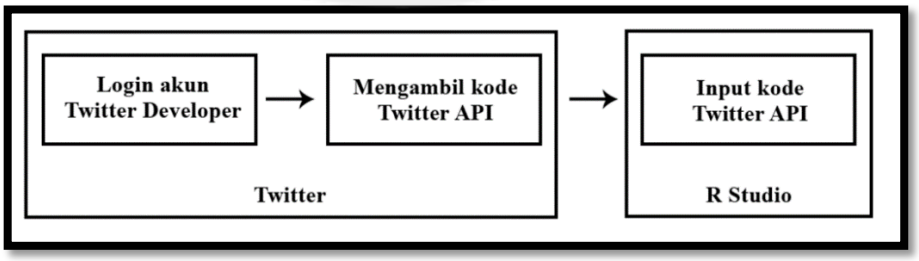
BAB III

METODOLOGI PENELITIAN

3.1 *Authentication*

Tahap *authentication* ini merupakan tahap integrasi antara media sosial Twitter dengan *software* R studio sebagai tempat dimana tahap berikutnya dilakukan, yaitu *data collecting.* Tahap *authentication* ini menggunakan API. API (*application programming interface*) memungkinkan pengguna untuk dapat mengintegrasikan dua bagian dari aplikasi atau dengan aplikasi yang berbeda dalam waktu yang bersamaan.

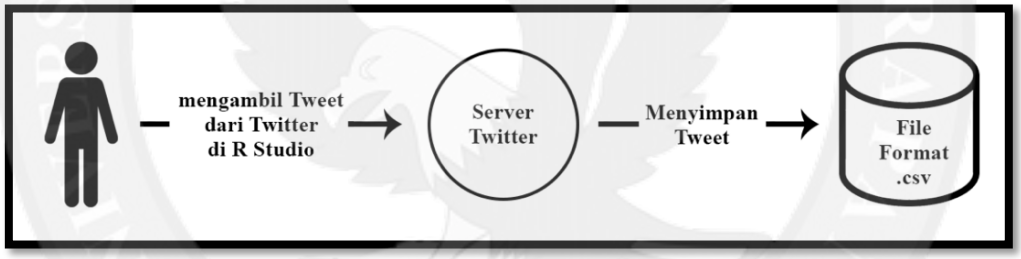
R Studio membutuhkan beberapa kode khusus untuk dapat menggunakan API yang disediakan oleh pihak twitter. Untuk mendapatkan kode-kode tersebut, maka terlebih dahulu harus mendaftar akun sebagai akun *developer*. Beberapa kode tersebut yaitu API *key,* API *secret, access token, access token secret.* Kode tersebut digunakan untuk proses integrasi antara Twitter API dengan *software* R Studio, dimana proses integrasi dilakukan dengan menggunakan *library* dengan fungsi *library(twitteR),* namun perlu untuk menginstall *package library* twitteR terlebih dahulu, setelah itu diakhiri dengan menggunakan fungsi *'setup_twitter_oauth'* di *software* R studio.



Gambar 3.1 Skema proses *authentication*

3.2 *Data Collecting*

Tahap *data collecting* ini merupakan tahap pengambilan data *tweet* dari media sosial twitter setelah integrasi dari *software* R studio dengan API Twitter pada tahap *authentication* selesai dilakukan. Tahap ini dilakukan secara *real time* dari twitter dengan menggunakan fungsi '*tweets'* pada *software* R Studio dan data-data yang telah diambil tersebut akan dimasukkan ke dalam satu *file.* Contoh penggunaan fungsi *'searchTwitter'* yaitu *'tweets <- searchTwitter('Tokopedia', n=10, lang='id')*'. Fungsi tersebut akan mengambil *tweets* dari media sosial Twitter dengan kata kunci 'Tokopedia' berjumlah 10 berbahasa Indonesia. Setelah data *tweets* yang dibutuhkan telah berhasil diambil, maka data-data *tweets* tersebut butuh sebuah *file* untuk menampung data-data *tweets* tersebut. Hal tersebut dapat dilakukan menggunakan fungsi '*write.csv'.*



