

BAB II

TINJAUAN PUSTAKA

2.1. Studi Sebelumnya

Penelitian yang dilakukan oleh Dachi,dkk yang berjudul klasifikasi sentimen opini pada aplikasi Guitar Tuner menggunakan dua model metode yaitu Naive Bayes dan Support Vector Machine (SVM) adalah untuk memberikan rekomendasi pada pengembang dalam meningkatkan kualitas aplikasi – aplikasi *Guitar Tuner*. Hasil dari penelitian adalah model SVM memiliki skor akurasi yang lebih tinggi dibandingkan model Naive Bayes [4].

Penelitian yang dilakukan oleh Ariq membahas tentang analisis sentimen positif aplikasi GuitarTuna dan Fender Guitar Tuner. Penelitian ini bertujuan untuk menganalisis sentimen positif dari aplikasi *Guitar Tuner* (GuitarTuna dan Fender Guitar Tuner) menggunakan model Naive bayes dan Support Vector Machine (SVM). Hasil yang didapatkan adalah metode SVM menghasilkan akurasi yang lebih tinggi untuk kedua aplikasi dibandingkan dengan metode Naive Bayes [5].

Penelitian yang dilakukan oleh Wulandari,dkk membahas tentang analisis sentimen aplikasi Spotify pada ulasan pengguna di Google Play Store. Penelitian ini bertujuan untuk menganalisis ulasan positif dan negatif sehingga bisa dijadikan acuan dalam mengembangkan aplikasi Spotify menjadi lebih baik menggunakan model Support Vector Machine (SVM). Hasil yang didapatkan adalah sentimen positif berjumlah lebih banyak dibandingkan sentimen negatif [6].

Penelitian yang dilakukan oleh Lesmana,dkk membahas tentang analisis sentimen pada ulasan aplikasi Joox Music. Penelitian ini dilakukan untuk mengklasifikasi sentimen ulasan pengguna Joox Music dengan akurat menggunakan model Naive Bayes dan Support Vector Machine (SVM). Hasil yang didapatkan adalah akurasi dari metode SVM lebih tinggi dibandingkan metode Naive Bayes [7].

Penelitian yang dilakukan Ainunnisa,dkk membahas tentang analisis sentimen pada aplikasi TikTok. Penelitian ini bertujuan untuk menganalisis opini pengguna TikTok dengan mengelompokkan ulasan positif, negatif, dan netral menggunakan model Support Vector Machine (SVM), Logistic Regression, dan Naive Bayes. Hasil yang didapat adalah sentimen positif menduduki peringkat pertama sentimen terbanyak, lalu disusul oleh sentimen negatif dan terakhir sentimen netral. Metode yang menghasilkan akurasi yang paling tinggi adalah metode Logistic Regression, lalu disusul oleh metode SVM dan terakhir metode Naive Bayes [8].



Tabel 2.1. Penelitian Sebelumnya

No.	Peneliti	Tahun	Tujuan	Metode	Hasil
1.	Dachi, dkk[4]	2024	Untuk memberikan rekomendasi pada pengembang dalam meningkatkan kualitas aplikasi – aplikasi <i>Guitar Tuner</i> .	<i>Naive Bayes and Support Vector Machine (SVM)</i> .	Dalam menggunakan kedua metode untuk proses klasifikasi sentimen, didapatkan bahwa metode SVM (skor akurasi 0,97) lebih unggul dibanding metode <i>Naive Bayes</i> (skor akurasi 0,95).
2.	Ariq [5]	2024	Untuk menganalisis sentimen positif dari aplikasi <i>Guitar Tuner</i> (<i>GuitarTuna</i> dan <i>Fender Guitar Tuner</i>).	<i>Naive Bayes</i> dan <i>Support Vector Machine (SVM)</i> .	Hasil penelitian menunjukkan bahwa metode <i>Naive Bayes</i> menghasilkan akurasi sebesar 92% untuk kedua aplikasi <i>tuner</i> , sedangkan metode SVM menghasilkan akurasi lebih tinggi yakni 97% untuk <i>GuitarTuna</i> dan 95% untuk <i>Fender Guitar Tuner</i> .
3.	Wulandari, dkk [6]	2024	Untuk menganalisis ulasan positif dan negatif sehingga bisa dijadikan acuan dalam	<i>Support Vector Machine (SVM)</i>	Dari ulasan sejumlah 5000 ulasan, didapatkan hasil 3193 sentimen positif dan 1347 sentimen negatif. Metode SVM menghasilkan akurasi sebesar

