

Hal Besar Berikutnya dalam AI: SYSTEM ONE

1 Okt 2026 · 4 menit dibaca · Oleh [Viktor Zaika](#)

Jev dan Liquid d1 menghasilkan pilihan dan probabilitas, bukan teks generatif. Mengapa model keputusan penting bagi produk AI.

Sistem Satu

Jev

Model keputusan

Arsitektur AI

Liquid d1



Oke, yang ini benar-benar menarik.

Untuk beberapa tahun terakhir cukup banyak semua produk AI dibangun di sekitar LLM. Anda mengirim sesuatu, model menghasilkan sesuatu kembali, mungkin itu memberi Anda JSON, maka perangkat lunak Anda menggunakan apa pun yang keluar dari itu. Yang bekerja dengan baik untuk banyak hal, tapi juga berarti kita kadang-kadang memanggil beberapa model besar hanya untuk menjawab pertanyaan kecil konyol seperti apakah sesuatu yang mencurigakan atau di mana persis permintaan ini harus pergi.

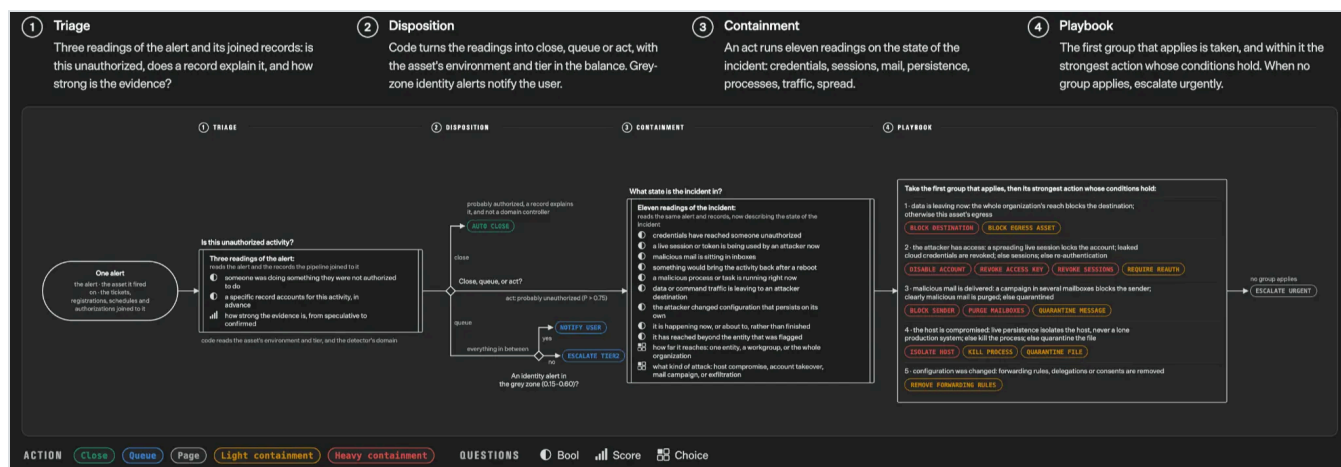
Sekarang ada Jev, yang diciptakan oleh TypeSafe, dan mereka menyebut sistem kelas baru ini sebagai model satu. Jadi bayangkan sesuatu yang lebih dekat dengan fungsi keputusan yang sangat cerdas daripada chatbot lain. [TypeSafeasures introduct](#) .

Reaksi pertama saya adalah agak oke, bukankah ini pada dasarnya sebuah klasifikasi? dan dalam beberapa hal ya. tapi kemudian saya mulai berpikir tentang berapa banyak LLM panggilan mahal pada produk nyata pada dasarnya adalah klasifikasi mengenakan setelan \$ 20.

Katakanlah Anda memiliki beberapa SaaS di mana setiap permintaan yang masuk saat ini pergi ke model perbatasan beberapa permintaan sepele beberapa harus pergi ke model yang lebih murah beberapa membutuhkan ulasan manusia beberapa mungkin curiga Jev bisa membuat keputusan pertama ini

sangat cepat dan kemudian model mahal sebenarnya hanya bangun ketika Anda benar-benar membutuhkannya

request → Jev → decision → whatever actually needs to happen



Contoh dari keputusan dalam aliran perangkat lunak yang lebih besar. [Lihat yang asli dan penjelasannya.](#)

