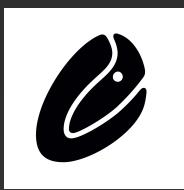


ニューラル言語モデルによる 個人最適な日本語入力システムの開発

未踏ITプロジェクト提案



三輪敬太

東京大学大学院
Turing株式会社



高橋直希

早稲田大学
株式会社CoeFont

日本語入力は1億人以上の
日本語話者にとって不可欠

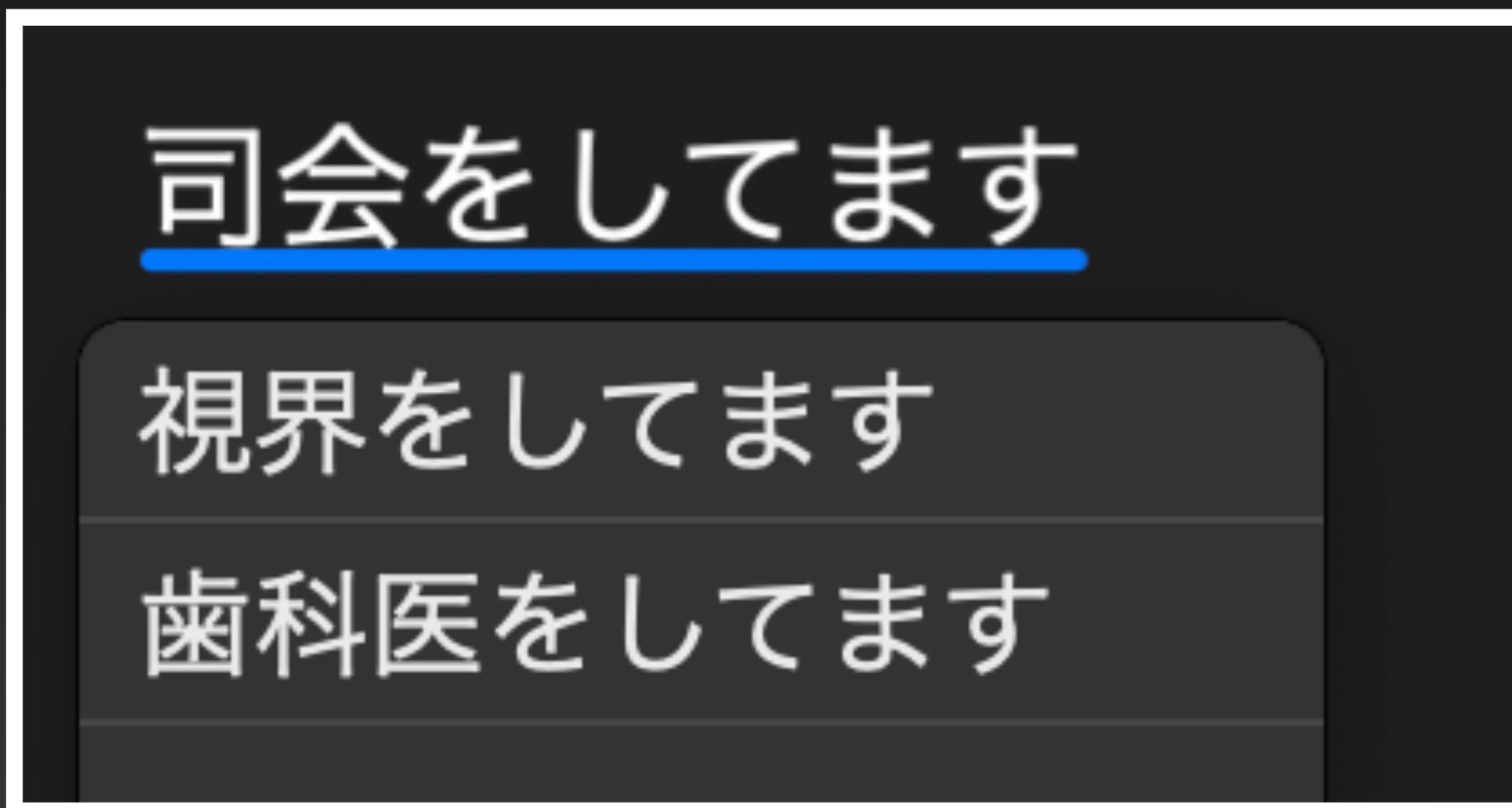
しかし、既存のシステムは
1億人の多様性を
サポートしきれていない

個人最適化の不足

現状の日本語入力システムの問題点