No.	Peneliti	Tahun	Tujuan	Metode	Hasil
			mengembangkan aplikasi Spotify menjadi lebih baik.		85%, <i>precision</i> 86%, <i>recall</i> 92%, dan f1-score 89%.
4.	Lesmana, dkk [7]	2023	Untuk mengklasifikasi sentimen ulasan pengguna Joox Music dengan akurat.	<i>Naive Bayes and Support Vector Machine</i> (SVM)	Hasil klasifikasi metode SVM (akurasi 55%) menghasilkan akurasi yang lebih tinggi dibandingkan metode <i>Naive Bayes</i> (akurasi 52.5%).
5.	Ainunnisa, dkk [8]	2023	Untuk menganalisis opini pengguna TikTok dengan mengelompokkan ulasan positif, negatif, dan netral.	<i>Support Vector Machine</i> (SVM), <i>Logistic Regression</i> , <i>Naive Bayes</i> .	Berdasarkan hasil penelitian dengan tiga metode tersebut pada aplikasi TikTok, dari data ulasan sebanyak 2100 didapatkan 934 positif, 313 netral, dan 853 negatif. Skor akurasi yang didapat dari metode: 81% SVM, 84% <i>Logistic Regression</i> , dan 79% <i>Naive Bayes</i> .

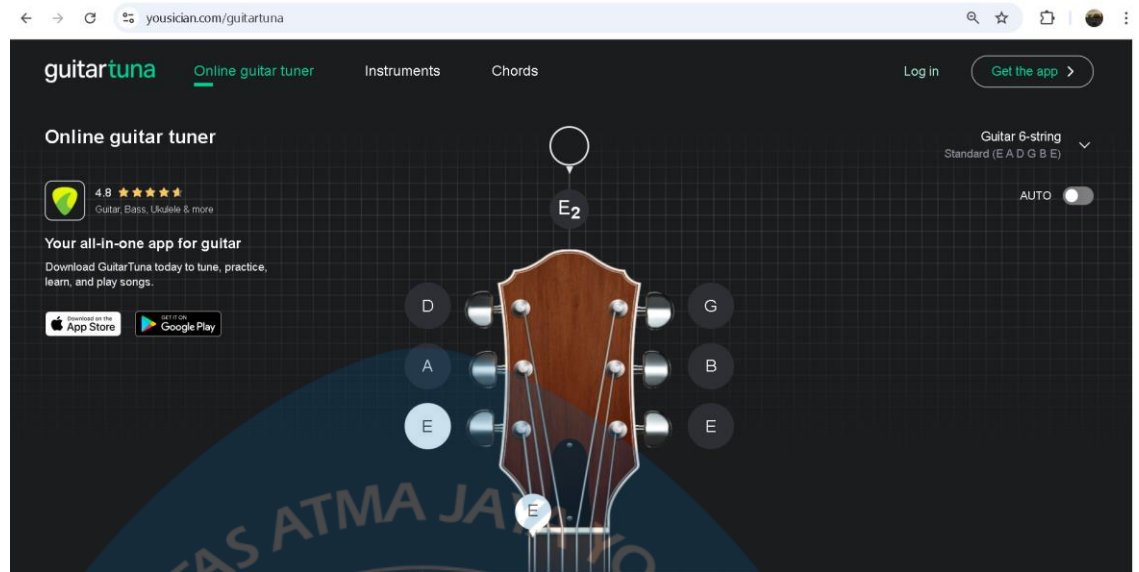
2.2. Dasar Teori

2.2.1 Pengertian Musik

Musik adalah cabang seni yang berfokus pada rangkaian suara atau bunyi yang tersusun dalam komposisi dengan kesatuan ritme dan tempo yang harmonis. Komposisi ini dapat dinikmati oleh banyak orang dan umumnya dihasilkan melalui suara manusia maupun dengan bantuan alat musik seperti gitar, drum, biola, piano, dan lainnya. Komposisi dasar dari musik tersusun dari adanya melodi, ritme, harmoni, dinamika, dan tempo. Secara fisiologis, musik berhubungan dengan indra pendengaran, namun secara psikologis musik berhubungan dengan fungsi psikis manusia seperti persepsi, *mood*, dan berbagai fungsi psikologis lainnya[9]. Musik sendiri dapat merangsang pola pikir dan menjadi jembatan bagi pemikiran-pemikiran yang kompleks seperti matematika contohnya, sebab itu ada hubungan yang logis antara musik dan matematika karena keduanya menyangkut skala yang naik turun, yaitu adanya unsur ketukan dalam musik dan angka dalam matematika[10].

