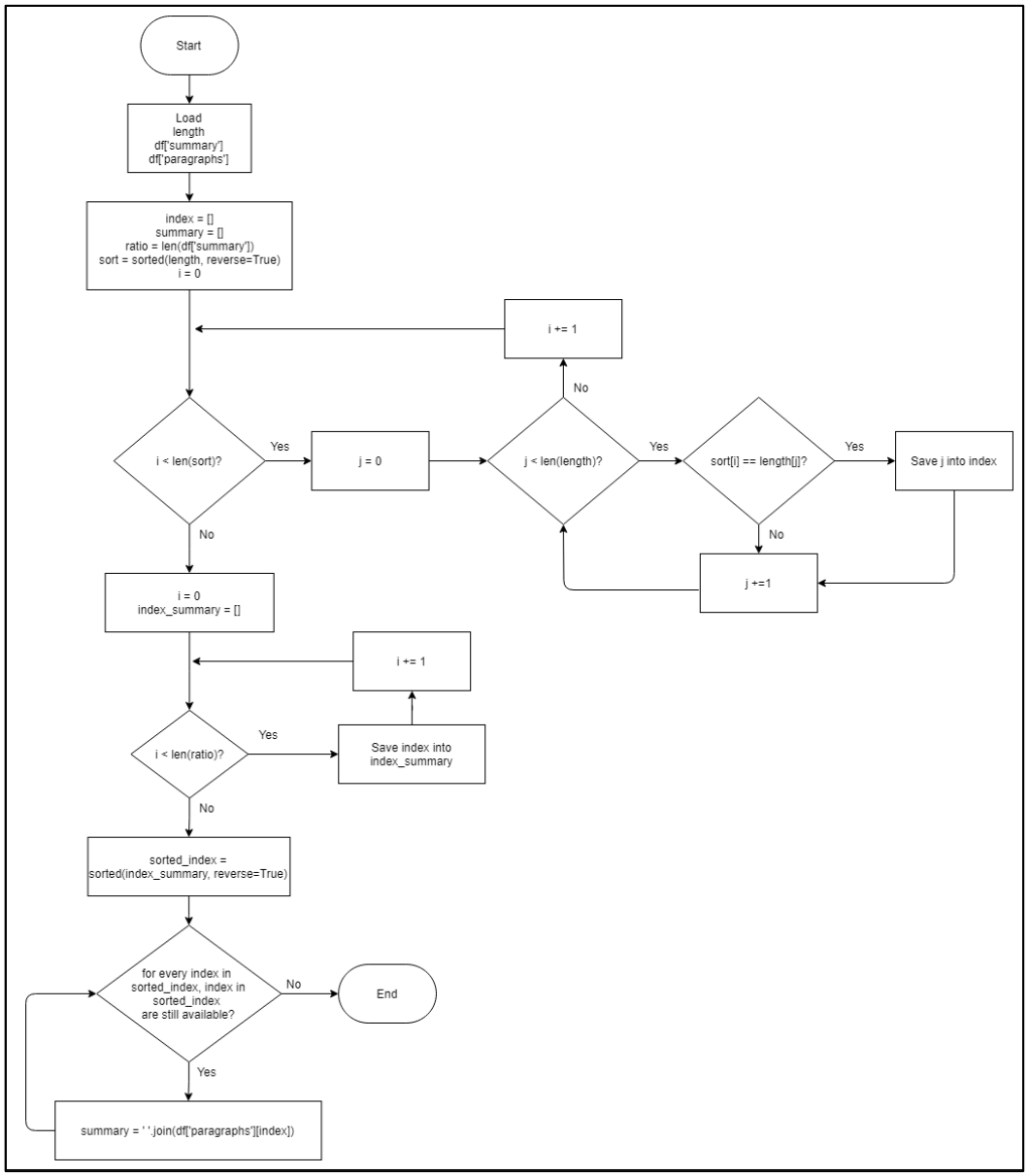
dihitung menggunakan persamaan 2.5. *Flowchart calculate length* ini dapat dilihat pada Gambar 3.19.