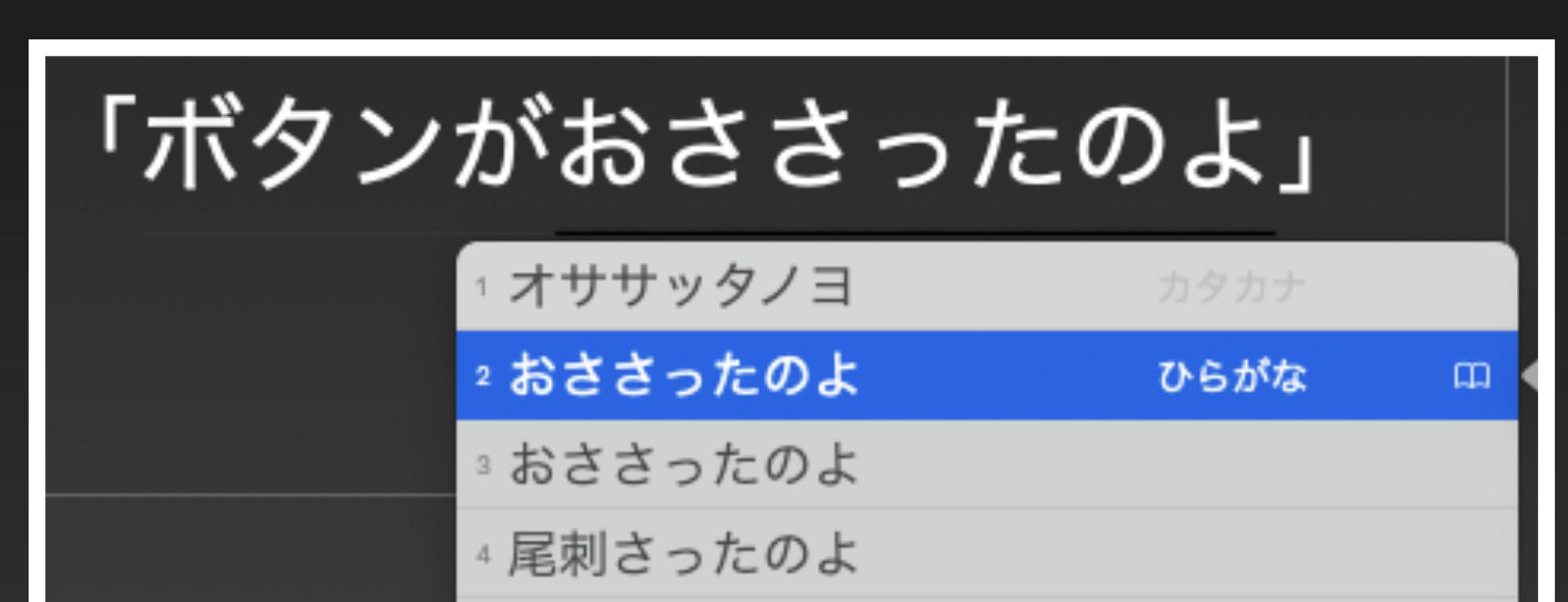
ユーザ文脈に合わない誤変換

例: 「歯科医をします」



方言に最適化されていない

例: 「ボタンが押さったのよ」

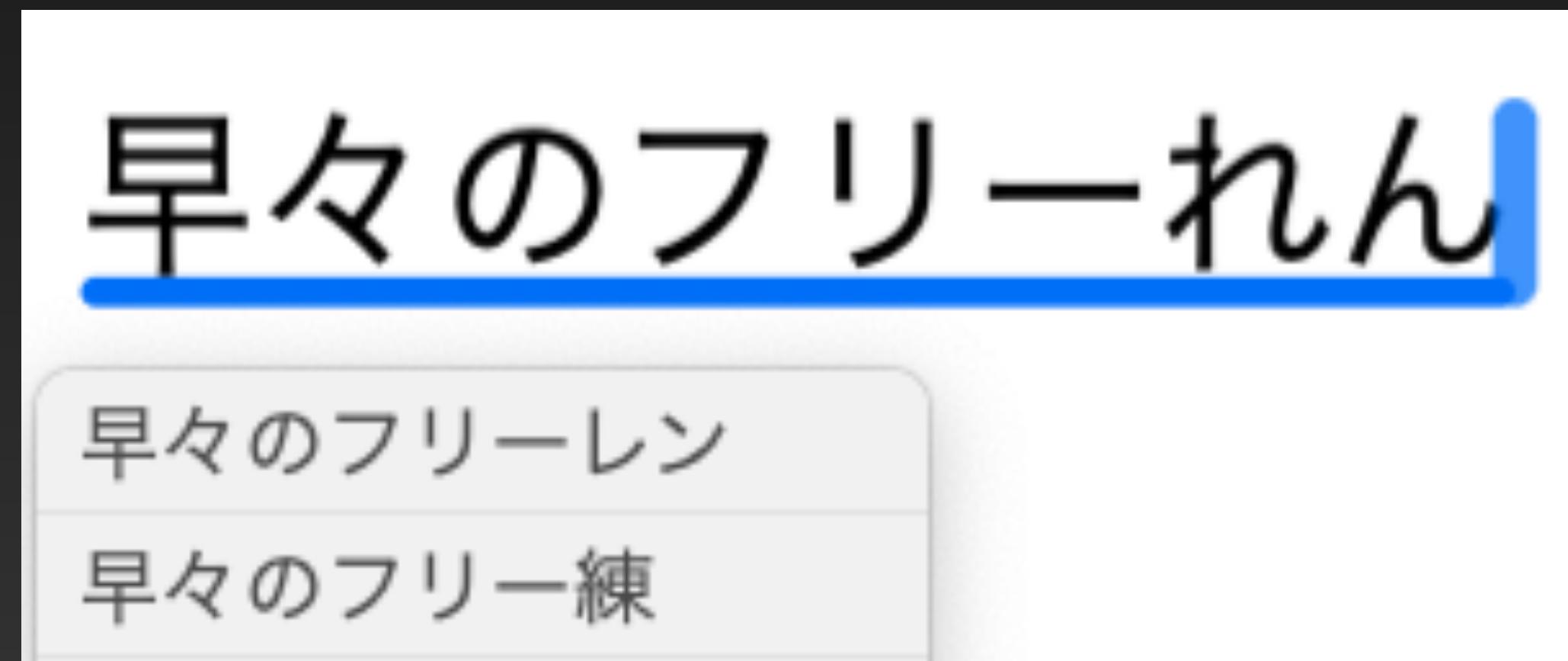


変換精度の限界

現状の日本語入力システムの問題点

新語・流行語への対応の遅れ

例: 「葬送のフリーレン」



意味関係の考慮が困難

例: 「猫を百匹飼いたい」



弱い予測入力

現状の日本語入力システムの問題点