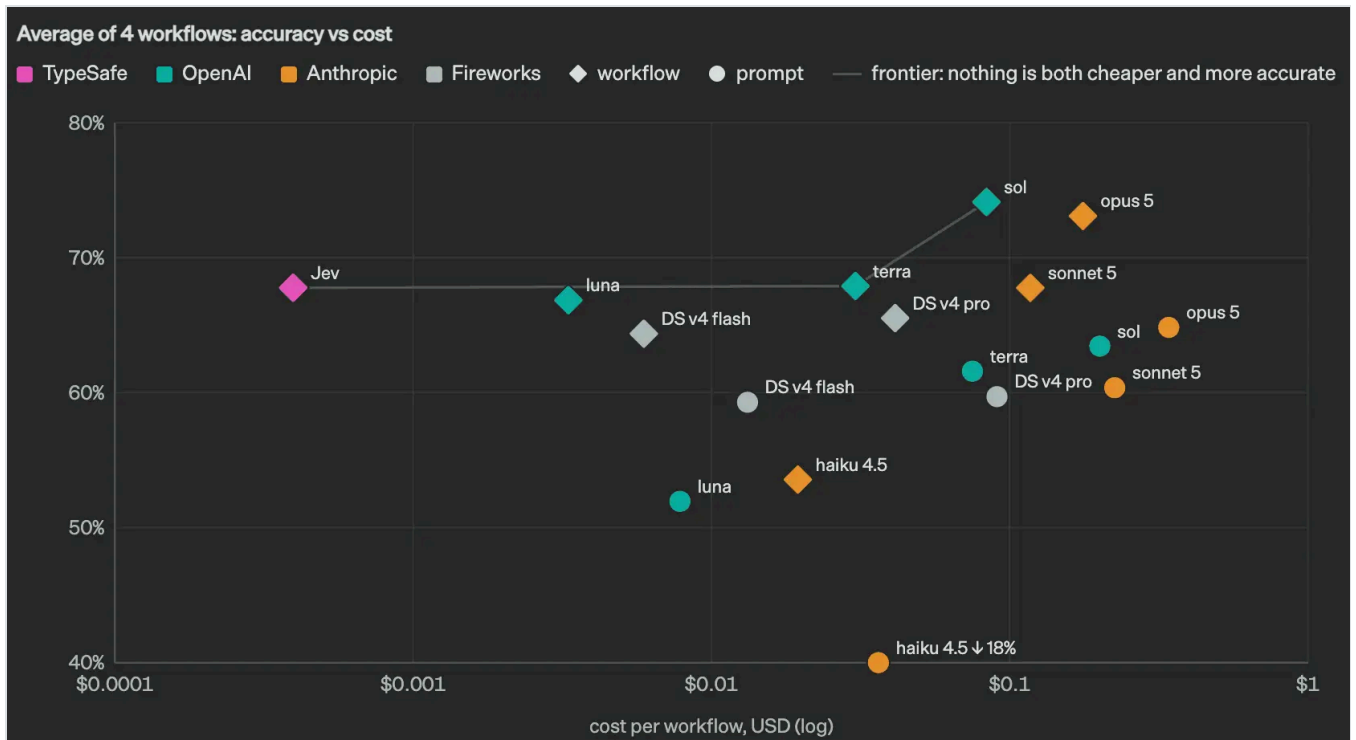
Ada satu contoh Reddit yang saya suka persis karena sangat sederhana. mereka memiliki konteks besar masuk ke LLM pada dasarnya untuk mencari tahu apakah ada sesuatu yang perlu dikhawatirkan. Setelah memindahkan cek pertama ini ke Jev mereka melaporkan akan dari 3 sampai 6 detik sampai milidetik, sementara LLM yang lebih besar hanya disebut jika Jev menemukan sesuatu yang layak diperiksa lebih jauh. [Baca diskusinya.](#)

Dan sejujurnya, untuk orang-orang produk ini mungkin lebih menarik dari beberapa model baru dengan nilai 2% lebih tinggi pada benchmark lain produk dapat menjadi lebih cepat. undang-undang inferensi Anda dapat menjadi lebih kecil. Anda juga dapat memiliki keputusan kecil di mana-mana tanpa merasa seperti Anda mengirimkan setiap hal kecil ke Godzilla.

Anda tidak perlu model raksasa yang cerdas untuk setiap keputusan kecil dalam produk ini, terkadang yang Anda butuhkan pada dasarnya adalah pernyataan yang sangat cerdas.

Sekarang ada juga Liquid d1, yang dibuat untuk jenis tugas pengambilan keputusan seperti ini. Jadi kita mulai melihat lebih dari satu implementasi gagasan tersebut, dan saya penasaran ke mana kategori ini akan berkembang. Dokumentasinya menunjukkan pola yang sama: masukkan state, lalu dapatkan jawaban Choice, Score, atau Noul untuk pertanyaan ya/tidak. [Baca dokumentasi Liquidés](#)

LLM tetap sangat berguna ketika saya perlu menghasilkan, menjelaskan, mengubah, atau merangkum sesuatu. Tetapi jika di suatu bagian arsitektur saya ada panggilan LLM yang hasil akhirnya cuma true, "sales", atau 0.83, saya akan mulai bertanya apakah panggilan itu masih perlu. Artikel komunitas Hugging Face tentang Jev menjelaskan perbedaan antara keluaran yang terbatas dan keputusan di dalam perangkat lunak ini. [Baca artikel komunitasnya.](#)



Evaluasi akurasi melawan biaya sendiri, melalui empat aliran kerja yang dipilih. [Baca metodologi dan caveats](#)

Sekarang jelas tidak membangun kembali seluruh perusahaan di sekitar ini besok Jev adalah baru, banyak yang mengesankan angka kinerja berasal dari TypeSafe sendiri, dan benang Reddit telah orang mengatakan ini adalah klasifikasi dimuliakan dan / atau sangat overhyped. cukup adil. Kita perlu pengujian yang lebih independen sebelum menyatakan beberapa era baru arsitektur AI. [Hasil TypeSafeasures](#) . [Diskusi Reddit](#) .

Tapi kalau kamu sedang membangun produk AI, saya pasti akan mencobanya.

Terutama jika Anda melihat penggunaan LLM Anda dan tiba-tiba menyadari setengah dari itu ada hanya untuk membuat keputusan kecil.