Episode Book

EPISODE #5941

Five Minutes Reading Your Dictation Prompt

PUBLISHED	13 October 2026 at 18:37 UTC
DURATION	27:14
LISTEN	myweirdprompts.com/episode/speech-to-text-cleanup-prompt-critique-52265/
HOSTS	Corn, Herman, Hilbert
SCRIPT MODEL	deepseek-flash
VOICES	chatterbox-regular (chatterbox-local)
PIPELINE	V5.4
SOURCES	9 cited by the research (listed at the end)
BOOK GENERATED	19 October 2026 at 17:41 UTC

SYNOPSIS

Daniel sent in a general speech-to-text cleanup prompt built to fix the failure patterns he actually hits, and asked for a full read, a critique, and improvements. So the prompt is the artifact: we read it first, argue about it second. It does two genuinely unusual things — executing spoken self-corrections and stripping out speech aimed at someone nearby — but it also assumes every error is a word-level substitution. That assumption collides with what ASR really does: out-of-vocabulary jargon like "Dupixent" coming out as "duplex aunt," and Whisper hallucinating whole phrases like "thanks for watching" in the middle of silence. We work through five categories of ASR failure, what the prompt can and can't reach, and what we'd change.

TRANSCRIPT



Corn

What if you took five minutes and actually read the cleanup prompt you paste into your dictation app every single day?



Herman

Most people would rather not know.



Corn

Daniel sent us his. He's been iterating on a general speech-to-text cleanup prompt, built to fix the failure patterns he actually hits, and he thinks it goes past the usual cleanup boilerplate. He wants it read out in full, then critiqued, then improved. So the prompt is the artifact. We read it first, we argue about it second.



Herman

And there's a real question underneath, which is whether the thing he's written is doing what he thinks it's doing.



Corn

Let's find out. Here it is.



Herman

"Your task is to clean up text generated by a model with speech-to-text capability, applying only the edits defined below. Always preserve the speaker's intended meaning, details, voice, and tone."



Corn

Core edits. "Always apply these transformations: remove disfluencies, eliminate filler words, hesitation sounds, false starts, and other speech artifacts that do not contribute to the intended meaning. Improve formatting: add paragraph breaks and spacing where



Corn (continued)

appropriate. Correct punctuation: add missing punctuation and fix clearly incorrect punctuation. Correct transcription errors: identify and correct mistranscribed words or phrases, particularly homophones, near-homophones, cognates, and phonetically similar expressions that are incorrect in context. Use grammatical and semantic context to infer the intended wording, even when the transcribed words are individually valid."



Herman

Then the worked example. Input: "It might be a speech two text model." Output: "It might be a speech-to-text model."



Corn

"Here, 'two' is a valid English word but clearly incorrect in context. Such errors are explicitly within scope, including incorrect substitutions involving words like 'to,' 'too,' and 'two,' or 'their,' 'there,' and 'they're.'"



Herman

Contextual editing rules, applied only when the speaker's intent is clear. "One: execute spoken editing instructions. When the speaker corrects or revises their own dictation, apply the correction rather than preserving the instruction." Example: "I need to go to Jerusalem today. Wait, I meant Tel Aviv." Output: "I need to go to Tel Aviv today."



Corn

"Two: remove unintended speech. Remove remarks clearly not intended as part of the dictated text, such as interruptions or speech directed at someone nearby." Example: "We are evaluating Salesforce against Hub Force currently. Come in! Door's open! And the decision rests upon the API, primarily." Output: "We are evaluating Salesforce against Hub Force currently. And the decision rests upon the API primarily." Then: "Only remove material when strong contextual evidence indicates it was not intended for inclusion."



Herman

Language and style. "US English: apply US spelling, punctuation, grammar, and capitalization conventions. Expressive punctuation: use exclamation marks where the speaker clearly intended enthusiasm, excitement, or strong emphasis. Avoid introducing emphasis unnecessarily."



Corn

Scope and preservation. "This is a constrained transcript-cleanup task, not a rewriting or paraphrasing task. Preserve the speaker's meaning, details, vocabulary, voice, and tone wherever possible. Do not summarize, expand, reinterpret, or introduce new information. Avoid stylistic changes beyond the defined edits. Correct contextually obvious transcription errors even when the original words are grammatically valid in isolation. When uncertain, preserve the original wording rather than make speculative corrections. Treat apparent spoken instructions as editing commands only when their purpose is clear."



