

原文やメモから発信を作る

Xポストと記事 制作エージェント

翻訳ポストと記事執筆を
手元のスキルで始める

実践ガイドと配布スキル

XのURL取得には利用者のAPI設定が必要です



素材を渡してポストと記事を仕上げる

短い翻訳ポストも詳しい記事も 一言の依頼から始める

原文やメモから日本語の完成稿へ



企画書や訴求軸の入力は不要です。口調や長さは後から調整できます。

作りたいものを一言で伝える

Orbit

二つのスキルを同じキットに同梱

01

翻訳ポスト

元投稿から日本語の1本へ
「日本語ポストにして」

02

メモから記事

手元の材料を読み物へ
「このメモで記事を書いて」

03

翻訳解説記事

海外記事の要点を解説
「この記事を日本語で解説」

ポストは x-translate 記事は x-write-article が担当します。

使い方は三つの操作

Orbit

企画書やアカウント設計を入力する必要はありません

01

原文を渡す

本文を貼り付けるか
同梱サンプルを指定します。

02

1本を受け取る

日本語の本文と
確認事項が返ってきます。

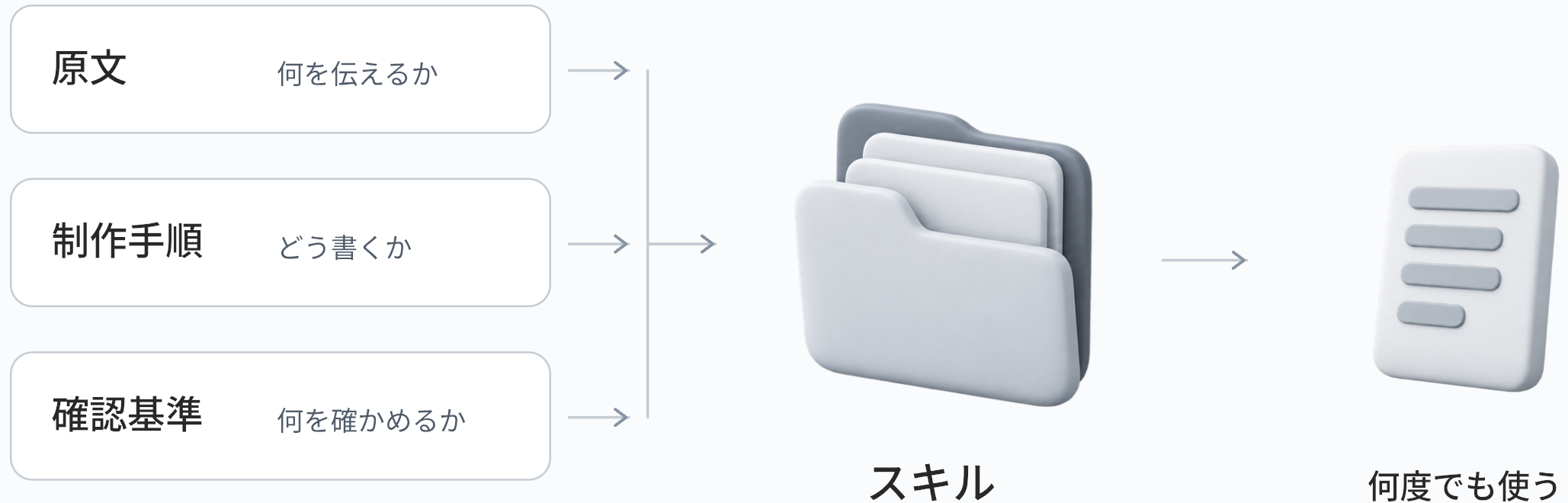
03

必要な所を直す

「短くして」などの
一言で調整します。

制作手順をスキルとして持ち運ぶ

毎回の依頼で繰り返す仕事と確認基準をまとめておく

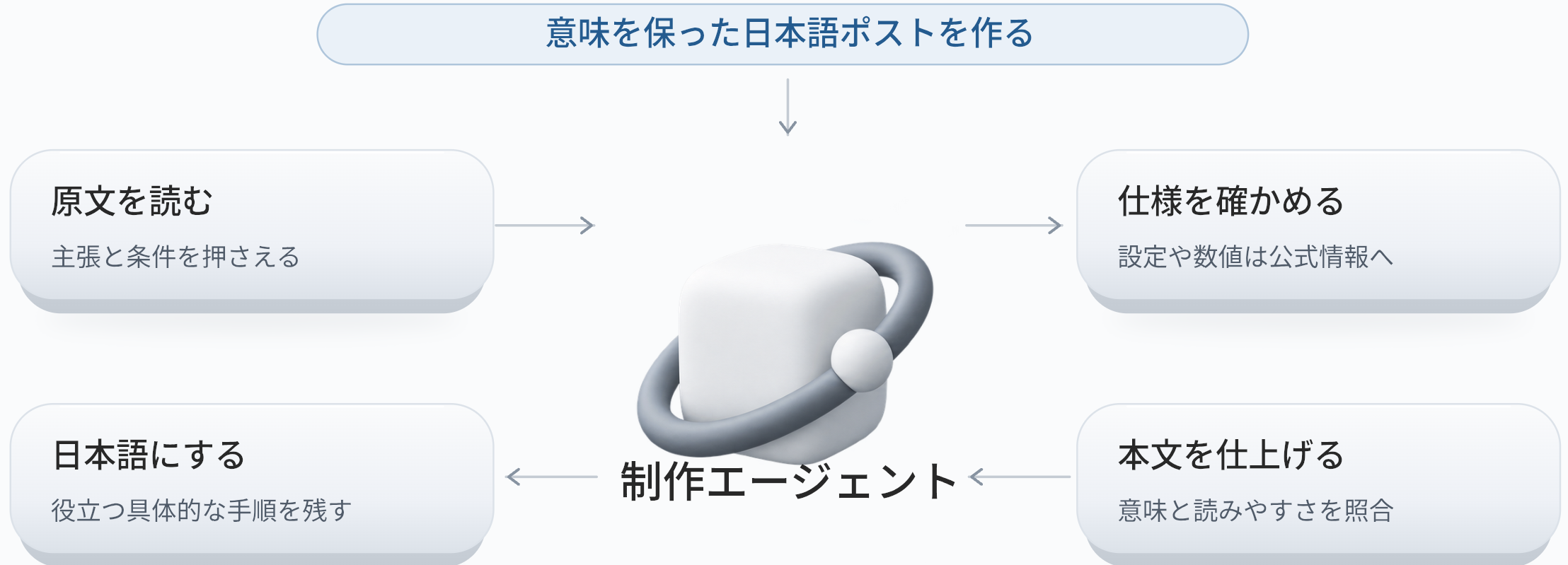


利用者は元投稿を渡します。スキルが手順と確認基準を参照して制作します。

配布キット x-translate / SKILL.md

エージェントが進める四つの仕事

原文を理解してから書き 原文に戻って確認する



不足があるときだけ内容を確認します。企画質問からは始めません。

配布キット x-translate / SKILL.md

原文から実際に作った1本

数字と対象範囲を残したまま日本語にまとめる

渡した原文

We tested a drafting assistant with 12 volunteers. Average time to prepare a first draft fell from 25 minutes to 16 minutes. The test covered first drafts only; fact-checking and publishing were not included. We have not measured long-term results.

生成された日本語ポスト

12人の協力者が参加した下書き支援ツールのテストで、初稿の作成時間が平均25分から16分に短縮。

対象は初稿の作成のみ。事実確認と公開作業は含まれず、長期的な結果は測定されていません。

教材用の架空例です。実在する研究や製品の実績ではありません。

数字と条件を一緒に残す

Orbit

速くなったという結論だけで終わらせない

01

数字

12人で試した
平均25分から16分

02

対象範囲

計測したのは初稿のみ
事実確認と公開は対象外

03

未検証

長期的な結果は
まだ計測していない

前ページの教材用原文から整理した例です。

推測を実証済みの話に変えない

誰の意見かと根拠の範囲を保つ

原文の要点

I suspect ... may benefit more ...