- 単語や慣用表現の予測程度にとどまる
- 文脈を踏まえた予測入力はまだまだ不十分

3

ニューラル言語モデルを使えば、

4

もっと高度な予測入力が可能になる。

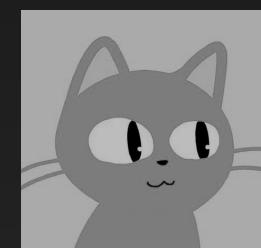
GitHub Copilotによる日本語コメント補完の例

最高の日本語入力？

最高の日本語入力

それぞれのユーザに最適化

高橋さんは数学科です



高橋

感情の家宝族に対して定義された情報



高橋

環上の加法族に対して定義された乗法

最高の日本語入力

一瞬で新語に対応



早々のフリー練めっちゃよかったです

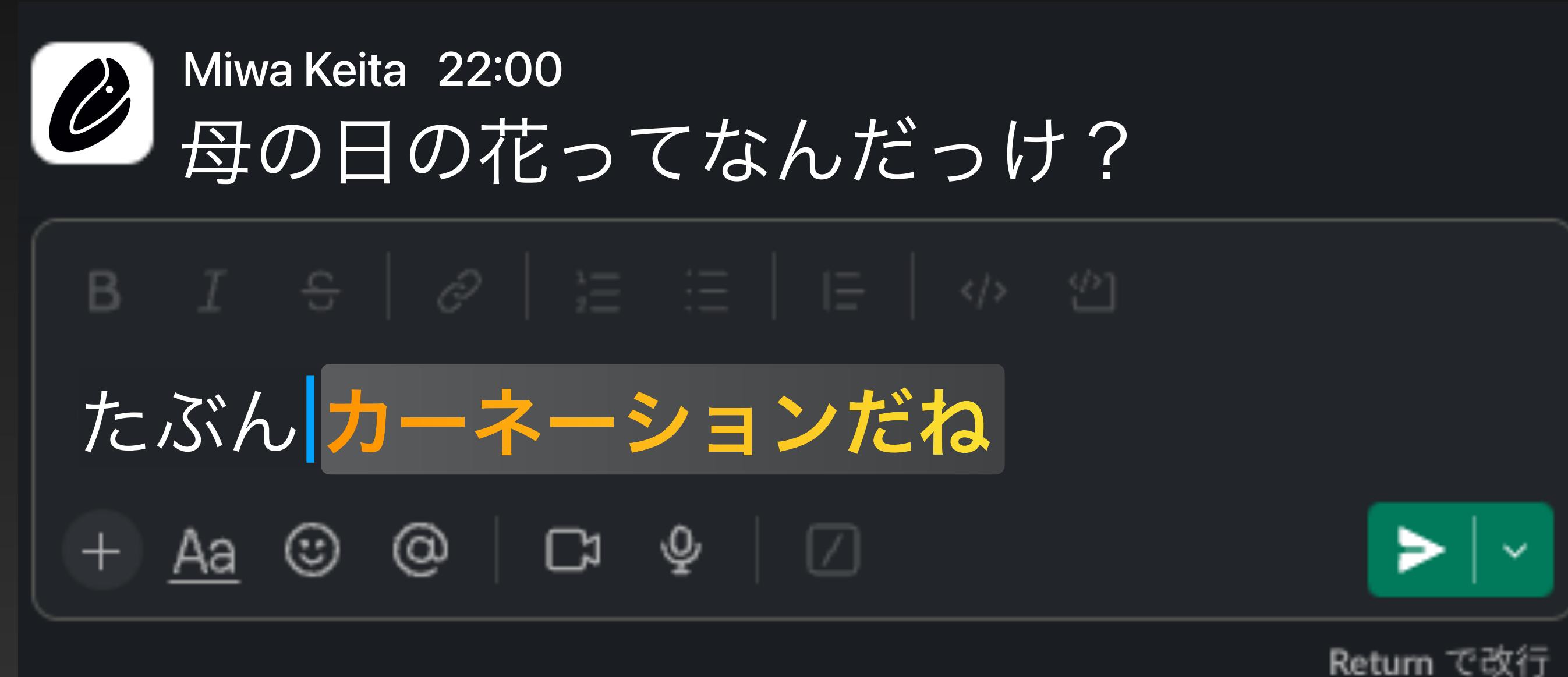


葬送のフリーレンめっちゃよかったです

葬送のフリーレンの画像

最高の日本語入力

文脈に応じた賢い予測入力



提案プロジェクト

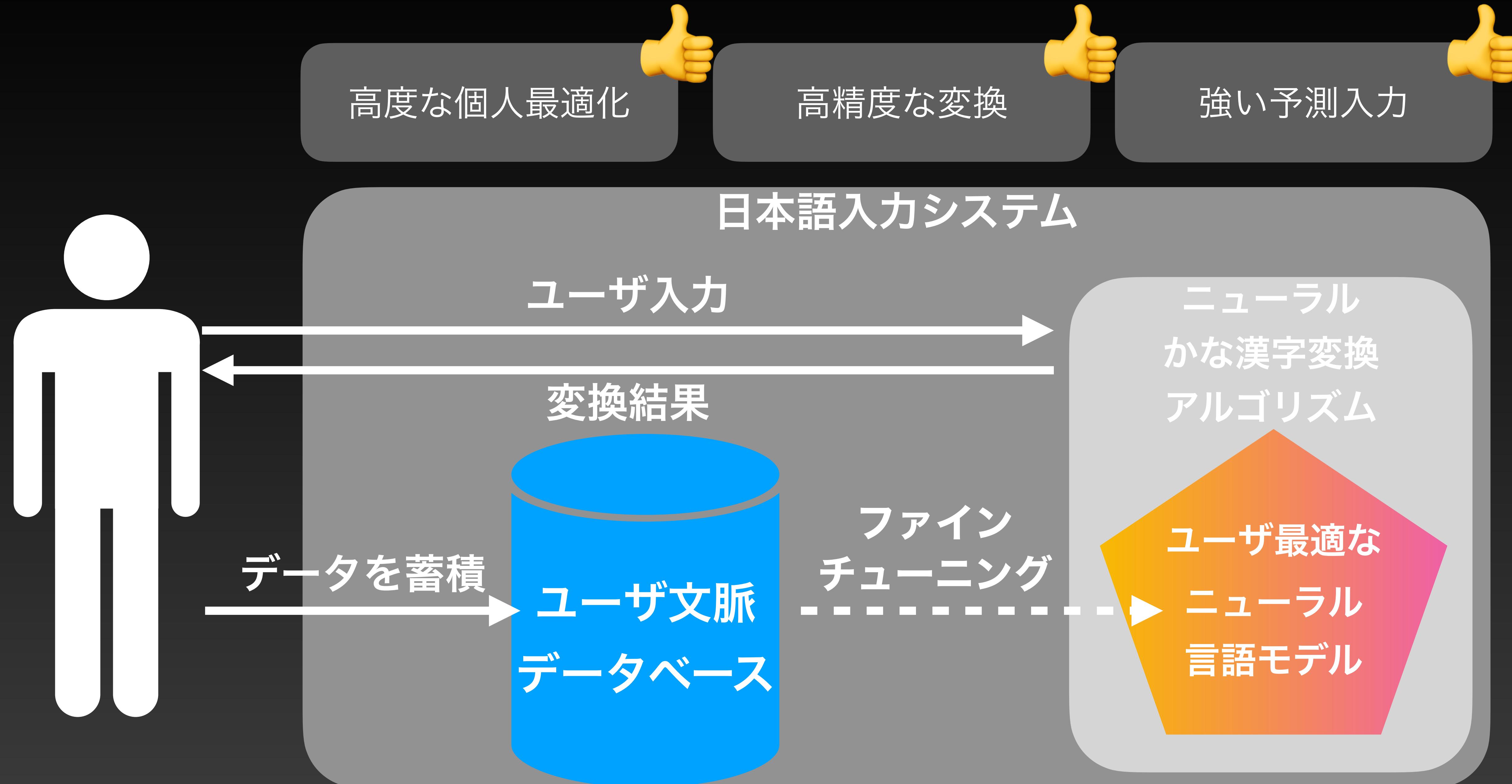
これらを実現する日本語入力の開発

無償で配布
オープンソース

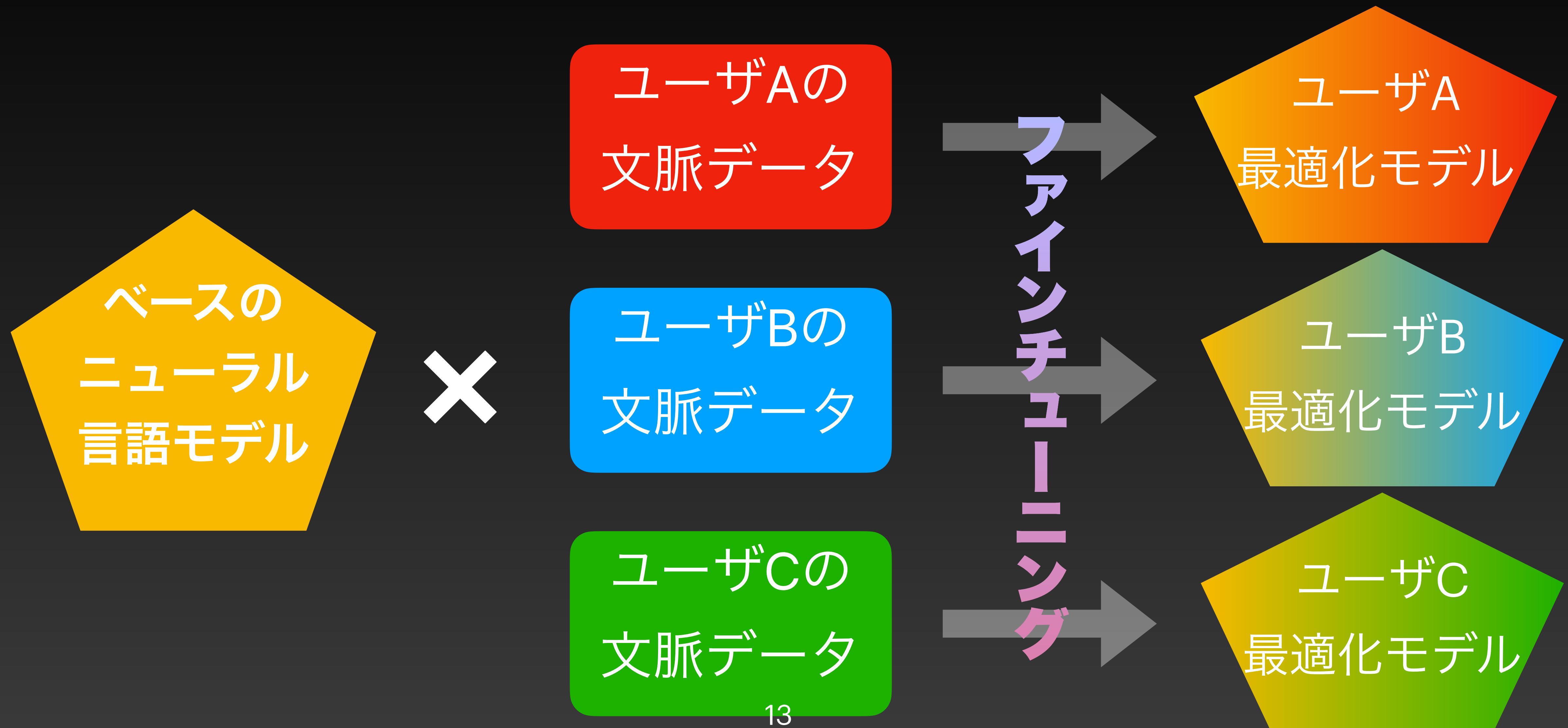
デスクトップ志向
(macOS)