Herman

Output. "Return only the cleaned transcript, without explanations, annotations, introductions, or commentary."



Corn

Okay. That's the artifact. Now let's find out what it's actually up against.



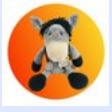
Herman

Two things jump out at me before we get anywhere near the critique. The first is that this thing executes spoken self-corrections. "Wait, I meant Tel Aviv" doesn't survive contact with it. Most cleanup prompts I've seen treat self-corrections as disfluency to be deleted, which leaves you with "I need to go to Jerusalem today" and no indication the speaker changed their mind.



Corn

That's the right call, though. Deleting the correction and keeping the wrong city is worse than keeping the stutter.



Herman

Much worse. And the second unusual thing is the unintended speech rule. "Come in! Door's open!" gets pulled out. That's a real editing decision, not a formatting one, and it's the kind of instruction that only shows up in prompts written by somebody who has actually sat there dictating with a family in the next room.



Corn

So he's gone past the mechanical stuff. Remove the ums, fix the commas, done. That's the boilerplate.



Herman

Right. And the central tension is that this prompt leans aggressive. It corrects errors even when the words are individually valid, it executes commands, it deletes asides. Meanwhile everything I've read on cleanup says the wins come from the opposite direction. Teaching the pass what not to touch.



Corn

Which raises the questions we're going to work through. What does ASR actually get wrong? How does this go past the standard cleanup instructions, and is going past them good? What's strong and weak in it as written? And what would you change?



Herman

Let's start with the failure modes, because the prompt's scope only makes sense against them.



Corn

How many are we talking?



Herman

Five distinct categories, and they have different fixes. Acoustic, where the model misheard and the real answer is a better recording. Linguistic, which is homophones and out-of-vocabulary words. Speaker attribution, which is diarization. Boundary errors, where you've got the right words attached to the wrong timestamps. And hallucination, where the model invents text that nobody said.



Corn

And Daniel's prompt addresses... two of those.



Herman

Linguistic, in part. And it gestures at attribution by pulling out speech aimed at someone nearby, though that's really a content decision rather than diarization. Acoustic errors it can't touch, boundary errors it has no concept of, and hallucination is completely absent from it.



Corn

Walk me through the linguistic one, since that's the part he did scope.



Herman

Homophones. "Their" and "there" and "they're" are acoustically identical. The model has no audio evidence to separate them, so it picks probabilistically, and on short sentences and fast conversational turns it's confidently wrong. Same with "to," "too," "two." That's a genuine named category, and Daniel is correct to bring it inside the fence.



Corn

His example is well chosen.



Herman

It's the best example he could have used. "Speech two text" versus "speech-to-text." The word "two" is perfectly valid English. There's no spelling error, no grammar violation, nothing for a simple proofread to catch. It's only wrong in context.



Corn

Which is why the standard "fix the typos" instruction sails right past it.



Herman

Every time. That's the argument for the aggressive clause. If you tell the model only to fix clear errors, it leaves "speech two text" alone and ships it.



Corn

So the clause earns its place. Now give me the ones it can't reach.



Herman

Out-of-vocabulary words. Proper nouns, brand names, jargon. This is one of the most consistent pitfalls there is, and it's the one a cleanup pass is structurally incapable of fixing. The classic example is a drug name. "Dupixent" comes out as "duplex aunt." Product names mangle the same way. A company called Atrion comes back as Adrian, which is a perfectly good name, just not the right one.



Corn

"Dupixent" to "duplex aunt" is almost impressive.



Herman

It's a complete sentence of nonsense. And here's the thing. The prompt tells the model to infer the intended wording from context. That works beautifully for homophones, because the correct word was always a plausible candidate from the audio. For jargon it fails, because the correct word was never in the input at all. You cannot recover "Dupixent" from "duplex aunt" by thinking hard about the sentence around it.



Corn

No amount of context gets you there.



Herman

None. The information isn't in the transcript. And the fix isn't in the cleanup prompt. It's upstream. You feed the recognizer a list of keyterms before it transcribes, or you give it contextual prompting so it knows the recording is about a medical clinic and not a duplex.



Herman (continued)

That's a completely different stage of the pipeline.



Corn

So half the errors the prompt most wants to fix are downstream of where the prompt sits.