2.2.2 Aplikasi GuitarTuna

Aplikasi GuitarTuna adalah aplikasi berbasis *platform mobile* dan PC yang dibuat oleh Yousician. Aplikasi ini memiliki teknologi pengenalan suara berupa *microphone* yang memungkinkan pengguna menyetem senar instrumen seperti (gitar, bass, ukulele, biola, dll) secara otomatis dengan cara mendekatkan perangkat (aplikasi GuitarTuna) ke sumber suara yakni alat musik. Aplikasi GuitarTuna dapat mendeteksi nada dari instrumen musik yang dimainkan dengan bantuan mikrofon bawaan ponsel, lalu memberikan panduan apakah nadanya sudah tepat (*in-tune*) atau masih terlalu rendah (*flat*) dan tinggi (*sharp*). Untuk gambaran fitur *tuner* alat musik, dapat dilihat pada gambar 2. 1:



Gambar 2. 1 Fitur Tuner Aplikasi GuitarTuna

Sumber: <https://yousician.com/guitartuna>

Selain fitur utama yakni *tuner*, aplikasi GuitarTuna juga memiliki berbagai *toolkit* pendukung kegiatan bermusik seperti *metronome*, *chord libraries*, dan *mini games* (menebak suara dan bentuk dari suatu *chord*). Adapun juga fitur premium dari aplikasi GuitarTuna yakni GuitarTuna Play, yaitu media pembelajaran *online* dari GuitarTuna yang memungkinkan pengguna untuk belajar memainkan lagu-lagu populer dengan berbagai tingkat kesulitan yang beragam, seperti *simple* untuk pemula yang baru belajar hingga tingkat *intermediate / advanced* untuk pengguna yang sudah berpengalaman dan ingin menambah variasi kompleks dalam bermain alat musik. Aplikasi GuitarTuna juga sudah dilengkapi dengan AI atau *Artificial Intelligence* dalam penggunaan mikrofon untuk mendeteksi suara alat musik pada saat *tuning* nada dan bisa melakukan *auto-scroll* dari *chord sheet* yang menyesuaikan dengan permainan pengguna. Fitur *auto-scroll* ini berguna untuk mengontrol tempo dan menampilkan *chord* serta lirik secara otomatis saat pengguna mulai bermain.

Dari banyaknya fitur yang bermanfaat di atas untuk keperluan belajar bermusik untuk pengguna, aplikasi GuitarTuna banyak diunduh oleh jutaan pengguna dan keragaman dari ulasan yang diberikan pengguna dari berbagai kalangan baik yang masih pemula maupun yang sudah

berpengalaman. Faktor tersebut memungkinkan aplikasi GuitarTuna memiliki jumlah data yang banyak dan adanya variasi dari isi ulasan pengguna terkait aplikasi, sehingga dapat dijadikan objek penelitian analisis sentimen yang memadai dari segi jumlah data ulasan berkat kepopuleran aplikasi.

2.2.3 Analisis Sentimen

Analisis Sentimen merupakan proses menganalisis teks berbentuk digital untuk menentukan apakah nada emosional (sentimen) dari suatu pesan bersifat positif, negatif, atau netral. Dalam konteks aplikasi, analisis sentimen penting untuk dilakukan supaya membantu pengembang aplikasi dalam meningkatkan kualitas performa dan layanan dari aplikasi, berdasarkan pesan digital (ulasan aplikasi) dari pengguna aplikasi. Analisis sentimen bisa dilakukan di beberapa media sosial, seperti *Twitter*, *Youtube*, *Instagram*, media pemerintahan, *platform e-commerce*, dll [11]. Dalam proses melakukan analisis sentimen, diperlukan metode yang menunjang klasifikasi, salah satu metodenya adalah *Support Vector Machine* (SVM), berdasarkan hasil penelitian analisis sentimen sebelumnya yang dilakukan oleh Putranti dan Winarko, penelitian tersebut dilakukan pada aplikasi *twitter* dengan mengimplementasikan metode *Maximum Entropy* dan SVM hingga mendapat nilai akurasi sebesar 86.61% [12].