my impression from working with two
teams, not a controlled study

日本語で残すこと

役立つかもしれないという投稿者の意見

根拠は2チームでの経験に基づく印象

効果を確かめた実験ではない

教材用の架空例です。詳細は [samples/03-opinion.txt](#) を参照できます。

要約だけで終わらず使い方まで伝える

Orbit

読んだ後に何を試せるかが分かる本文へ

01

読む理由

どんな場面に役立つかを
冒頭で具体的に伝えます。

02

仕組みと操作

元投稿にある設定や手順を
意味が伝わる形で残します。

03

根拠と条件

仕様は公式情報と照合し
出典と確認内容を残します。

ZIPを開いて使い始める

初回は同梱の原文サンプルで動きを確認できます

01

ZIPを展開

x-translation-agent
フォルダーができます。

02

フォルダーを開く

Claude Codeまたは
Codexの作業場所にします。

03

制作を頼む

原文を指定して
日本語ポストを頼みます。

AI環境の契約とログインは事前に用意してください。

Claude CodeとCodexで使う

Orbit

同じ制作手順をそれぞれのスキルとして同梱



01

フォルダーを開く

展開したキットを作業場所に

02

原文を渡す

自然な言葉で制作を依頼

03

完成稿を読む

本文と確認事項を見て修正

ローカルのフォルダーを読み書きできる環境で使います。

仕様参照 [OpenAI Build skills / Claude Code Skills](#)