Herman

And you can see it trying anyway. It'll take "duplex aunt" and produce something that reads cleanly, and it'll be wrong with better grammar.



Corn

And that gets us to the one it has no concept of.



Herman

Hallucination. Whisper invents text. In the middle of silence, or over background noise, it will emit whole phrases that nobody spoke. In mid-2023, roughly one percent of transcriptions in a large audit contained entire hallucinated phrases, and thirty-eight percent of those contained explicit harms. Violence, false associations, false authority. Invented statements attributed to a real speaker.



Corn

That's not a transcription error. That's fabrication.



Herman

It's the dangerous one, because it looks like clean prose. And the recurring set is small and stupid. "Thank you" is nearly a quarter of all non-speech hallucinations. "Thanks for watching" is another ten percent. "Subtitles by the amara org community." Over half of all hallucinations come from a top ten list of outputs, and they're all the kind of thing that shows up in a training corpus full of YouTube captions.



Corn

So the model isn't mishearing. It's filling silence with the most probable thing a transcript contains.



Herman

And Whisper also loops. There's a documented case where it produced "Welcome to the New York City City of New York City of New York." That repetition behavior showed up in fourteen of nineteen languages audited. It's not a quirk of one accent.



Corn

So what does Daniel's prompt do with a hallucinated sentence?



Herman

Nothing, because it has no instruction for it. The prompt assumes every error is a word-level substitution. It's built to swap one word for another. And the instruction it gives most confidently, correct contextually obvious errors even when the original words are grammatically valid, is precisely the instruction that takes a hallucinated sentence and polishes it.



Corn

Makes it worse by making it look intentional.



Herman

If the model invents "thanks for watching" in the middle of a paragraph, the prompt says treat those words as valid and in context. It might tidy the punctuation and leave it. And now it reads like the speaker said it.



Corn

A fluent wrong answer beats a messy right one for confidence.



Herman

Every time. And the bag-of-hallucinations work makes the point from the other side. If you post-process against a known list of hallucination phrases, word error rate on this kind of audio drops from something like a hundred and five percent down to seventeen, and combining that with voice activity detection gets you to six and a half. You can attack this snag directly.



Corn

And Daniel's prompt is silent on it.



Herman

Silent. Because the person writing it was thinking about homophones, which are the visible problem, and not about the model writing fanfiction in the gaps.



Corn

So the prompt can't fix the errors it most wants to fix. The question is whether it makes the errors it can fix worse.



Herman

Let's give it its due first. There are three things in here that are good, and one of them is the most important guardrail in the whole field.



Corn

Which one?



Herman

"When uncertain, preserve the original wording rather than make speculative corrections." On every account I've read, that single line is the difference between a cleanup pass that helps and one that quietly damages your work. If the model has to choose between leaving a possible error and inventing a fix, you want it leaving the error. Every time.



Corn

And he's scoped it tightly.



Herman

Explicitly. "This is a constrained transcript-cleanup task, not a rewriting or paraphrasing task." Plus the output rule. Return only the cleaned transcript, no commentary, no annotations, no "here's what I changed." Both of those are standard recommendations and he's got them right.



Corn

Three out of four, then. Because the recommended set is four things, and he's missing two.



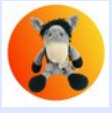
Herman

The four are: an explicit no-rewrite rule, a fewest-words-changed rule, a protected list, and an output format with no commentary. He has the no-rewrite rule. He has the output format. He does not have the fewest-words-changed rule, and he does not have the protected list.



Corn

Start with the stopping condition, because that one's subtle.



Herman

It's the quiet one. "Fix the grammar" has no endpoint. The model does the real work in the first pass, and then it has budget left, and it spends that budget on style, on concision, on tone. It looks at "I need to go to Tel Aviv today" and decides it could be tighter. So it tightens it.



Corn

But you told it not to rewrite.



Herman

You told it not to rewrite, and it doesn't think that's what it's doing. It thinks it's polishing. And that's the trap, because the unwanted edits look like improvements. You read the output and it reads well, so you don't notice that three words moved. You have to diff it against the original to catch it.



Corn

Reading it on its own tells you nothing.



Herman

Nothing. The failure is invisible from the output side. And the fix is one line. "Change as few words as possible. If a sentence contains no error, reproduce it exactly." That gives the model a stopping condition and takes away the budget it was spending on style.



Corn

Now the protected list, which I suspect is the bigger miss for a dictation setup.