Gambar 3.19 *Flowchart Calculate Length*

C. Summary Sentence Selection

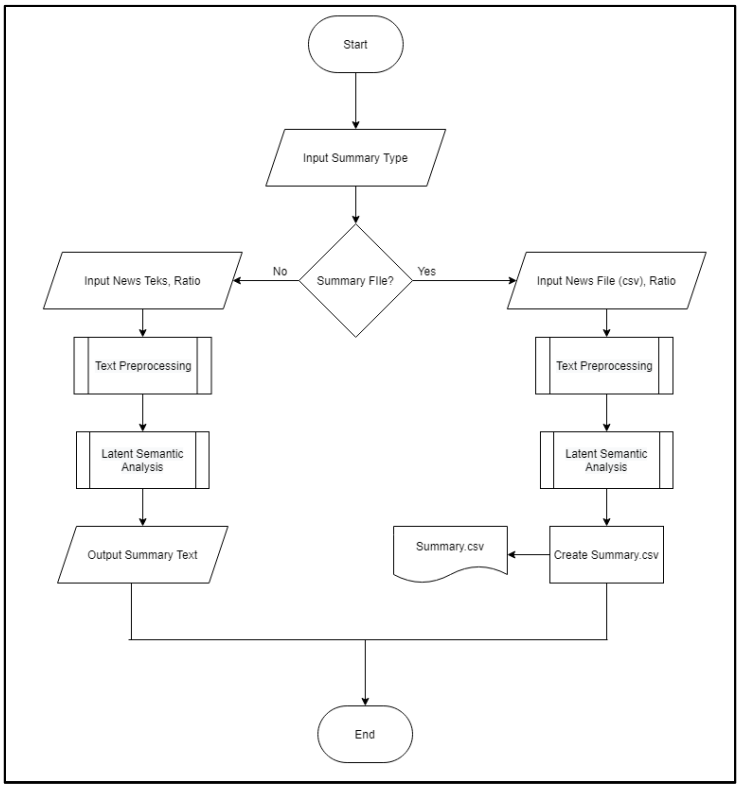
Proses pemilihan ini dilakukan dengan memilih nilai *length* tertinggi yang telah dihitung. Proses ini dilakukan untuk memilih kalimat yang dijadikan sebagai ringkasan. Panjang ringkasan dipilih berdasarkan panjang ringkasan yang ada dalam *dataset* IndoSum. *Flowchart summary sentence selection* dapat dilihat pada Gambar 3.20.



Gambar 3.20 *Flowchart Summary Sentence Selection*

3.6 Perancangan Aplikasi

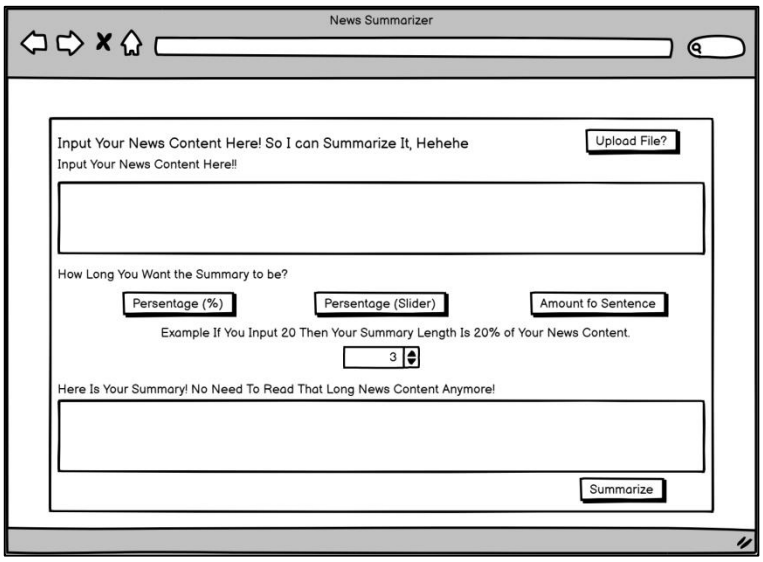
Proses perancangan aplikasi dibuat untuk menentukan alur kerja dari aplikasi *web* yang akan dibangun. Aplikasi memiliki dua fitur utama, yaitu meringkas teks berita yang dimasukkan oleh pengguna dan meringkas *file* csv yang di-*upload* oleh pengguna. *Flowchart* aplikasi *web* dilihat pada Gambar 3.21.



