

BAB 1

PENDAHULUAN

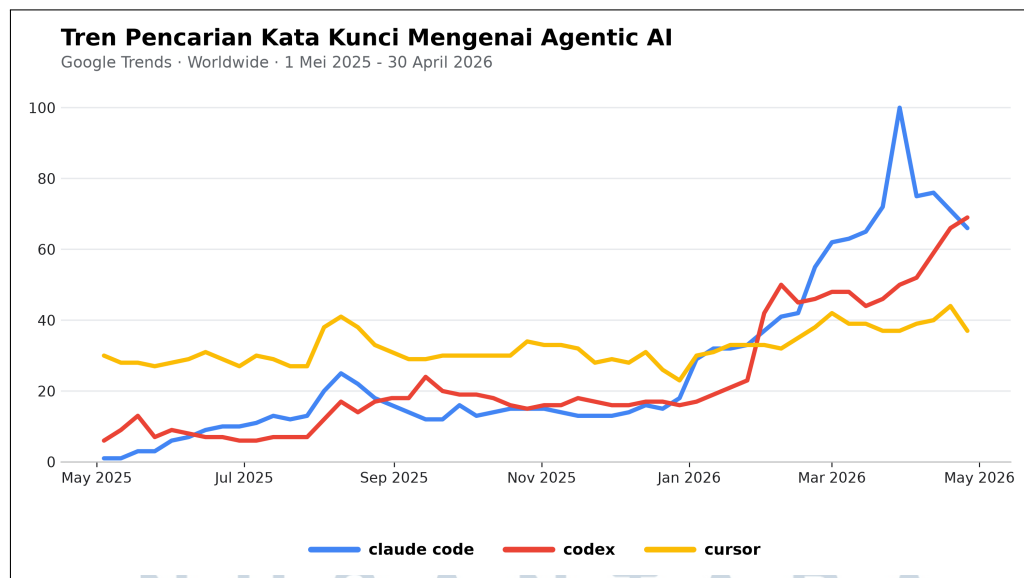
1.1 Latar Belakang Masalah

Perkembangan teknologi kecerdasan buatan (AI) dalam rekayasa perangkat lunak telah mengalami perubahan signifikan dari sekadar pelengkap kode (*autocomplete*) menjadi *Agentic AI* yang beroperasi secara otonom pada skala repositori secara menyeluruh. Sistem *agentic modern* seperti Claude Code, Cursor, dan Codex kini tidak hanya melengkapi baris kode, tetapi mampu memecah tugas menjadi beberapa langkah, mengeksekusi perintah terminal, menjalankan pengujian, dan melakukan perbaikan *error* secara iteratif [1]. Namun, lompatan teknologi ini memicu potensi disrupsi yang nyata bagi pasar tenaga kerja. Profesi *programmer* menempati peringkat teratas sebagai pekerjaan yang paling terekspos otomatisasi dengan tingkat paparan mencapai 74,5% terhadap tugas-tugas intinya [2]. Ancaman tersebut diperkuat oleh data pasar tenaga kerja yang menunjukkan penurunan angka perekrutan dan penyusutan lapangan kerja hingga 16% bagi kelompok pekerja muda pada rentang usia 22 hingga 25 tahun di profesi yang terekspos AI [2].

Ancaman substitusi ini memicu kepanikan di lapangan, terutama pada kelompok pekerja di awal karir yang terekam sebagai demografi paling rentan berdasarkan survei ekonomi AI terhadap 81.000 pengguna [3]. Kepanikan tersebut diperparah oleh munculnya beban kognitif serta ancaman penurunan keahlian akibat adopsi AI secara berkelanjutan saat mempelajari tugas baru [4]. Evaluasi eksperimental membuktikan bahwa ketergantungan pada AI secara drastis menurunkan pemahaman konseptual, kemampuan membaca kode, dan keahlian *debugging* mandiri pada *programmer* pemula [4]. Di sisi lain, *programmer* profesional mengeluhkan pergeseran peran fungsional yang kini dituntut menjadi pengawas ketat atas logika buatan AI. Alih-alih menikmati praktik pemrograman yang menyerahkan sebagian besar proses kepada AI atau yang dikenal dengan istilah "*vibe coding*", praktisi senior menyadari penggunaan AI tetap memerlukan arahan, pengawasan, dan validasi manusia agar hasil yang dihasilkan sesuai dengan kebutuhan pengembangan perangkat lunak [5]. Hal ini sejalan dengan temuan bahwa proses verifikasi serta pemahaman terhadap kode yang dihasilkan oleh kecerdasan buatan justru menuntut beban kognitif dan upaya mental yang lebih

tinggi dibandingkan meninjau kode buatan manusia, sehingga intervensi manual tetap diperlukan demi menjaga standar kualitas perangkat lunak [6].