完全に
ローカルで動作

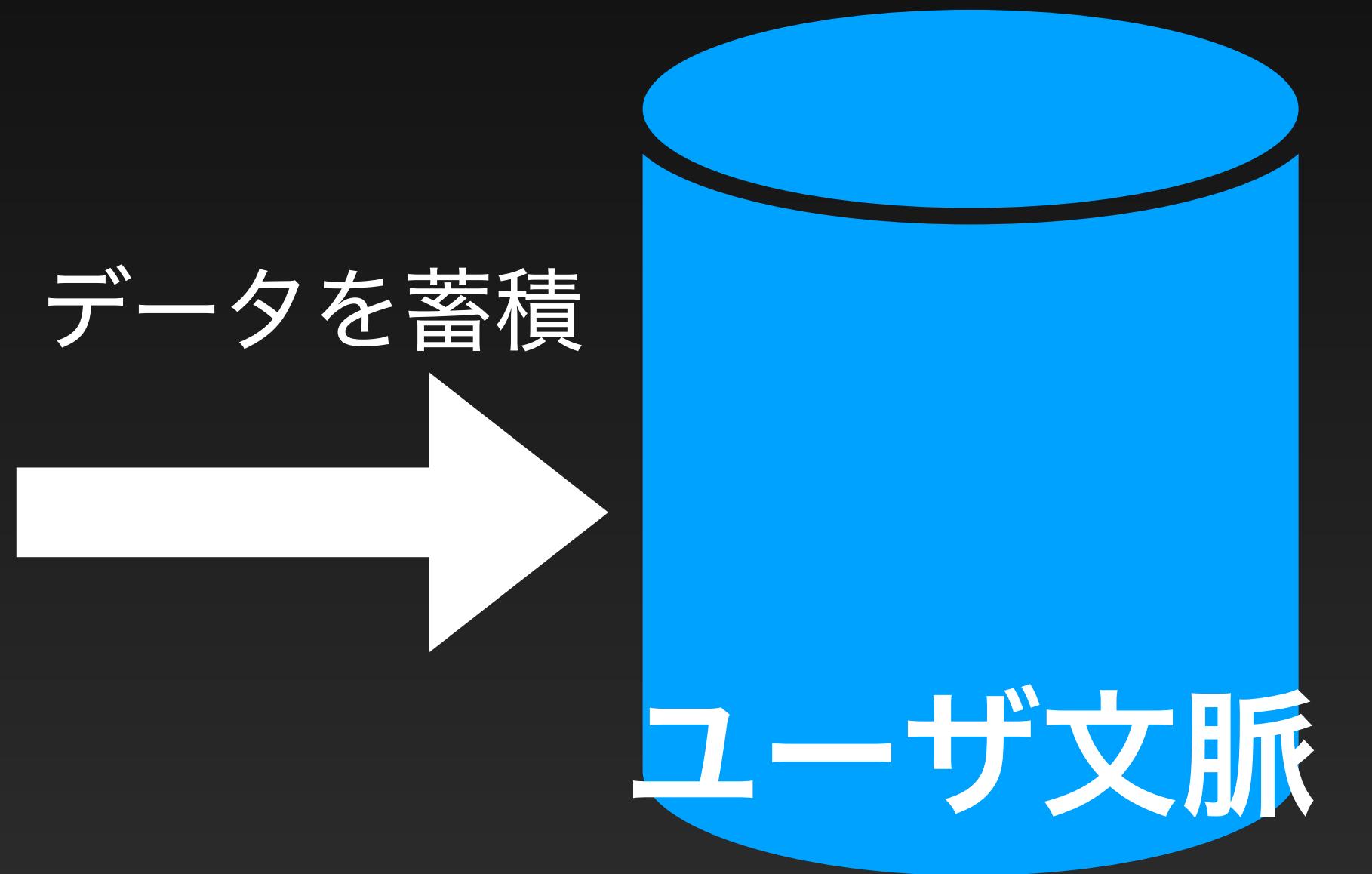
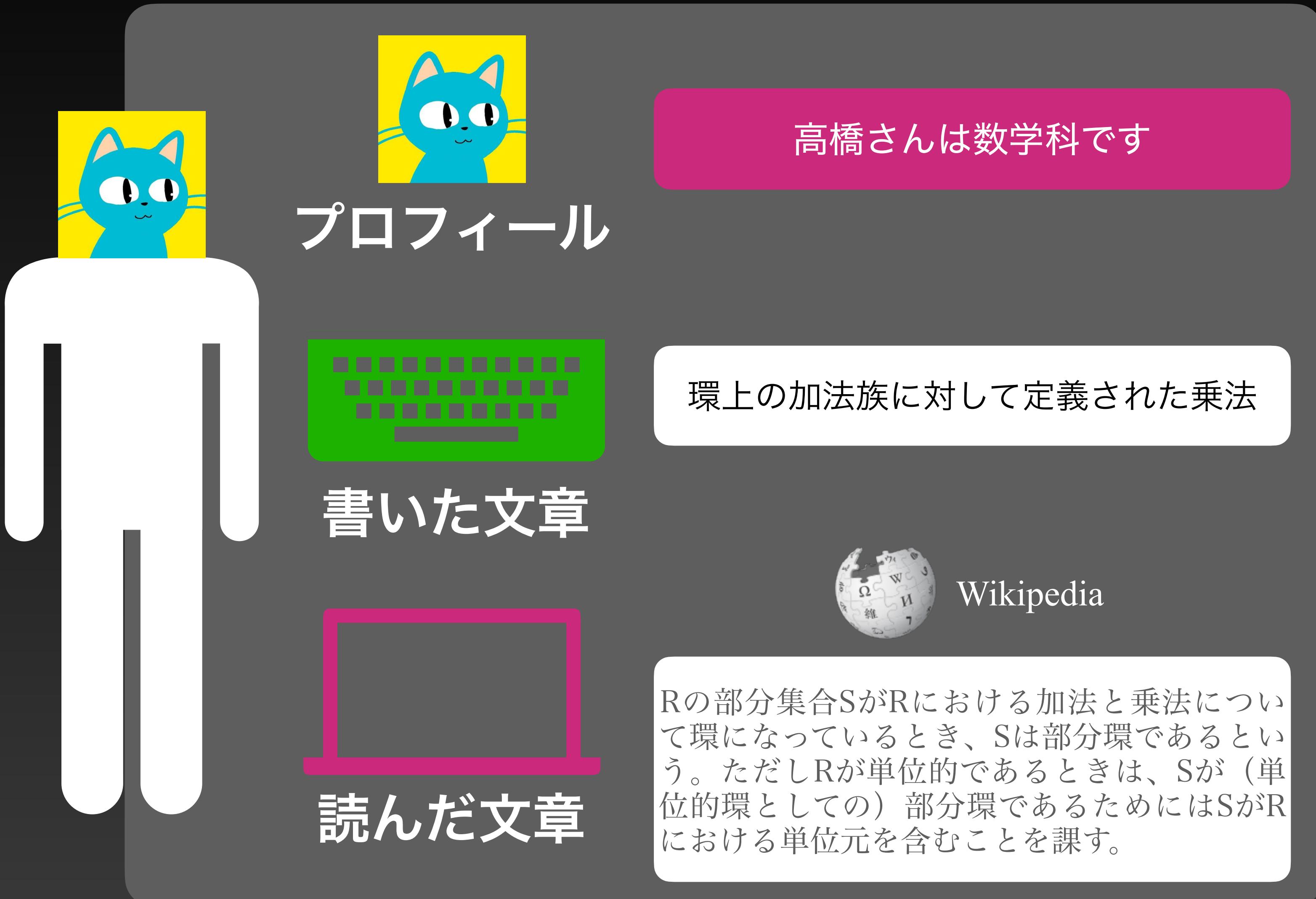
提案する日本語入力システム