2.2.4 Metode Support Vector Machine

Metode yang menggunakan *machine learning* ini adalah algoritma yang efektif untuk melakukan analisis sentimen. Metode ini biasanya digunakan dalam hal klasifikasi teks dengan mencari *hyperlane* atau garis batas pemisah yang dapat memisahkan dua kelas data misalnya sentimen positif dan negatif. Metode *Support Vector Machine* dikenal sebagai teknik pembelajaran mesin paling mutakhir dan berhasil setelah pembelajaran mesin sebelumnya yakni *Neural Network* dalam

pengenalan pola sebagai usaha mencari fungsi pemisah (*hyperlane*) dengan memaksimalkan *margin* atau jarak antar kelas teks[13].

Pemilihan metode SVM dalam penelitian ini dapat menjadi pilihan dalam menangani klasifikasi teks seperti analisis sentimen. Metode SVM juga efektif untuk *dataset* berukuran sedang karena tidak membutuhkan sumber daya komputasi yang kompleks, namun tetap dapat memberikan performa yang baik. Selain itu, pada beberapa penelitian sebelumnya yang tercantum pada tabel 2.1, metode SVM menunjukkan hasil akurasi yang lebih tinggi dibandingkan metode lain seperti Naive Bayes contohnya, sehingga metode SVM dinilai sesuai untuk digunakan pada klasifikasi sentimen penelitian ini. Pada tabel 2.2 berikut membahas mengenai perbandingan metode SVM dengan 2 metode lain yang digunakan pada tinjauan pustaka, yaitu Naive Bayes dan Logistic Regression:

Tabel 2.2 Perbandingan Model Analisis Sentimen

SVM	Naive Bayes	Logistic Regression
Unggul dalam mengklasifikasi berdasarkan pembobotan kata yang berdimensi tinggi dan margin pemisah teks meski waktu pengerjaan lebih lama jika dilakukan pada data yang besar, tetapi dapat menghasilkan akurasi yang baik di angka 80% ke atas.	Model lebih sederhana dan cepat digunakan, namun tidak bisa mengklasifikasikan teks ulasan secara keseluruhan (per kata) yang menyebabkan ambigu atau kesalahan nada sentimen pada suatu ulasan. Contoh (tidak bagus bisa menjadi positif karena ada kata bagus).	Memiliki keunggulan dalam mengklasifikasikan kata yang paling berpengaruh, data non-linear, tetapi menghasilkan hasil akurasi yang kurang maksimal pada teks yang kompleks.

2.2.5 Pre-processing

Tahap *pre-processing* adalah tahapan awal dalam hal *machine learning*, yakni mengolah data yang belum terstruktur sebelum dilakukan proses klasifikasi lebih lanjut. Proses ini mencakup beberapa tahapan awal seperti pembersihan data, integrasi data, transformasi data, dan reduksi data. Pertama, pembersihan data dilakukan dengan menghilangkan data yang tidak lengkap, tidak relevan dan duplikasi data. Kedua, integrasi data dilakukan dengan menyatukan data dari berbagai tempat / sumber menjadi satu *dataset*. Ketiga, transformasi data dilakukan dengan memperbaiki format data supaya sesuai dengan kebutuhan dari analisis yang dilakukan dan model yang digunakan. Terakhir reduksi data dilakukan dengan mengurangi dimensi data supaya lebih mudah pada saat pemrosesan. Tahap ini bertujuan untuk mengubah data mentah menjadi data yang berkualitas dan dihindarkan dari data yang mengganggu atau tidak konsisten(*noise*), sehingga data layak untuk dilakukan pemrosesan lebih lanjut[14]. Pada proses ini, dilakukan *scrapping* data ulasan secara otomatis dengan mengakses halaman web yang diinginkan, dan menyimpan data ulasan tersebut dalam bentuk format CSV[15]. Berikut adalah tahapan yang ada dalam melakukan proses ini:

1. *Cleaning*: Proses pembersihan data dilakukan untuk menemukan dan memperbaiki data yang tidak valid atau tidak berguna dari suatu kumpulan data. Proses *cleaning* membantu mempersiapkan data untuk diuji pada tahap selanjutnya yakni pengolahan dan pengintegrasian data[16].
2. *Case Folding*: Proses mengubah seluruh huruf dalam suatu kalimat / teks menjadi huruf kecil. Proses ini diterapkan untuk mengubah semua karakter yang ada menjadi bentuk yang sama[17]. Oleh karena itu, proses ini dilakukan untuk memastikan kualitas data dan mengurangi dimensi data yang tidak perlu.