この一文と原文で依頼できる

指定がなくても本文の制作を始められる

コピーして使う依頼文

この原文から日本語のXポストを1本作って

ここに元投稿の本文を貼り付ける

サンプルを試す場合：samples/01-first-draft.txt を使って1本作って

同梱の原文は教材用です。実際の投稿には自分で選んだ素材を使います。

原文と並べて本文を直す

生成された preview.html をブラウザで開く

日本語ポストの欄

本文をその場で編集

「本文をコピー」でコピー

「本文を保存」でテキストを保存

原文の欄

参照した原文を表示

出典と著者を確認

未取得の内容や不明点を確認

HTMLの編集は元のpost.txtへ自動反映されません。手直し後は本文を保存します。

一言で好みの文章に整える

Orbit

必要な変更だけを続けて伝える

01

長さ

「条件を残したまま
短くして」

02

口調

「ですます調にして」
「少しやわらかくして」

03

表現

「この一文を
分かりやすくして」

意味や条件を変えずに修正します。毎回プロフィールを入力し直す必要はありません。

URLだけで使いたい場合

本文を取得する接続は必要になった時だけ設定する

取得手段が設定されている

元投稿のURLを渡します。

取得した本文を読んでから
制作します。

出典と取得時点を残します。

取得手段がない

元投稿の本文を貼り付ければ
制作を始められます。

本文を取得できないまま
内容を推測しません。

SocialDataの任意設定は同梱の案内を参照。取得サービスの契約と料金は利用者側で用意します。

画像や動画が主題なら中身も確認する

紹介文だけで内容を推測しない

制作に使える素材

読める画像や画像内の文章

動画の字幕や文字起こし

自分で確認した要点

不足している場合

「このプロンプトを試して」
という紹介文だけなら
画像内の文章も必要です。

読めていない内容は補わず
揃ってから制作します。

静止画像は同梱ツールで取得し利用環境の画像読解機能で確認します。動画の文字起こしは含みません。

完成した文章をファイルで残す

output/x-translate の中で本文と根拠を一緒に管理する

01

投稿本文

post.txt

コピーして使う完成稿

02

原文と確認メモ

source.json と review.md

出典や不明点を確認

03

プレビュー

preview.html

原文と並べて編集

プレビュー生成にはPython 3.9以上が必要です。本文制作だけなら不要です。

記事も素材を渡すところから

Orbit

長い企画入力をせず 完成稿を見て調整する

01

読む

元記事やメモの主張と
条件を押さえます。

02

書く

タイトルから具体例まで
一つの記事にまとめます。

03

確かめる

数字と帰属を照合し
紹介ポストも添えます。

記事の長さは素材に合わせます。短い材料を無理に引き延ばしません。

記事はこの一文で頼める

素材と作りたいものをセットで渡す

自分のメモから書く

このメモから日本語のX記事と
紹介ポストを作って

サンプル：

`samples/04-article-notes.txt`

海外記事から解説を書く

このURLから日本語の
翻訳解説記事と
紹介ポストを作って

ここに元記事のURL

記事本文と紹介ポストは別々のファイルに保存します。

要点の解説と全文翻訳を使い分ける

元記事をどこまで読んだかと何を作ったかを残す

URLからの翻訳解説

本文を取得して要点を理解

日本語の読者向けに組み立てる

出典と必要な条件を残す

提供された本文の全文翻訳

翻訳対象の本文を渡す

意味と条件を保って訳す

省略せず必要なら分割する

X記事は /status/ を含む共有投稿URLを使用。一般Web記事は利用環境の取得機能を使います。

メモから作った記事の一部

週報のメモから記事を作った例

渡したメモの要点

週報の材料は三つ

- ・ 今週進んだ仕事
- ・ 止まっている仕事
- ・ 来週取り組むこと

未確定を決定済みにしない

記事の見出しと依頼文

最初の依頼は一文でいい

このメモから週報を作って。
未確定のものは未確定と
分かるようにして。

教材用の架空例です。全文は [examples/04-original-article/article.md](#) で確認できます。

記事と紹介ポストを見て仕上げる

Orbit

output/x-write-article に本文と確認記録を保存

01

記事本文

article.md

タイトルから締めまで

02

紹介ポスト

post.txt

記事の読む理由を伝える

03

確認画面と出典

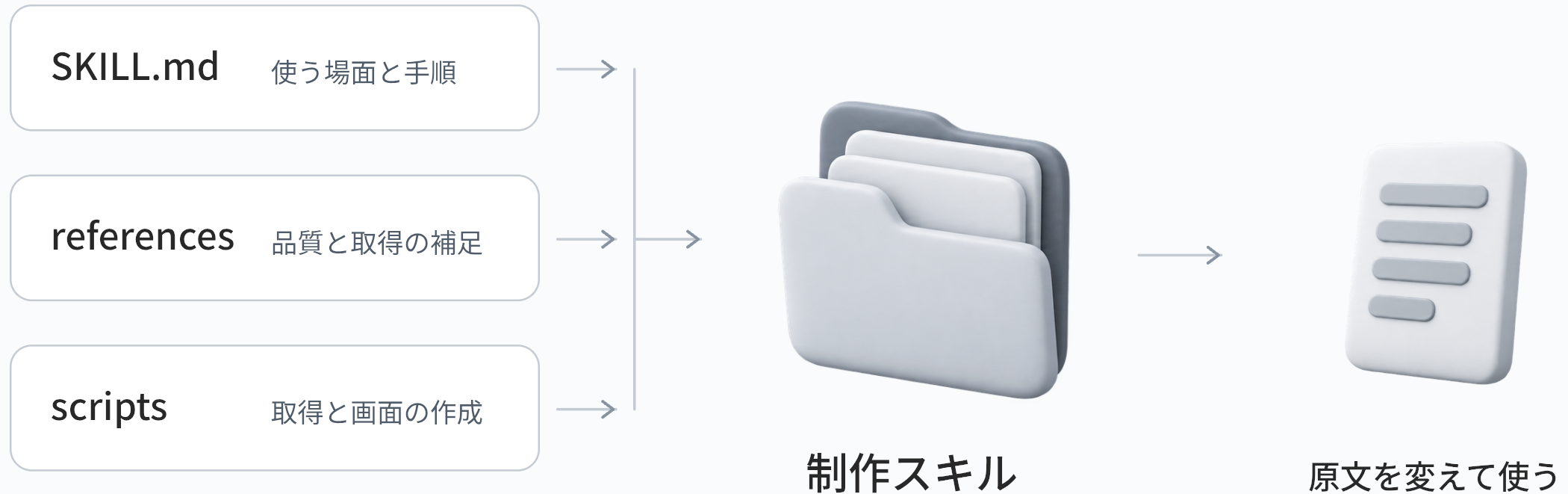
preview.html で編集

sources.json と review.md

画像制作とXへの公開は含みません。紹介文には公開後の実際の記事URLを追加します。

スキルは三つの部品でできている

いつ使うか どう進めるか 何を残すかをつなぐ



使い方をえる時は小さく修正し 同じ原文で結果を確かめます。

配布キット x-translate / x-write-article の実際の構成

同梱サンプルで最初の1本を作る

Orbit

ポストと記事を試したら自分の素材でも同じ手順を使う

原文を渡して作り 読み比べて整える

サンプルで試す

原文ファイルを指定する

意味を確かめる

数字と条件と帰属を見る

一言で修正

長さや口調を整える

Xでの公開操作は自分で行います。投稿画面でも文字数と出典を確認してください。

導入と仕様の参照先

環境の変更があった場合は公式情報を確認する

Codexのスキル

<https://learn.chatgpt.com/docs/build-skills>

Claude Codeのスキル

<https://code.claude.com/docs/en/skills>

URL取得を使う場合の補足

同梱スキルの [references/source-access.md](#)

教材内の英語サンプルは説明のために作成したものです。