ファインチューニングによる個人最適化



「文脈」のデータベースシステム



「文脈」のデータベースシステム

ユーザは「読んだ文章」を元に文章を書く

ここも必要

従来の「学習」対象

プロフィール



Wikipedia

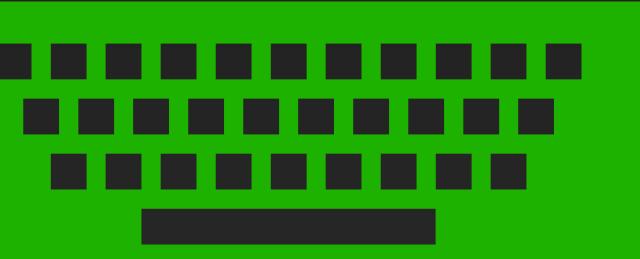
Rの部分集合SがRにおける加法と乗法について環になっているとき、Sは部分環であるという。ただしRが単位的であるときは、Sが（単位的環としての）部分環であるためにはSがRにおける単位元を含むことを課す。

読んだ文章

高橋さんは数学科です



環上の加法族に対して定義された乗法



書く文章

ニューラル言語モデルによる日本語入力

ファインチューニングで
柔軟に個人最適化

従来システムよりも
高度な処理を実現

予測入力を容易に実現

ニューラル言語モデルによる日本語入力

従来システムよりも
低速・高メモリ負荷

100Mパラメタ = 320MB
推論回数は毎秒100未満



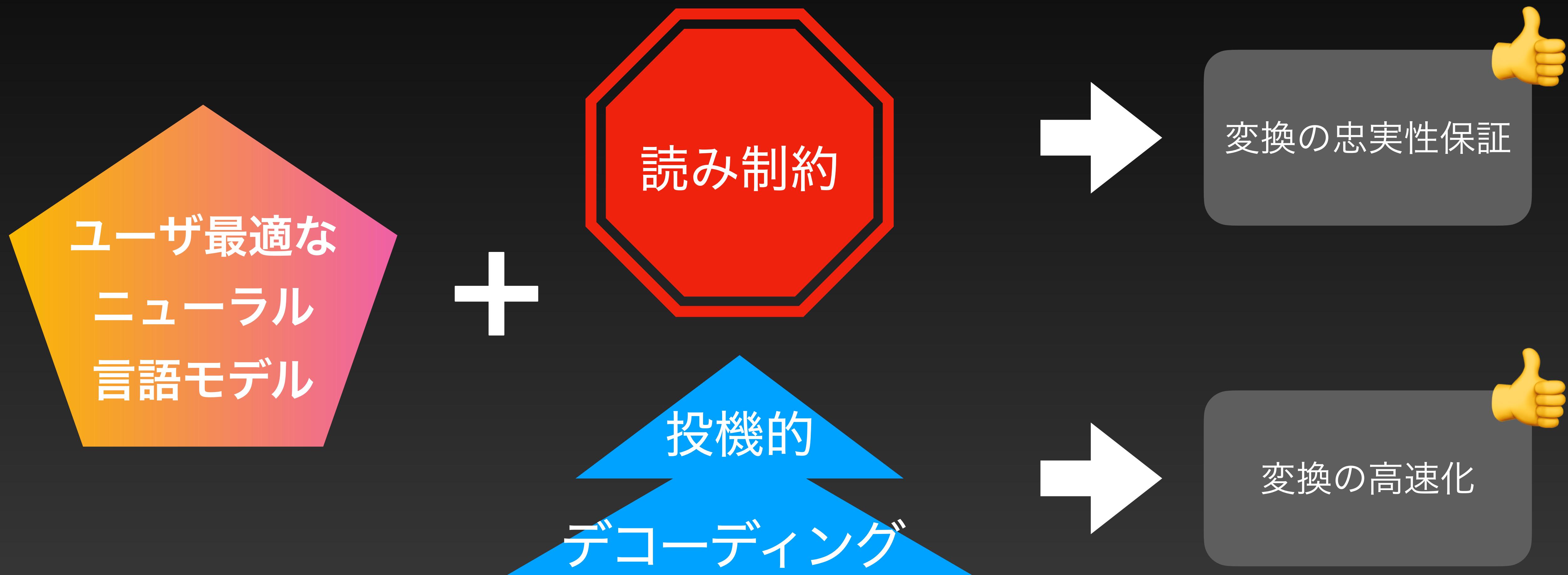