3. *Normalization*: Proses mengubah kata yang tidak baku menjadi baku atau bentuk formal[18]. Proses ini bertujuan untuk menyamakan bentuk kata yang memiliki makna sama, namun berbeda penulisan, sehingga dapat meningkatkan akurasi dari model dan mengurangi dimensi fitur.
4. *Stopword Removal*: Proses memisahkan kata-kata yang tidak terlalu penting[19]. Tujuan proses ini untuk mengurangi *noise* dan mempercepat efisiensi kinerja model dalam menentukan sentimen (positif, negatif, netral).
5. *Stemming*: Proses menghapus awalan atau akhiran kata yang terdapat kata sambung, kata depan, kata ganti, menjadi kata dasar sesuai dengan KBBI[20]. Tujuan dari proses ini untuk menyederhanakan suatu kata sehingga dapat mengurangi jumlah variasi kata yang memiliki makna yang sama dan membantu model seperti SVM dalam memahami konteks kata dengan lebih konsisten.
6. *Tokenizing*: Proses untuk memecah kalimat berdasarkan dari kata penyusunnya dan kata tersebut dipecah berdasarkan spasi[21]. Tujuan dilakukannya proses ini adalah memisahkan kata supaya saat proses klasifikasi bisa dilakukan dengan lebih efisien.
7. *Labeling*: Proses ini dilakukan untuk memberikan kode atau simbol dari masing-masing ulasan untuk mengetahui apakah berbobot positif, negatif atau netral. Bagian dari proses ini berisikan simbol dari skor sentimen untuk setiap ulasan, sebagai contoh yaitu 1 untuk positif, 0 untuk negatif, dan 2 untuk netral[22].

2.2.6 TF-IDF

TF-IDF atau yang dikenal sebagai *Term Frequency-Inverse Document Frequency* adalah hasil perkalian dari dua hal yakni TF dan IDF yang digunakan untuk mengevaluasi frekuensi suatu kata. TF di sini digunakan untuk mengukur frekuensi kata dalam suatu dokumen, dan IDF digunakan untuk memperkecil bobot kata yang sering muncul dalam dokumen secara keseluruhan. Proses ini termasuk dalam *term weighting*, yaitu suatu proses pembobotan dari setiap kata agar bisa mengoptimalkan kemampuan analisis sentimen pada proses *text mining*[23]. Dengan adanya proses pembobotan TF-IDF, dapat membantu dalam proses identifikasi kata-kata kunci yang paling relevan dalam suatu dokumen[24].

2.2.7 Bahasa Pemrograman Python

Bahasa pemrograman Python adalah bahasa pemrograman tingkat tinggi yang dikembangkan pada tahun 1991 oleh Guido Van Rossum. Bahasa Python memiliki sintak yang mudah dibaca dan dipahami, biasanya banyak digunakan untuk keperluan pengembangan web hingga analisis data. Bahasa pemrograman Python cocok juga digunakan sebagai *scripting language*, implementasi bahasa pemrograman web, bahkan bisa di *extended* ke bahasa pemrograman lain seperti C dan C++[25]. Dalam menerapkan bahasa pemrograman Python, terdapat beberapa aturan yang harus dipenuhi, untuk mengantisipasi adanya *error* pada program yang dibuat. Oleh karena itu, aturan sintak dari bahasa pemrograman Python yang pertama adalah penulisan suatu perintah atau *statement* yang mendukung kode bahasa pemrograman tersebut[26].

2.2.8 Scikit-learn

Scikit-learn atau sklearn adalah sebuah *library* dari bahasa pemrograman Python yang berguna untuk hal pembelajaran *machine learning* (ML), dengan menyajikan algoritma untuk tugas ML seperti proses klasifikasi, melakukan *preprocessing data*, dan mengukur evaluasi kinerja model. Penggunaan *library* Scikit-learn sangat efisien untuk keperluan *data mining* dan analisis data [27]. Scikit-learn menyediakan berbagai fungsi dan metrik untuk proses klasifikasi model dan evaluasi performa model seperti fungsi `accuracy_score` untuk menghitung akurasi dan memiliki metode `confusion_matrix` untuk menghitung dan menampilkan hasil akurasi dengan lebih detail[28].