Di tengah kepanikan pemula dan beban pengawasan teknis tersebut, kehadiran alat *agentic* tidak dapat dipungkiri turut memberikan dampak positif berupa peningkatan produktivitas yang signifikan. Penggunaan alat berbasis *Agentic AI* terbukti mampu menghemat waktu penyelesaian tugas perangkat lunak secara signifikan hingga 55,8% dalam eksperimen terkontrol [1]. Integrasi pengetahuan yang tepat dengan dukungan AI secara langsung meningkatkan efisiensi dan performa pengembangan sistem secara keseluruhan [7]. Kontras antara optimisme terhadap peningkatan produktivitas dan kekhawatiran terhadap penurunan keahlian serta kelelahan kognitif inilah yang menciptakan perbedaan opini yang cukup kuat di ruang publik. Mengingat komunitas teknologi sangat aktif mengekspresikan perdebatan teknis dan sentimen emosional di media sosial, *platform X* menjadi medium representatif untuk memetakan dinamika polarisasi ini secara luas [8]. Urgensi pemetaan opini publik tersebut sejalan dengan meningkatnya perhatian terhadap teknologi *agentic* dalam pengembangan perangkat lunak. Hal ini tercermin jelas pada Gambar 1.1, yaitu tren pencarian global di Google Trends untuk alat seperti Claude Code, Cursor, dan Codex menunjukkan peningkatan signifikan pada periode Januari hingga April 2026.



Gambar 1.1. Grafik pencarian kata kunci mengenai agentic ai dari Mei 2025 - April 2026

Sumber: [9]

Seiring dengan tren adopsi tersebut, banyaknya opini publik yang mengalir

di *platform* X membuat klasifikasi sentimen secara manual tidak lagi efisien, sehingga diperlukan pendekatan *machine learning*. Di antara berbagai algoritma yang tersedia, *Support Vector Machine* (SVM) diakui dalam penelitian sebagai metode yang andal untuk klasifikasi teks karena memiliki landasan teoretis yang kuat dan secara empiris berkinerja lebih akurat dibandingkan sebagian besar algoritma lainnya dalam memisahkan batas kelas secara maksimal [10]. Keunggulan ini turut dibuktikan pada penelitian klasifikasi sentimen data Twitter, yakni SVM terbukti mengungguli algoritma seperti *K-Nearest Neighbor* (KNN), *Decision Tree* (DT), dan *Naive Bayes* (NB) dengan tingkat akurasi mencapai 86,42% [11]. Performa superior SVM juga ditemukan pada penelitian analisis sentimen finansial di Twitter dengan capaian akurasi tertinggi sebesar 89% dibandingkan algoritma pembanding lainnya [12]. Lebih lanjut, algoritma ini terbukti sangat tangguh untuk memproses data cuitan berbahasa Indonesia, terutama ketika dihadapkan pada masalah kelas data yang tidak seimbang (*imbalanced dataset*). Integrasi SVM dengan metode penyeimbangan SMOTE terbukti mampu mengoptimalkan akurasi klasifikasi secara signifikan, sebagaimana ditunjukkan oleh penelitian yang mendongkrak akurasi hingga 83,89% [13], serta penelitian sentimen isu lokal berkernel linear yang meraih akurasi 85,56% [14]. Penelitian tersebut juga menunjukkan bahwa SVM tetap menjadi alternatif yang relevan untuk analisis sentimen karena, dibandingkan pendekatan *Deep Learning* seperti LSTM maupun model berbasis *Transformer* seperti BERT, metode tersebut umumnya memerlukan ukuran dataset yang lebih besar, kebutuhan komputasi yang lebih tinggi, serta waktu pelatihan yang lebih lama [14].