変換結果の忠実性の
保証が難しい

単語の読み情報を
明示的に扱えない



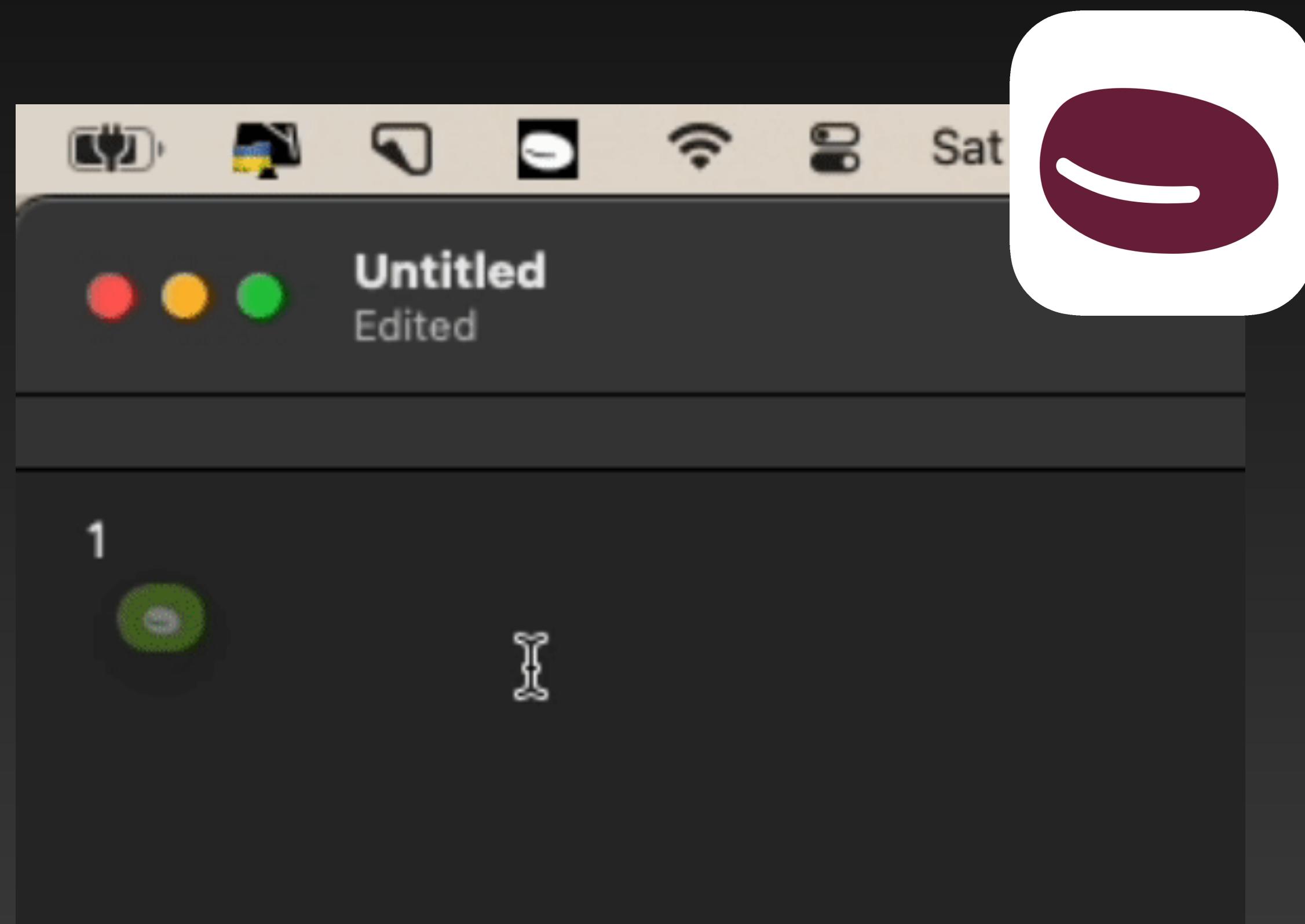
ニューラル言語モデルによる日本語入力

高速なニューラルかな漢字変換アルゴリズム



プロトタイプ

macOSの日本語入力システム (azooKey)
GPT-2 (1億パラメータ) を統合済み



macOSのアクセシビリティAPIを用い
文脈データベースプロトタイプを構築



プロトタイプ

- ・ 高橋のツイートでファインチューニングしたニューラル言語モデルを構築
- ・ 既存の日本語入力システムと統合し、個人最適化への有効性を検証済み

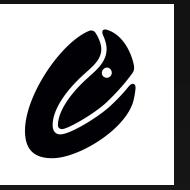
入力例	GPT-2無し	GPT-2 (Rinna)	GPT-2 (高橋化モデル)	目標候補	成功
すうがくのかん	数学の漢	数学の漢	数学の環	数学の環	YES
のはらしんのすけ	野原慎之助	野原慎之助	野原しんのすけ	野原しんのすけ	YES
のはらひろし	野原広	野原浩	野原浩	野原ひろし	NO