Gambar 3.21 *Flowchart* Aplikasi *Web*

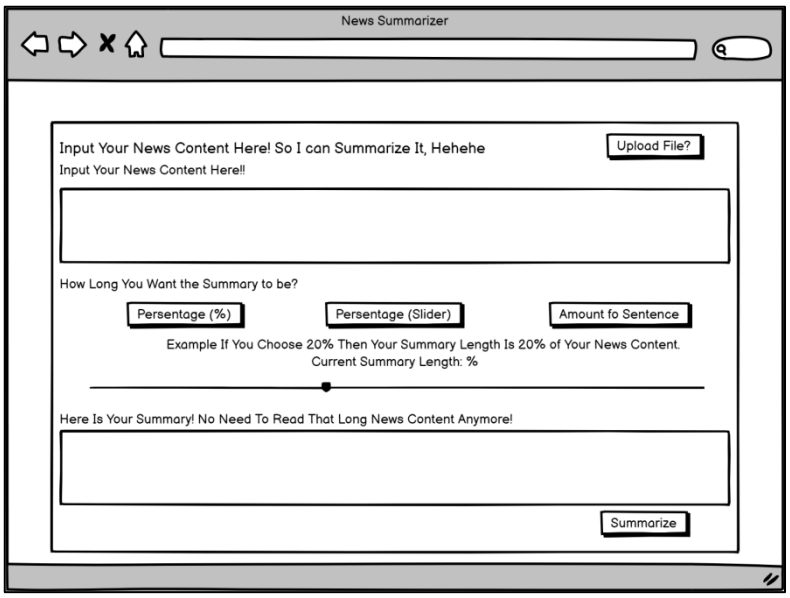
Fitur meringkas teks berita dapat diakses pada halaman awal aplikasi. Pada halaman awal aplikasi disediakan sebuah *textarea* yang digunakan oleh pengguna untuk memasukan berita yang ingin diringkas. Selain itu, pengguna juga dapat memilih panjang ringakasan yang diinginkan. Terdapat dua cara yang dapat dilakukan pengguna untuk menentukan panjang ringkasan, yaitu dengan memasukkan nilai *percentage ratio* dan menentukan sendiri jumlah kalimat ringkasan. Pada cara *precentage ratio* terdapat dua tipe *input* yang dapat digunakan, yaitu dengan menggunakan memasukkan nilai 1 sampai 100 dan dengan menggunakan *slider*. Setelah menetukan panjang ringkasan yang diinginkan pengguna hanya perlu menekan tombol "*Summarize*" untuk meringkas

berita yang telah dimasukkan. Rancangan *interface* halaman awal dengan *percentage ratio* dapat dilihat pada Gambar 3.22.



