

BAB II

TINJAUAN PUSTAKA

2.1 Cyberbullying

Cyberbullying merupakan fenomena dan bentuk baru dari perilaku bullying, yaitu ketika seseorang berulang kali mengolok-olok orang lain secara online atau berulang kali mengganggu orang lain melalui email, pesan teks atau memposting sesuatu tentang orang lain yang tidak disukai (Suparyanto dan Rosad (2015 2020)). *Cyberbullying* adalah tindakan yang menggunakan teknologi informasi dan komunikasi untuk keperluan yang disengaja, dilakukan terus menerus, dengan tujuan untuk merugikan orang lain dengan cara mengintimidasi, mengancam, menyakiti/menghina harga diri orang lain, hingga menimbulkan permusuhan oleh seorang individu atau kelompok penggunaan teknologi komunikasi dalam penggunaan layanan internet dan teknologi mobile seperti halaman web dan grup diskusi serta telepon selular dan pesan teks (SMS).(Wiwit, 2015)

2.2 Algoritma Squeezer

Squeezer merupakan sebuah algoritma yang digunakan untuk mengelompokkan data (*clustering*) sekumpulan data bertipe kategorikal. Ide dasar dari algoritma tersebut sangat sederhana. *Squeezer* secara berulang membaca tiap pasangan data(*tuple*) dari kumpulan data satu persatu. Saat pasangan data pertama dibaca, akan dibuat kelompok data (*cluster*) baru. Pasangan data berikutnya dimasukkan kedalam kelompok data yang sudah ada atau ditolak oleh semua kelompok yang ada sehingga membentuk kelompok baru berdasarkan fungsi kemiripan yang diberikan antara kelompok dengan pasangan data.

Algoritma Squeezer (D,s)

Begin

```
while (D has unread tuple){
```

```
tuple = getCurrentTuple(D)
```

```

if(tuple.tid == 1){
addNewClusterStructure(tuple, tid)}
else{
for each existing cluster C
simComputation(C, tuple)
get the max value of similarity : sim_max
get the corresponding Cluster Index: index
if sim_max >= s
addTupleToCluster(tuple, index)
else addNewClusterStructure(tuple, tid)}
}
handleOutliers()
outputClusteringResult()
End

```

2.3 Sistem operasi

Sistem Operasi merupakan sebuah penghubung antara pengguna mesin dengan perangkat keras yang dimiliki mesin tersebut. Sebelum ada Sistem Operasi, orang hanya menggunakan komputer dengan menggunakan sinyal analog dan digital (Abas ali and Dony, 2019). Seiring dengan berkembangnya pengetahuan dan teknologi, pada saat ini terdapat berbagai Sistem Operasi dengan keunggulan masing-masing. Sistem Operasi bertindak sebagai antarmuka antara program aplikasi dengan perangkat keras komputer, level dari pengguna setiap lapisan juga berbeda-beda. Program aplikasi hanya digunakan oleh pemakai terakhir (*End user*), sedangkan Sistem Operasi dan perangkat keras merupakan tugas pemrogram dan pendesain Sistem Operasi tersebut.

2.4 Android

Android merupakan suatu sistem operasi yang digunakan perangkat mobile yang dengan berbasis linux serta mencakup *middleware*, aplikasi dan sistem operasi (N. Safaat 2018). Android adalah sistem operasi berbasis *Linux* yang dirancang untuk

perangkat bergerak layar sentuh seperti telepon pintar dan komputer tablet. (Kusniyati and Pangondian Sitanggang 2016). Sebagai *open source* dan bebas dalam memodifikasi, di dalam Android tidak memiliki ketentuan yang tetap di dalam konfigurasi *hardware* dan *software*.

2.5 Aplikasi

Aplikasi dapat diartikan sebagai suatu program berbentuk perangkat lunak yang berjalan pada suatu sistem tertentu yang berguna untuk membantu berbagai kegiatan yang dilakukan oleh manusia. (Huda and Priyatna, 2019). Aplikasi merupakan program yang berisi perintah untuk melakukan olah data, secara umum aplikasi adalah suatu proses dari cara manual yang dipindahkan ke dalam komputer dengan membuat sistem atau program agar data diolah lebih berdaya guna secara optimal.