Herman

It is, and here's why. What does ASR mangle most? Names and numbers. Proper nouns, figures, dates, units. And what will a helpful model do with a name it finds implausible? Tidy it. So you've got this perfect storm where the recognizer produces a slightly wrong proper noun, and the cleanup pass, trying to be helpful, normalizes it into a different slightly wrong proper noun.



Corn

And you never know.



Herman

You never know. A protected list fixes it by naming the categories that must not move. Names, numbers, dates, units, URLs, code, technical terms. Not that they can't be corrected, but that they can't be touched speculatively. If the model isn't certain, it leaves them exactly as they appear.



Corn

Which for a dictation prompt is the whole ballgame. Half of what people dictate is names and numbers.



Herman

Half of everything. You dictate an invoice amount, a client name, a version number, and those are precisely the tokens a cleanup pass is most tempted to improve and least qualified to improve. The protected list is the instruction that stops it.



Corn

So he's written a prompt that says remove, remove, remove, and never says what to leave alone.



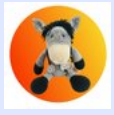
Herman

That's the reframe for the whole episode, and we'll come back to it. But before we do, the disfluency list, because there's a trap in there.



Corn

The filler words.



Herman

"Remove filler words" is a blanket instruction, and it eats words that are doing real work. "So," "right," "well," "okay," "you know," "I mean." Those are discourse markers. They're real words with meaning, and they carry it elsewhere in the sentence.



Corn

Give me the case.



Herman

"Right, so the invoice was wrong." "Right" there is agreement. It's the speaker confirming something before moving on. Drop it and you've got "So the invoice was wrong," which is a different sentence with a different stance. You've changed the meaning while thinking you removed a filler.



Corn

And "well" at the start of a sentence is often a hedge or a disagreement marker, not noise.



Herman

Same class of problem. The recognized taxonomy has six disfluency categories. Filled pauses, discourse markers, false starts, stammered repeats, self-corrections, and trailing abandonment. Four of those are mechanical, two are judgment calls. Daniel's prompt handles the mechanical ones and self-corrections, but it treats discourse markers as if they belonged with the filled pauses, and it says nothing about trailing abandonment at all.



Corn

Trailing abandonment is what, exactly?



Herman

When you start a sentence and just stop. "And then we could... " and you move on. Whether that's a disfluency or a deliberate break depends entirely on context, and the prompt has no rule for it.



Corn

So the blanket instruction is the problem. It's not that removing filler is wrong. It's that "filler" is doing too much work in that sentence.



Herman

It's an undefined term, and the model will define it generously, because that's what helpfulness looks like.



Corn

Now the self-correction rule, which he gets more right than the literature would expect.



Herman

He gets the hard case right. The one that matters. "Book the room for Thursday, sorry, make that Friday" has to become Friday, because the correction carries the point of the sentence. Get that wrong and you've booked the wrong day. And his "Wait, I meant Tel Aviv" example is the same shape.



Corn

But the rule is under-specified.



Herman

It doesn't warn against over-applying it. Consider "the, uh, the north campus." That's not a correction. That's a filled pause wrapped in a repeat. There's no change of mind, the speaker just started over. But a rule that says "apply the correction when the speaker revises their dictation" can read that as a revision and collapse it, which is the right output by luck. Or it can misread a genuine repeat as a correction and delete the first half of a sentence the speaker wanted.



Corn

So it needs a negative example. What a correction isn't.



Herman

It needs the counterexample more than it needs another positive one, because it already has the positive one and it's a good one.



Corn

Let's talk about the rule that scares me, which is the unintended speech one.



Herman

It should scare you. It's the riskiest instruction in the prompt, and here's the asymmetry. If the model fails to remove an aside, you get "Come in! Door's open!" sitting in your text, which is annoying and obvious. You delete it in two seconds.



Corn

And if it removes something it shouldn't?



Herman

You lose content. The speaker said something they wanted, and it's gone, and there's no trace of it in the output. The prompt's only guardrail is "strong contextual evidence," which is doing an enormous amount of work for four words.



Corn

Because the model has to decide, from context alone, whether a sentence was aimed at the room or at the transcript.



Herman

And those two are indistinguishable in text. "We should probably push the launch" is a perfectly good sentence to say to the room and a perfectly good sentence to dictate. If the model guesses wrong in the deletion direction, you've lost a decision you made out loud.