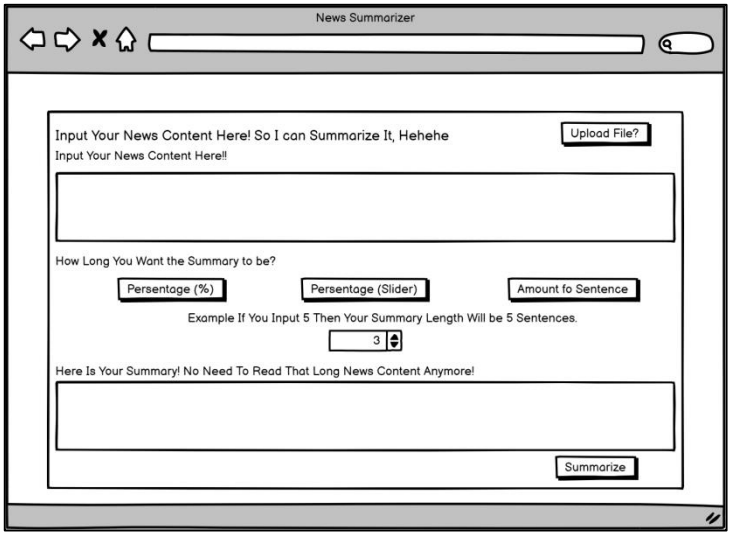
Gambar 3.22 Rancangan *Interface* Halaman Awal Dengan *Percentage Ratio*

Gambar 3.23 merupakan rancangan *interface* halaman awal dengan *percentage slider*.



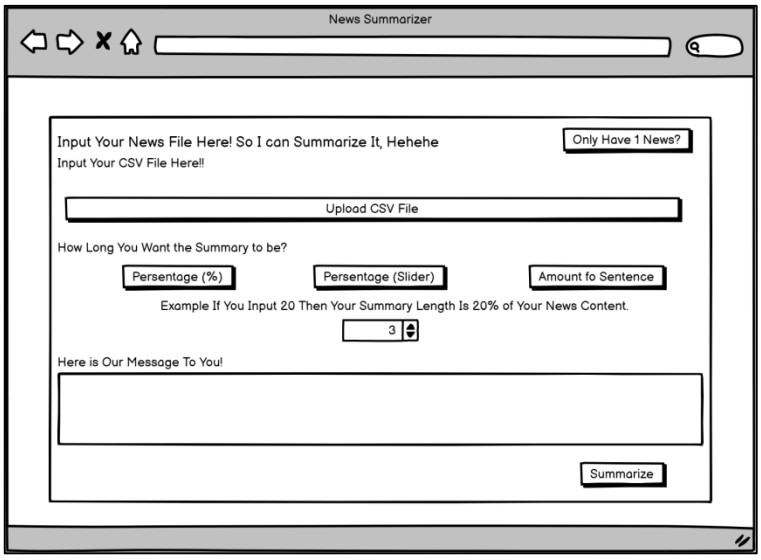
Gambar 3.23 Rancangan *Interface* Halaman Awal Dengan *Percentage Silder*

Rancangan *interface* halaman awal dengan panjang kalimat ringkasan dapat dilihat pada Gambar 3.24.



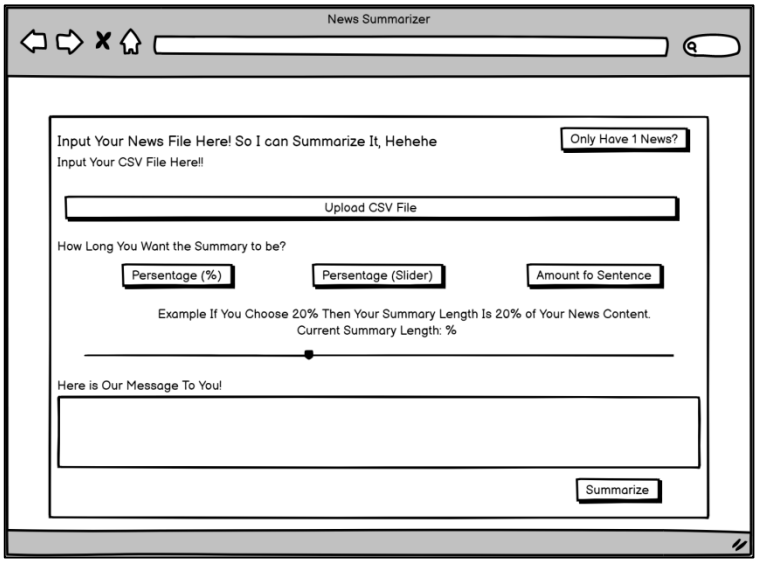
Gambar 3.24 Rancangan *Interface* Halaman Awal Dengan Panjang Kalimat Ringkasan