2.6 Perangkat Lunak Yang digunakan

2.6.1 *Application Programming Interfaces (API's)*

Application Programming Interface (API) bukan hanya signature yang sederhana, dan satu *set class* atau *method* fungsi API Menyediakan perangkat untuk mengatasi ketidakpahaman dalam membangun *software* yang berukuran besar, mulai dari yang sederhana sampai kompleks serta komponen yang sulit dipahami (Alfeno and Devi, 2017)

2.6.2 Android Studio

Android studio adalah IDE (*Integrated Development Environment*) resmi untuk pengembangan aplikasi android dan bersifat *open source* atau gratis (Juansyah, 2015). Android Studio menggantikan *Eclipse* sebagai IDE resmi untuk mengembangkan aplikasi Android. (Yudha Yudhanto and Ardhi Wijayanto, 2018) Android studio ini bersifat *free* dibawah *Apache License 2.0*. Android Studio awalnya bermula dengan versi 0.1 pada mei 2013, kemudian dibuat versi beta 0.8 yang dirilis 2014. Berbasiskan JetBrains IntelliJ IDEA, Studio di desain khusus untuk Android Development. Android studio memiliki fitur:

- a. Projek berbasis pada *Gradle Build*

- b. *Refactory* dan pembenahan *bug* yang cepat
- c. *Tools* baru yang bernama "*Lint*" dikalim dapat memonitor kecepatan, kegunaan, serta kompetibelitas aplikasi dengan cepat
- d. Mendukung *Proguard And App-signing* untuk keamanan.
- e. Memiliki *GUI* aplikasi Android lebih mudah. didukung oleh *Google Cloud Platfrom* untuk setiap aplikasi yang dikembangkan.

2.6.3 Java

Bahasa pemrograman Java merupakan salah satu dari sekian banyak bahasa pemrograman yang dapat dijalankan di berbagai sistem operasi termasuk telepon genggam. Bahasa pemrograman ini pertama kali dibuat oleh James Goslingsaat masih bergabung Sun Microsystem. Bahasa pemrograman ini merupakan pengembangan C++, saat ini Javamerupakan bahasa pemrograman yang paling populer digunakan, dan secara luas dimanfaatkan dalam pengembangan berbagai jenis perangkat lunak aplikasi ataupun aplikasi berbasis *web*. Kelebihan java dari bahasa pemrograman yang lain adalah bisa dijalankan di berbagai jenis sistem operasi sehingga dikenal juga bahasa pemrograman multiplatform, bersifat pemproframan berorientasi object (PBO), memiliki library yang lengkap(Harumy, T.H.F., Julham Sitorus 2018)

2.6.4 Database

Database merupakan kumpulan informasi yang disimpan di dalam komputer yang secara sistematik agar dapat diperiksa menggunakan suatu program komputer. Secara konsep Database berupa kumpulan dari data-data yang membentuk file yang antara satu dan lainnya terhubung dengan tata cara tertentu untuk membentuk data baru. Sistem basis data merupakan sistem yang terdiri dari atas kumpulan tabel yang saling berhubungan (dalam sebuah basis data disebut sistem komputer) dan sekumpulan program (yang biasa disebut DBMS atau *DataBase Management System*) yang memungkinkan beberapa pemakai atau program lain untuk mengakses dan memanipulasi tabel-tabel data tersebut.(Yani, Saputra, and Jurnal, 2018)

2..6.5 Visual Studio Code

Visual Studio Code merupakan sebuah kode editor yang dikembangkan oleh *microsoft* untuk *windows*, *linux* dan *macOs*. Program ini termasuk dukungan untuk debugging, kontrol git yang tertanam dan GitHub, penyorotan sintaktis, penyelesaian kode cerdas, snippet dan *refactoring* kode, ini sangat dapat disesuaikan, memungkinkan pengguna untuk mengubah tema, penyintasan keyboard, preferensi dan meninstall ekstensi yang menambah fungsional tambahan. memudahkan developer atau programmer dalam membangun sebuah aplikasi (Herdiana and Azhari, 2021)

2.6.6 NoSQL

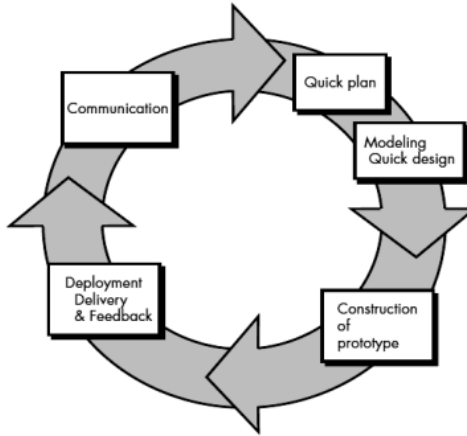
NoSQL adalah suatu cara penyimpanan data (datastore), dimana cara menyimpan dan mengambil kembali datanya dapat dilakukan dengan cepat NoSQL database bisa bekerja secara cepat dibanding basis data relasional. Perkembangan web yang cukup cepat menimbulkan kemajuan NoSQL karna selaku alternatif dalam memesatkan akses dibandingkan memakai basis informasi relasional. Ukuran kecepatan di dalam NoSQL ialah salah satu topik yang sangat menarik dalam dikaji untuk suatu riset (朝倉 2019). Sebagian riset yang akan terhubung dengan ukuran kecepatan, perbandingan performa sempat dicoba dalam riset sebelumnya, antara lain:

- a. Perbandingan performa nosql dengan relational database.
- b. Perbandingan kinerja relational database dengan data manipulation language nosql.
- c. Riset perbandingan konsep relational database dengan nosql. Bagi survey, peringkat awal dalam urutan database NOSQL berbasis stored document adalah MongoDB (Gunawan, 2018)

2.7 Metode Pengembangan perangkat lunak *Prototype*

(Pressman, 2012) menjelaskan bahwa *prototype* bukanlah hal yang lengkap, tetapi sesuatu yang harus dievaluasi ulang dan dimodifikasi. Ketika *prototype* yang memenuhi kebutuhan pengguna dibuat, semua perubahan akan terjadi, sekaligus

memungkinkan pengembang untuk lebih memahami kebutuhan pengguna. Tahapan model *prototype* dapat yang dapat dilihat pada gambar berikut



Gambar 2.1 Metode Prototype (pressman, 2012)

Tahapan dalam metode *prototype* antara lain:

1. Komunikasi dan pengumpulan data awal yaitu wawancara terhadap pihak yang terkait dalam penelitian dan analisis terhadap kebutuhan pengguna
2. Perencanaan Secara Cepat, yaitu pembuatan desain secara umum untuk selanjutnya dikembangkan kembali.
3. Pemodelan Perancangan Secara Cepat, yaitu perancangan dilakukan secara cepat dan berfokus pada tampilan perangkat lunak yang akan digunakan oleh pengguna.
4. Pembentukan *Prototype*, yaitu pembuatan perangkat *prototype* yang disesuaikan dengan kebutuhan pengguna.
5. Penyerahan Aplikasi pada Pengguna, yaitu tahapan akhir dari pembuatan aplikasi yang selanjutnya diserahkan kepada pengguna.

2.8 Pengujian Black-Box Testing

Black-Box Testing yaitu pengujian yang dilakukan menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak (Astuti and Puji, 2018).

Pengujian Black-Box Testing diterapkan dalam kategori :

1. Uji *interface*
2. Uji fungsi menu dan tombol
3. Uji struktur dan *database*



2.9 Sistem Pemodelan



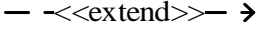
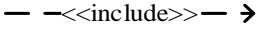
2.9.1 Unified Modeling Language (UML)

UML berkembang pesat pada akhir 1980-an dan awal 1990-an UML lahir dari penggabungan banyak bahasa permodelan grafis yang berorientasikan objek. UML dibuat oleh James Rumbaugh, Ivar Jacobson dan Grady Booch. di bawah bendera Rational Software Corp. UML memberika notasi-notasi yang dapat membantu memodelkan sistem dari berbagai perspektif. UML juga tidak hanya dapat digunakan dalam pemodelan perangkat lunak, hampir dalam semua bidang yang membutuhkan pemodelan (Fowler, 2019).

2.9.2 Use Case Diagram

Use Case diagram merupakan pemodelan untuk kegiatan pada sistem yang akan dibuat. Adapun penjelasan simbol-simbol *use case diagram* dapat dilihat pada Tabel 2.1 berikut

No	Simbol	Keterangan
1	Use Case 	<i>Use Case</i> : Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja diawal <i>frase</i> nama <i>use case</i> .
2	Aktor  Nama Aktor	Aktor: Seseorang yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri.


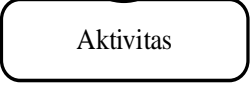
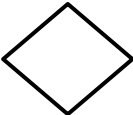

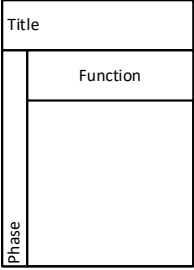

		Terkadang simbol aktor tidak selalu diartikan sebagai seseorang.
3	Asosiasi 	<i>Association</i> : Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> yang memiliki interaksi dengan aktor
4	Generalisasi 	<i>Generalization</i> : Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
5	Ekstensi/Extend 	<i>Extend</i> : Merupakan <i>use case</i> tambahan kesebuah <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan.
6	Inculde 	<i>Include</i> : Relasi <i>use case</i> ditambahkan kesebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> .

Tabel 2 1 Simbol Use Case diagram

2.9.3 Activity Diagram

Activity diagram ataupun Diagram aktivitas menggambarkan aliran fungsionalitas sistem. Pada sesi pemodelan bisnis, diagram kegiatan bisa digunakan untuk menggambarkan aliran peristiwa (flow of events) dan membuktikan aliran kerja bisnis (business work flow) dalam usecase. Contoh diagram aktivitas disajikan pada tabel 2.2 (John W. Satzinger, 2016).