Berdasarkan uraian tersebut, diperlukan analisis sentimen untuk memetakan kecenderungan opini publik terhadap penggunaan *Agentic AI* dalam pengembangan perangkat lunak, khususnya di tengah perbedaan pandangan mengenai peningkatan produktivitas, risiko penurunan keterampilan, dan perubahan peran *developer*. Oleh karena itu, penelitian ini menerapkan algoritma SVM dengan ekstraksi fitur *Term Frequency-Inverse Document Frequency* (TF-IDF) untuk mengklasifikasikan sentimen pada data cuitan dari *platform* X. Untuk memperoleh konfigurasi model yang sesuai dengan karakteristik data, pembentukan model juga mempertimbangkan penggunaan SMOTE dalam menangani ketidakseimbangan kelas serta metode *hyperparameter tuning*. Hasil penelitian memberikan gambaran berbasis data mengenai sentimen publik terhadap *Agentic AI* sekaligus menjadi referensi bagi *junior developer* dan praktisi teknologi dalam memahami dinamika adopsinya serta mempertimbangkan keterampilan yang perlu dikembangkan di

masa mendatang.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan sebelumnya, maka rumusan masalah dalam penelitian ini adalah sebagai berikut:

1. Bagaimana analisis sentimen Agentic AI dalam pengembangan perangkat lunak di X menggunakan algoritma *Support Vector Machine*?
2. Bagaimana kinerja algoritma *Support Vector Machine* dalam menganalisis sentimen terhadap *Agentic AI* dalam pengembangan perangkat lunak di *platform X* berdasarkan metrik akurasi, *precision macro*, *recall macro*, dan *F1-score macro*?

1.3 Batasan Permasalahan

Berdasarkan uraian pada latar belakang, penelitian ini merumuskan dua permasalahan utama sebagai berikut:

1. Sumber data penelitian hanya berasal dari *platform X*.
2. Data cuitan yang digunakan dibatasi pada rentang waktu Januari 2026 hingga April 2026.
3. Data cuitan yang digunakan dibatasi pada cuitan berbahasa Inggris.
4. Klasifikasi sentimen dibatasi pada dua kategori utama, yaitu positif dan negatif.
5. Metode yang digunakan terbatas pada algoritma *Support Vector Machine* dengan teknik ekstraksi fitur *Term Frequency-Inverse Document Frequency*.

1.4 Tujuan Penelitian

Tujuan penelitian ini disusun untuk menjawab rumusan masalah yang telah ditetapkan, yaitu:

1. Menganalisis sentimen terhadap Agentic AI dalam pengembangan perangkat lunak di X dengan menggunakan *Support Vector Machine* ke dalam kategori positif dan negatif.

2. Mengukur kinerja algoritma *Support Vector Machine* dalam menganalisis sentimen terhadap *Agentic AI* dalam pengembangan perangkat lunak di platform X berdasarkan akurasi, *precision macro*, *recall macro*, dan *F1-score macro*.

1.5 Manfaat Penelitian

Manfaat penelitian adalah sebagai berikut:

1. Memberikan wawasan mengenai penerapan algoritma *Support Vector Machine* dalam menganalisis sentimen pada data tekstual dari *platform X* terkait *Agentic AI* dalam pengembangan perangkat lunak.
2. Memberikan referensi berbasis data bagi *junior developer* dan praktisi teknologi mengenai dinamika adopsi *Agentic AI* dalam pengembangan perangkat lunak, sehingga dapat menjadi salah satu pertimbangan dalam pengembangan keterampilan di masa mendatang.

1.6 Sistematika Penulisan

Berisikan uraian singkat mengenai struktur isi penulisan laporan penelitian, dimulai dari Pendahuluan hingga Simpulan dan Saran.

Sistematika penulisan laporan adalah sebagai berikut:

1. Bab 1 PENDAHULUAN

Bab ini memuat dasar penelitian yang terdiri atas latar belakang masalah, rumusan masalah, batasan permasalahan, tujuan penelitian, manfaat penelitian, dan sistematika penulisan.

2. Bab 2 LANDASAN TEORI

Bab ini menjelaskan teori, metode, dan penelitian terdahulu yang digunakan sebagai dasar dalam penelitian, meliputi analisis sentimen, *platform X*, *Agentic AI*, *Cohen's Kappa*, prapemrosesan data, TF-IDF, SMOTE, SVM, *hyperparameter tuning*, *cross validation*, dan *confusion matrix*.

3. Bab 3 METODOLOGI PENELITIAN

Bab ini menguraikan tahapan penelitian yang dilakukan, mulai dari studi literatur, pengumpulan data, perancangan sistem, pengolahan data, pembentukan model, hingga pengujian dan evaluasi.

4. Bab 4 HASIL DAN DISKUSI

Bab ini menyajikan hasil implementasi dan pengujian model, termasuk spesifikasi sistem, implementasi program, skenario uji coba, evaluasi performa model, analisis sentimen, serta diskusi terhadap hasil yang diperoleh.

5. Bab 5 SIMPULAN DAN SARAN

Bab ini berisi simpulan berdasarkan hasil penelitian serta saran yang dapat digunakan sebagai pertimbangan untuk penelitian selanjutnya.