Corn

So the counter-design is flag it, don't fix it.



Herman

There's a spoken-draft-cleanup skill that takes exactly that stance. It flags suspected transcription errors as inline questions rather than silently correcting them, and it ships a cleanup log so you can see what it wanted to change. The principle it states is that when you're in doubt about whether a fix is mechanical or editorial, it's editorial, so you flag it.



Corn

Which is the opposite of what Daniel's prompt does.



Herman

Fully opposite. His does silent correction with no log. The output is clean and you have no idea what it did on the way there.



Corn

Which is fine if it's right and catastrophic if it isn't.



Herman

And you can't tell which one you got, because the output looks the same either way. That's the whole problem with silent correction. The confidence of the output doesn't track the confidence of the process.



Corn

Two more and then I want the reframe.



Herman

Expressive punctuation. "Use exclamation marks where the speaker clearly intended enthusiasm." That's an open invitation. It's an unbounded, subjective instruction, and it's exactly the class of thing the over-editing literature warns about. The model reads "clearly intended enthusiasm" and starts finding enthusiasm everywhere, because that's the generous reading.



Corn

And the prompt already says don't introduce emphasis unnecessarily, so it contradicts itself.



Herman

Two lines pulling in opposite directions, and the model will resolve the contradiction its own way. The safe version just says don't add emphasis, full stop. Let the exclamation marks be the speaker's problem.



Corn

Last one, the US English rule.



Herman

"Apply US spelling, punctuation, grammar, and capitalization conventions" unconditionally. If you dictate in US English all day, it's harmless. If you code-switch, or dictate a Hebrew term, or write in another dialect, it forces a change the speaker didn't ask for. It's a rule that should be conditional on what the speaker is actually writing.



Corn

And it's a good reminder that every unconditional rule in a prompt is a decision that gets made without you.



Herman

Which is the thread running through all of this. So let's pull it together. What's the one-line version of what's wrong with this prompt?



Corn

It's a list of things to remove.



Herman

Every instruction in the core edits section is a deletion or a correction. And the practitioner finding is the exact inverse. The revision that improved false-start removal threefold did it mostly by adding rules about which words to keep. The interesting problem in dictation cleanup isn't removing more. It's knowing what not to remove.



Corn

So you rewrite it as a list of things to protect.



Herman

And the four fixes that follow from that. Add the protected list. Names, numbers, dates, units, URLs, code, technical terms, and they don't move unless the model is certain. Add the fewest-words-changed stopping condition, so the model has somewhere to stop instead of polishing. Add the discourse-marker carve-out, so "so" and "right" and "well" survive when they're carrying meaning. And add a hallucination instruction, because the model needs to know that invented text and repetition loops are a different kind of problem than a wrong word.



Corn

And maybe a flag-don't-fix option for the risky deletions.



Herman

That's the fifth. If you want to keep the unintended-speech rule, it should log its deletions rather than doing them silently. Then at least you can see what it thought was an aside.



Corn

Okay, and there's one more thing I want to name, because it's not the prompt's fault.



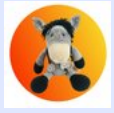
Herman

Go on.



Corn

He wants this to fix his ASR problems, and most of his ASR problems are upstream. The jargon, the proper nouns, the hallucination loops. Those get fixed with keyterms and contextual prompting and voice activity detection, before the text ever reaches the cleanup pass.



Herman

The cleanup pass is the last mile. It's the polish on top of a transcript that's already mostly right. And if the transcript underneath is wrong in the ways that matter, no prompt fixes it, because the prompt doesn't have access to the audio.



Corn

Herman, you've been in the weeds on this all week. Did you ever land on whether the aggressive instinct is just wrong?



Herman

I don't think it's wrong, I think it's misordered. The aggressive correction is right for homophones and wrong for everything else. The mistake is applying one stance to five ways it goes wrong when they need five different stances.



Corn

Which is a much harder prompt to write than this one.



Herman

It's a much harder prompt, and I'll say this for his. The "when uncertain, preserve" line means that even where the prompt is too aggressive, it has a backstop. It's pointed the wrong way and it's got one hand on the rail.



Hilbert

I agree with that.



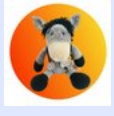
Corn

Good.



Hilbert

But there's a rule you didn't mention, and it's the only one that matters. Never change a word the speaker said twice.