No	Simbol	Keterangan
----	--------	------------

1		Status awal aktif sistem (<i>Initial</i>): Merupakan sebuah diagram aktivasi memiliki sebuah status awal.
2		Aktivitas yang dilakukan sistem, aktivitas biasanya diawal dengan sebuah kata kerja.
3		Percabangan (<i>Decision</i>): Merupakan asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4		Penggabungan (<i>Join</i>): Merupakan asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
5		Swimlane: Memisahkan organisasi bisnis yang bertanggung jawab terhadap sebuah aktivitas yang terjadi.
6		Status akhir yang dilakukan sistem (<i>Final</i>): Merupakan sebuah diagram aktivitas yang memiliki sebuah status akhir.

Tabel 2.2 Tabel Activity Diagram

2.9.4 Class Diagram

Class Diagram merupakan penggambaran struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat dalam membangun sistem.

2.10 Penelitian Relevan

Penelitian ini mencari dan mengumpulkan berbagai penelitian terdahulu untuk di jadikan referensi sebelum melakukan penelitian ada beberapa studi Relevan yang menggunakan algoritma Squeezer namun fokus pada aspek dan subjek yang berbeda

adapun perbedaan penelitian terkait ataupun penelitian terdahulu dengan penelitian ini adalah

1. Implementasi Algoritma Squeezer dan Term Frequency Ranking Dalam Pembangunan Sistem Rekomendasi Tempat Makan.

Penelitian ini dilakukan oleh Wirawan, Hansun dan Nugroho, I : 2004. Tujuan dibuatnya implementasi algoritma squeezer adalah bahwa rekomendasi yang dihasilkan oleh algoritma Squeezer ini sangat bergantung pada pilihan user. Pada penelitian ini, peneliti menggunakan algoritma squeezer dalam pembangunan sistem rekomendasi tempat makan.

2. Penerapan Algoritma Squeezer untuk Memberikan Rekomendasi Pilihan Lagu Berdasarkan Daftar Lagu yang di Mainkan Pada Pemutar MP3.

Penelitian ini dilakukan oleh Wibowo, Rochiman dan Munif : 2013. Android dengan tujuan untuk memberikan rekomendasi pilihan lagu kepada pengguna berdasarkan data riwayat lagu yang dimainkan pengguna. Penelitian ini fokus pada memberikan rekomendasi pilihan lagu kepada pengguna dengan menggunakan algoritma squeezer sebagai implementasi.

3. Analisis Implementasi Algoritma Clustering Squeezer Pada Sistem Rekomendasi Item-Based Collaborative Filtering.

Penelitian ini dilakukan oleh Supangat: 2015. Dengan tujuan meningkatkan jumlah prediksi dengan nilai rata-rata kenaikan 3,563 % dengan kecenderungan semakin besar nilai threshold pada proses pembentukan cluster, jumlah rekomendasi semakin meningkat dan penurunan akurasi dengan nilai rata-rata 0.76% dengan kecenderungan semakin besar nilai threshold pada proses pembentukan cluster, nilai MAE yang dihasilkan semakin meningkat. Penelitian ini fokus pada untuk meningkatkan jumlah prediksi pada sistem rekomendasi Item-Based Collaborative Filtering.

Pada penelitian ini peneliti terfokus pada sistem pendeteksi Cyberbullying Terhadap komentar postingan media sosial dengan metode Squeezer serta mengurangi komentar yang sifatnya menyerang dan data yang di gunakan hanya berfokus pada penggunaan data komentar dari sosial media instagram.

Judul,penu- lis tahun	Jumlah atribut	algorit- ma	preproces- sing	valid- asi	Ope- n sour- ce	akurasi
Implementasi Algoritma Squeezer dan Term Frequency Ranking Dalam Pembangu- nan Sistem Rekomend- asi Tempat Makan	Ada 7 atribut: percobaan pilihan,kriteria populasi dari user,poin popu- lasi daru user, kriteria popu- lasi dari sistem.poin po- pulasi dari sistem	Squee- zer dan term freque- ncy rankin- g	normalias- i			Tingkat recal 1 dan procisi- on 0.46
Penerapan Algoritma Squeezer untuk Memberik- an Rekeomen- dasi Pilihan Lagu Berdasark- an Daftar	Ada 5 atribut: juudl,artis,album,genr- e,tahun	squee- zer				Tingkat akurasi 84% dari 300 data master dan 50 data uji

Lagu yang di Mainkan Pada Pemutar MP3.						
Analisis Implementasi Algoritma Clustering Squeezer Pada Sistem Rekomendasi Item-Based Collaborative Filtering.		squeezer				Kenaikan rata-rata prediksi 3,563% dan penurunan akurasi rata-rata 0,76%

Tabel 2.3 Penelitian Relevan