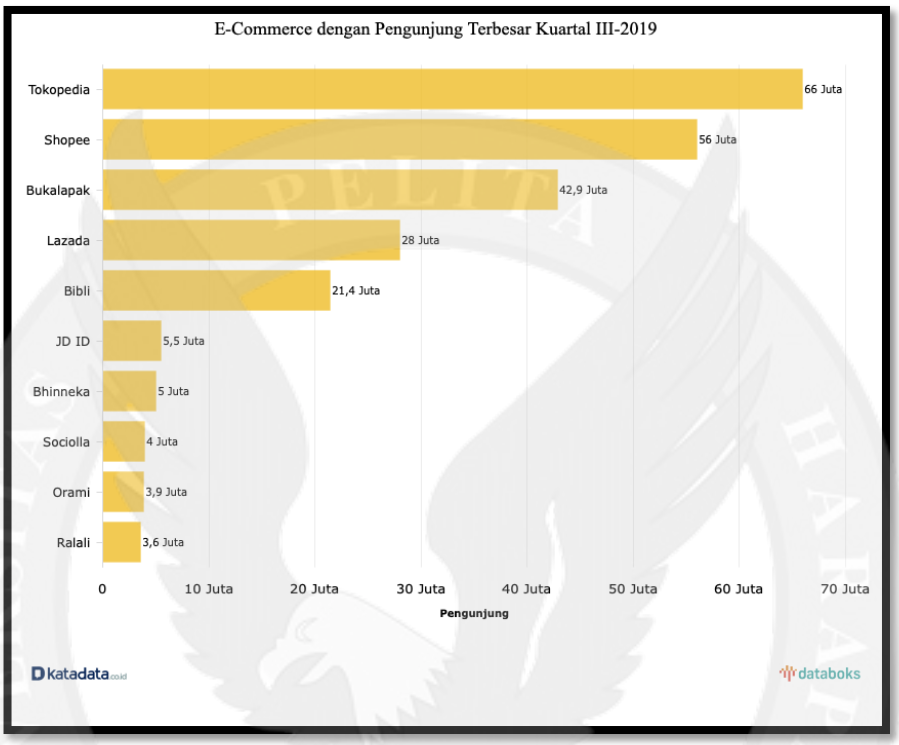
Gambar 3.2 Skema proses *data collecting*

Penelitian ini mengambil data *tweet* sebanyak 4500, dimana data ini terbagi menjadi beberapa bagian seperti yang ditunjukkan pada tabel 3.1

Tabel 3.1 Pembagian *data collecting*

Nama *Marketplace*	Jumlah *Tweets*
Tokopedia	1500 Tweets
Shopee	1500 Tweets
Bukalapak	1500 Tweets

Penelitian ini memilih ketiga *online marketplace* tersebut dengan alasan ketiga *online marketplace* tersebut memiliki jumlah pengunjung terbanyak di Indonesia menurut situs katadata [23].



Gambar 3.3 Grafik pengunjung terbesar *e-commerce* di Indonesia
Sumber: situs katadata [23]

3.3 *Text Preprocessing*

Tahap *text preprocessing* ini terdiri dari beberapa tahap antara lain:

3.3.1 Case Folding

Dalam *text preprocessing,* tahap *case folding* ini merupakan tahap dimana seluruh karakter huruf di dalam dokumen dijadikan *lowercase,* serta membuang seluruh huruf karakter selain huruf a sampai z. Tahap pertama yaitu menghilangkan '\n' pada data *tweets*. Setelah itu, seluruh *Uniform Resource Locator* (URL) akan dihilangkan menggunakan fungsi '*replace_html*'. Tahap selanjutnya yaitu seluruh *mention* dan juga *hashtag* yang ada di dalam dokumen juga akan dihapus dengan

menggunakan fungsi '*replace_tag*' dan '*replace_hash*', begitu juga dengan tanda baca pada dokumen yang akan dihapus dengan menggunakan fungsi *'strip'*. Fungsi '*strip*' juga akan mengubah seluruh dokumen menjadi *lower case.* Setelah itu, teks yang disingkat akan diubah menjadi teks yang tidak disingkat dengan menggunakan fungsi '*replace_Internet_slang*' menggunakan daftar *lexicon* bahasa Indonesia yang telah dibuat sebelumnya pada publikasi *Colloquial Indonesian Lexicon* [24].