Halaman *upload file* dapat diakses dengan menekan tombol "*Upload File?*" yang ada pada halaman awal. Pada halaman ini, pengguna dapat meringkas *file* csv yang berisikan kumpulan artikel berita. Pengguna juga dapat menentukan panjang kalimat ringkasan yang diinginkan. Setelah menentukan panjang ringkasan, pengguna cukup menekan tombol "*Summarize*" dan aplikasi akan meringkas *file* tersebut dan membuat pengguna men-*download* sebuah *file* csv yang berisikan artikel berita dan ringkasannya. Rancangan interface halaman *upload file* dengan *percentage ratio* dapat dilihat pada Gambar 3.25.



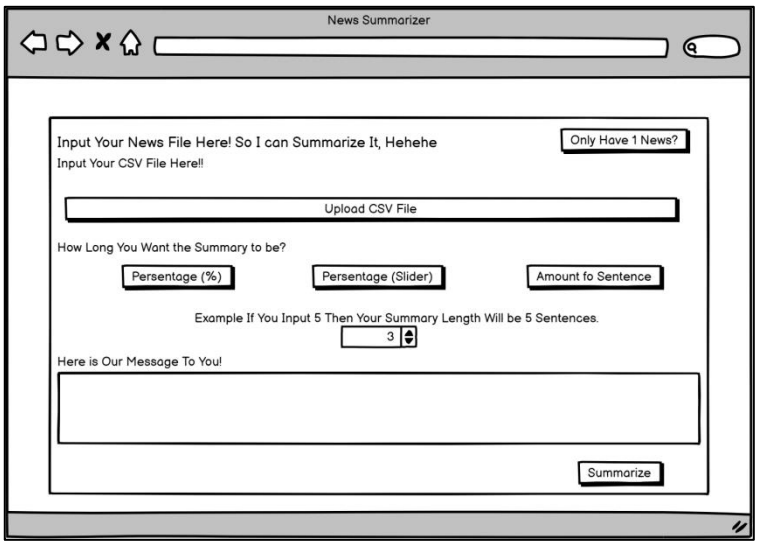
Gambar 3.25 Rancangan *Interface* Halaman *Upload File* Dengan *Percentage Ratio*

Gambar 3.26 merupakan rancangan *interface* halaman *upload file* dengan *percentage slider*.



Gambar 3.26 Rancangan *Interface* Halaman U*pload File* Dengan P*ercentage Slider*

Rancangan *interface* halaman *upload file* dengan panjang kalimat ringkasan dapat dilihat pada Gambar 3.37.



Gambar 3.27 Rancangan *Interface* Halaman *Upload File* Dengan Panjang Kalimat Ringkasan

3.7 Pembangunan Aplikasi

Pembangunan aplikasi bertujuan untuk membuat aplikasi berbasis web yang sesuai dengan rancangan aplikasi yang telah dibuat. Aplikasi dibuat menggunakan *framework* Flask berbasis Python dan di-*hosting* menggunakan *platform* Heroku.

3.8 Testing dan Evaluating

Proses *testing* dilakukan untuk *debugging* terhadap implementasi dari *Latent Semantic Analysis*, dan memperbaiki kode yang telah dibuat jika terdapat kesalahan. Proses *evaluating* yang dilakukan terbagi menjadi dua, yaitu menghitung nilai *cosine similarity*, *precision*, *recall* dan F1-*score* pada *dataset*

IndoSum dan menghitung nilai *cosine similarity*, *precision*, *recall* dan F1-*score* pada *dataset scraping*.

3.9 Penulisan Laporan

Penulisan laporan mencakup seluruh proses penelitian dari telaah literatur sampai dengan kesimpulan dan saran, dan ditulis secara terstruktur dan menurut kaidah laporan penelitian. Untuk penulisan laporan yang lebih baik dilakukan konsultasi dengan dosen pembimbing.