1 |

開発進行状況

提案プロジェクトは十分に実現可能

日本語入力システム



Miwa

✓ 量子化した100M規模

GPT-2を統合したIME

✓ スムーズな動作の確認

🚧 読み制約付き生成で
変換するシステムの実装

■ 投機的デコーディング

文脈データベース



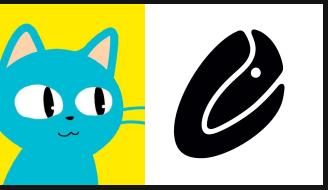
Takahashi

🚧 データベースの構築

✓ アクセシビリティAPIで
テキスト情報を取得

■ OCRなどで画面情報から
テキストを抽出

個人最適化



Both

✓ Twitterのログに基づく

ファインチューニングと
効果測定

🚧 効率的な個人最適化
手法の調査

■ 性能の改善・評価

提案する日本語入力システム



高度な個人最適化



高精度な変換



強い予測入力

日本語入力システム

ユーザ入力

変換結果

データを蓄積



ファインチューニング

ニューラル
かな漢字変換
アルゴリズム

ユーザ最適な
ニューラル
言語モデル

Appendix

既存技術との比較

変換システム	環境	ニューラル	書いたテキストを学習	読んだテキストを学習	そのほか	OSS	開発
Ours	macOS 他環境も目標	YES	YES	YES	ファインチューニングを行うプロフィールなども利用	YES	アクティブ
Kukura*	不明	/	YES	YES	Kukuraそのものは変換システムではないが、提案プロジェクト同様、予測入力への貢献を目指して構築されたデータベースシステムである	YES	終了済み
ATOK	Cross Platform	不明	YES	YES (ATOK Insight)	読んだテキストの取得方法は限定的	NO	アクティブ
Anthy	Cross Platform	NO	YES	NO		YES	終了済み
Wnn	組み込み	不明	YES	NO	位置情報なども利用	NO	アクティブ
Google日本語入力	Cross Platform	NO	YES	NO		YES	アクティブ
Microsoft IME	Windows	不明	YES	NO		NO	アクティブ
日本語入力システム (macOS)	macOS	YES	YES	NO		NO	アクティブ

開発タイムライン

期間	フェーズ	高橋の担当	三輪の担当
2024年 3月～5月	プロトタイプ 概念実証(完了)	ファインチューニングを行った言語モデルの準備	ニューラル言語モデルを用いたかな漢字変換アルゴリズムのプロトタイプ開発
6月～7月	フェーズ1 (基盤構築)	文脈データベースの基本システム構築、長期記憶システムの構築、macOSアプリ化	ニューラル言語モデルを用いたかな漢字変換アルゴリズムの実装、短期記憶システム構築
8月	フェーズ1の 統合・検証	フェーズ1で開発したシステムの統合、変換精度の検証と改善、方針確認	
9月～10月	フェーズ2 (機能開発)	OCRによる画面表示情報の取得システムの構築・検証	予測入力システムの実装、変換エンジンの効率化、ニューラル言語モデルの改善
11月	フェーズ2の 統合・検証	フェーズ2で開発した機能の統合、変換精度の検証と改善、方針確認	
12月～2025年1月	フェーズ3 (多様化と高機能化)	位置情報や音声情報、VRヘッドセットに基づく周辺情報などの取得システムの追加	さらなる高速化・高精度化、ユーザ辞書などの高機能化、検索拡張生成による予測入力の検討
12月～2025年1月	フェーズ3の 統合・検証	フェーズ3で開発した機能の統合、プロダクト全体の完成度向上	
2月	システムの仕上げ ドキュメント	GUI、インストーラ、アップデート機能などの整備、プロダクトとしての最終調整	
2月	未踏最終報告の準備		プロジェクトの総括

開発ステップ

フェーズ1

文脈データベースの基本システム構築

長期記憶システムの構築

macOSアプリ化

ニューラル言語モデルを用いたかな漢字変換アルゴリズムの実装

短期記憶システム構築

開発したシステムの統合・検証

フェーズ2

OCRによる画面表示情報の取得システムの構築・検証

予測入力システムの実装

変換エンジンの効率化

ニューラル言語モデルの改善

開発した機能の統合・検証

フェーズ3

位置情報や音声情報、VRヘッドセットに基づく周辺情報などの取得システムの追加

さらなる高速化・高精度化

ユーザ辞書などの高機能化

予測入力で検索拡張生成の利用

フェーズ3で開発した機能の統合

プロダクト全体の完成度向上

投機的デコーディング

- 低速なニューラル言語モデルによる生成の前に、高速で低精度なモデルを用いて「下書き」を作る
- ニューラル言語モデルの推論で下書きを検証し、妥当であればそのまま採用、妥当でなければニューラル言語モデルの出力で上書きする。下書きが成功した分だけ高速になる
- かな漢字変換は9割が高速な既存手法で十分なので、既存手法で下書きを生成し、ニューラル言語モデルで検証する投機的デコーディングを採用することで圧倒的な高速化を実現できる

ニューラル言語モデルの日本語入力への統合