3.3.2 Stemming

Tahap *stemming* merupakan tahap yang bertujuan untuk mengubah kata-kata yang terdapat dalam suatu dokumen ke kata-kata akarnya (*root word*). Pada proses *stemming* ini, akan menghilangkan imbuhan-imbuhan baik itu berupa prefiks, sufiks, maupun konfiks yang ada pada setiap kata yang ada pada dokumen. Tahap *stemming* pada R Studio menggunakan *library* yang telah disediakan. Tahap *stemming* menggunakan *package* katadasaR [25].

3.3.3 Stopwords Removing

Tahap *stopwords removing* merupakan tahap pengambilan kata – kata yang dianggap penting saja. Sebelum proses *stopwords removing* dilakukan, harus dibuat sebuah daftar *stopword* (*stoplist*). Jika ada kata-kata yang termasuk di dalam daftar *stoplist* yang telah dibuat sebelumnya, maka kata-kata tersebut akan dihapus dari deskripsi sehingga kata-kata yang tersisa di dalam deskripsi dianggap sebagai kata-kata yang mencirikan isi dari suatu dokumen atau *keywords* atau kata-kata yang dianggap penting saja. Hal ini dapat dilakukan dengan menggunakan fungsi *'removewords'*, dimana kata-kata yang telah dibuat di daftar *stopwords* akan dihilangkan dari dokumen. Daftar *stoplist* yang digunakan pada penelitian ini diambil dari jurnal Tala [26], dimana berjumlah 758 *stopwords* dan dikombinasikan dengan *stopwords* lainnya yang diambil pada situs github [27] sehingga berjumlah 833 *stopwords.* Contoh *stopwords* dapat dilihat pada tabel 3.2 [26] [27].

Tabel 3.2 Contoh *stopwords*

Nomor	*Stopwords*
1	ada
2	adanya
3	adalah
4	adapun
5	agak
6	agaknya
7	agar
8	akan
9	akankah

3.4 *Word Cloud Creating*

Tahap *word cloud creating* merupakan tahap dimana teks yang sudah dibersihkan dan terstruktur divisualisasikan menggunakan *package* yang tersedia pada *software* R studio. Metode ini dapat dikatakan cukup terkenal dalam *text mining* karena mudah dipahami serta visualisasi yang menarik. Ukuran kata-kata dalam *word cloud* dipengaruhi oleh seberapa sering kata tersebut muncul dalam dokumen. Semakin sering suatu kata muncul dalam dokumen, maka ukuran kata tersebut akan divisualisasikan semakin besar juga pada *word cloud.* Tahap ini menggunakan *library* '*wordcloud*' pada *software* R studio.

3.5 *Lexicon*

Tahap *lexicon* merupakan metode yang biasa digunakan untuk menggali atau mengetahui bagaimana pandangan ataupun opini masyarakat terhadap sesuatu. Pada tahap *lexicon* ini, setiap *tweets* akan diberikan nilai, dapat dilihat pada tabel 3.3 berikut:

Tabel 3.3 Penilaian sentimen

Sentimen	Nilai
Positif	1
Netral	0
Negatif	-1

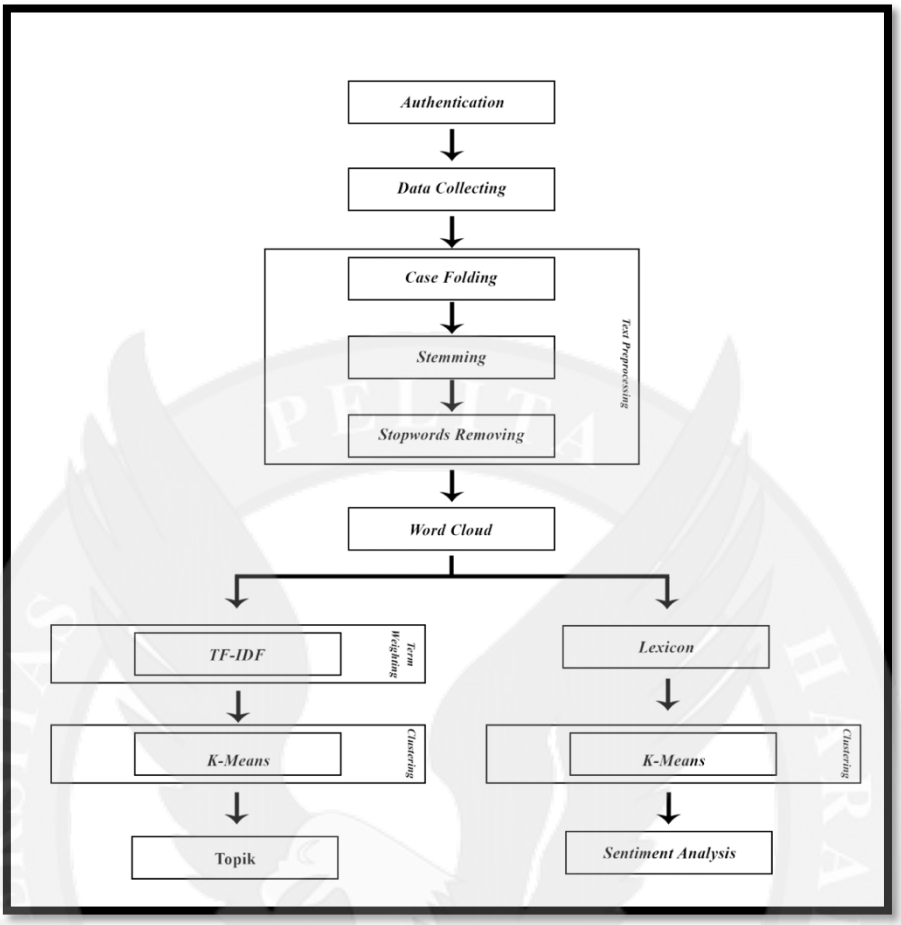
Konsep yang digunakan pada metode *lexicon* penelitian ini yaitu memberikan nilai pada setiap *tweets* berdasarkan berapa banyak *tweets* tersebut mengandung kata positif atau negatif menggunakan list positif negatif yang telah dibuat sebelumnya pada *Liu's opinion words list* dengan modifikasi dan terjemahan ke bahasa Indonesia [28][29]. Setiap *tweets* bisa saja memiliki lebih dari satu kata negatif atau positif sehingga nilai *tweets* bisa saja kurang dari -1 atau lebih dari 1. Setelah masing-masing *tweets* mempunyai nilai, selanjutnya akan divisualisasikan dengan menggunakan histogram untuk melihat perbandingan sentimen *tweets* dan menyimpulkan apakah hasil *lexicon* merupakan positif, netral, atau negatif.

3.6 *Term Weighting*

Tahap *term weighting* merupakan tahap dimana pengolahan serta analisis dari data-data *tweets* yang telah melewati tahap *text preprocessing* dilakukan. *Term weighting* ini menggunakan metode *Term Frequency – Inverse Document Frequency* (TF-IDF). Metode *Term Frequency – Inverse Document Frequency* (TF-IDF) dianggap cocok dalam penelitian ini karena frekuensi metode ini dapat menunjukkan seberapa pentingnya kata-kata yang ada di dalam dokumen.

3.7 *Clustering*

Tahap *clustering* merupakan tahap pengelompokan secara otomatis. Tahap *clustering* ini menggunakan metode algoritma *k-means clustering. K-means clustering* merupakan metode yang cukup populer saat dalam hal mendapatkan deskripsi-deskripsi dari sekumpulan data dengan cara mengungkapkan kecenderungan setiap individu data untuk berkelompok dengan individu-individu data lainnya. Tahap pertama dalam *clustering* ini yaitu, menentukan jumlah *cluster.* Setelah itu, tahap berikutnya yaitu alokasikan data ke dalam *cluster* secara acak, dilanjutkan dengan menghitung *centroid /* rata-rata data yang ada di masing-masing *cluster*, diakhiri dengan mengalokasikan masing-masing data ke *centroid* terdekat. Tahap ini dapat dilakukan di *software* R studio.



Gambar 3.4 Metodologi yang digunakan