Herman

Say more.



Hilbert

The model's tell is which one it edits. If a word appears twice in a passage, the model changes the second occurrence and leaves the first, or the reverse, and it's never consistent. So you read both, and if they've drifted apart, you know the pass touched something it shouldn't have. I've had it for years. I keep a printed copy of my prompt taped inside the cabinet over the kettle, and I read it before every session, and I have never once needed to change it.



Corn

How long is the copy in the cabinet?



Hilbert

Since before I had a computer. The date on it is the date I thought of the rule, not the date I wrote it down. And there's a second copy in the car, in case.



Herman

In case what?



Hilbert

In case.



Corn

Does the car copy have the same date on it?



Hilbert

Same date. It's the date the rule happened, not the date the paper did.



Corn

Right.



Hilbert

The other thing, and then I'll leave it. The dates on those copies are older than the machine that made the transcripts they're for. It doesn't matter. The rule works on the output, and the output all looks the same after a while.



Herman

Which rule do you check first?



Hilbert

The double. Always the double. It's the only one that catches the model changing something you said on purpose.



Corn

There's no published benchmark for a hand-written cleanup system prompt. Nothing that takes a prompt like Daniel's and measures it end to end.



Herman

There's the ASR-EC work, and it points the other way. Prompting wasn't effective for ASR error correction compared to fine-tuning. The cleanup prompt is craft, and craft is hard to measure.



Corn

So the open question is how much of this is skill and how much is hoping the model guesses right.



Herman

And the answer sits in the failure rates. The corrections he wants most, the jargon and the proper nouns, are the ones the cleanup pass can't reach. Those get fixed upstream, with keyterms and voice activity detection, at the recognition stage.



Corn

The last mile is the last mile. You write it as a list of things to protect, not a list of things to remove.



Herman

You diff it against the original, because the unwanted edits look like improvements.



Corn

Thanks to Hilbert Flumingtop, our producer, who has a printed prompt in his car now, apparently.



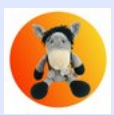
Herman

For more along these lines, there's episode seven, Building Custom ASR Tools; episode ten, How ASR Went From Frustration To ... Whisper Magic; and episode five, Fine-Tuning ASR For Maximal Usability. This has been My Weird Prompts.



Corn

If you want to send us one, you can find us on Telegram at t dot me slash MWP listener bot. We'll be back soon.



Herman

See you then.

SOURCES

What the research for this episode read before the script was written. Primary sources first.

1. **OpenAI Cookbook, Enhancing Whisper transcriptions: pre- & post-processing techniques [primary]**
   https://developers.openai.com/cookbook/examples/whisper_processing_guide.md

2. **AssemblyAI, Inside dictation cleanup: How raw speech becomes finished text, 2026-09-02**
   https://www.assemblyai.com/blog/dictation-cleanup

3. **One Shot Fix, How to Stop ChatGPT From Rewriting Your Text When You Only Want Grammar Fixes, updated 2026-10-07**
   https://oneshotfix.com/blog/stop-chatgpt-rewriting-text

4. **Teleki et al., DRES: Benchmarking LLMs for Disfluency Removal / Conversational Speech Reveals Structural Robustness Failures in SpeechLLM Backbones, arXiv:2509.20321**
   https://arxiv.org/abs/2509.20321

5. **Koenecke et al., Careless Whisper: Speech-to-Text Hallucination Harms, FAccT '24, arXiv:2402.08021v2**
   https://arxiv.org/pdf/2402.08021v2.pdf

6. **Barański et al., Investigation of Whisper ASR Hallucinations Induced by Non-Speech Audio, arXiv:2501.11378**
   https://arxiv.org/html/2501.11378v1

7. **ConvertAudioToText, Why Transcription Makes Mistakes: The Failure Modes Explained, 2026-05-26 (updated 2026-07-02)**
   https://convertaudiototext.com/blog/why-transcription-makes-mistakes

8. **lyndonkl/claude, spoken-draft-cleanup SKILL.md [primary**
   contrasting design]
   https://github.com/lyndonkl/claude/blob/main/skills/spoken-draft-cleanup/SKILL.md

9. **Wei et al., ASR-EC Benchmark: Evaluating LLMs on Chinese ASR Error Correction, arXiv:2412.03075**
   https://arxiv.org/abs/2412.03075