RAG-LLM Translator

RAG-based gettext PO file translator

DrupalCamp Kortrijk 2026

Mori Sugimoto

circular-orb.it

Hello! I’m Mori Sugimoto

Username on drupal.org: dokumori

On drupal.org: since 2006

Place of residence: Utrecht, the Netherlands

Place of origin: Tokyo, Japan

Freelance engineer, project/product manager

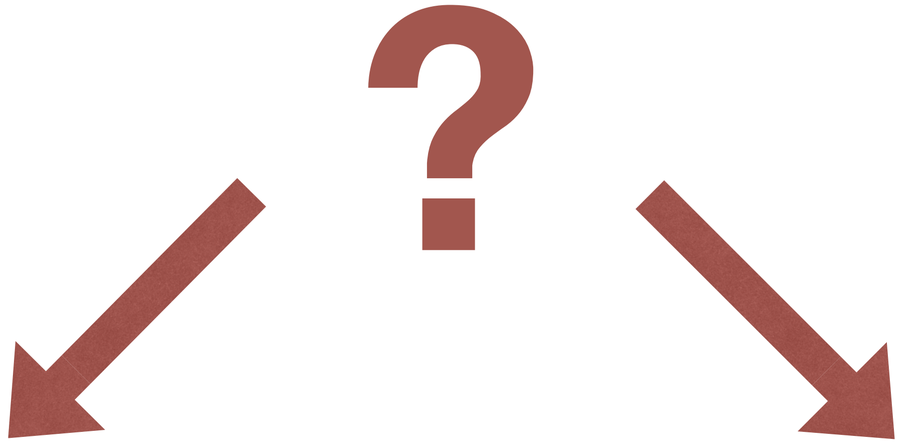
Email: mori@circular-orb.it

Why I built the RAG-LLM Translator

LLMs translate well, but lacks consistency

LLM only

‘Machine-readable name’

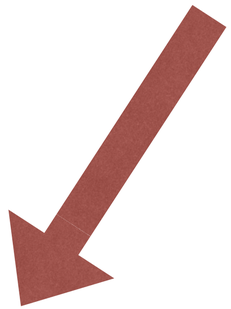


‘Systeem naam’

‘Machinale naam’

RAG + LLM

‘Machine-readable name’



‘Systeem naam’

~~‘Machinale naam’~~

RAG-LLM Translator

What it does:

It sends source strings in PO files to an LLM with RAG contexts

resulting in higher-quality translations

grounded in previously approved terminology and phrasing.

Terminology: RAG

RAG: Retrieval Augmented Generation

Searches a vector database of existing translations for the closest matching string, and feeds it into the LLM to produce more accurate, consistent output.

Architectural overview

RAG-LLM Translator

Infrastructure:	Docker Compose
LLM:	Local / major & custom external API endpoints
Gettext / file handling:	polib + custom code
Vector Engine:	ChromaDB
Text embedding model:	BAAI/bge-large-en-v1.5
Orchestration:	LiteLLM (LLM gateway) + RAG proxy (custom context injector)
Refinement:	Custom post-processing framework

Live demo (fingers crossed…)

Additional features

- Token counter / cost estimator
- Prompt override per language
- Glossary generator (from translation memory)
- RAG fine-tuning helper
- Translation evaluation (LLM-as-a-Judge)
- Post processing framework (accepts custom plugins)

Additional features

- **Token counter / cost estimator**
- Prompt override per language
- Glossary generator (from translation memory)
- RAG fine-tuning helper
- Translation evaluation (LLM-as-a-Judge)
- Post processing framework (accepts custom plugins)

Cost estimator (pre-execution)

```
📊 Cost Estimate — amazee.ai Claude 3.5 Haiku (1301 strings, ja)

   Strings to translate  : 1301

   Batches (~15 strings/batch) : 87

   Estimated cost (range): $0.4727 - $0.8666

   ⓘ  Actual cost will vary depending on the volume of RAG context

      injected per batch (size of glossary/TM).

----------------------------------------------------------------

Proceed with translation? [Y/n/q — q to quit]:
```

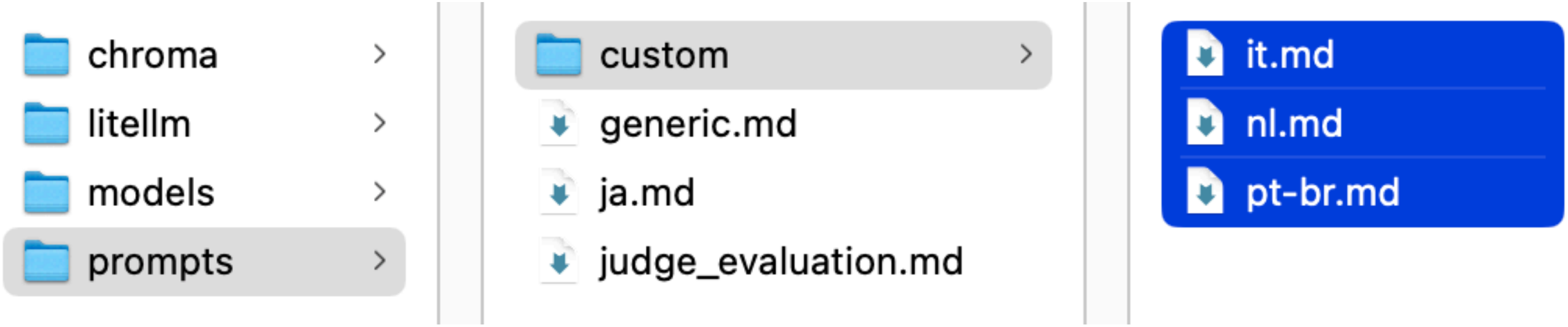
Cost estimator (post-execution)

```
═══════════════════════════════════════════════

💰 TOKEN USAGE SUMMARY

═══════════════════════════════════════════════

  Model             : amazee-ai-claude-haiku
  Requests          : 89
  Prompt tokens     : 109,386
  Completion tok.   : 24,110
  Total tokens      : 133,496
  Est. cost (USD)   : $0.2299

═══════════════════════════════════════════════
```

Additional features

- Token counter / cost estimator
- **Prompt override per language**
- Glossary generator (from translation memory)
- RAG fine-tuning helper
- Translation evaluation (LLM-as-a-Judge)
- Post processing framework (accepts custom plugins)

Prompt override per language



Additional features

- Token counter / cost estimator
- Prompt override per language
- **Glossary generator (from translation memory)**
- RAG fine-tuning helper
- Translation evaluation (LLM-as-a-Judge)
- Post processing framework (accepts custom plugins)

Glossary Generator

1. Import existing translations e.g.:
   - drupal-11.1.8.nl.po
   - pathauto-8.x-1.15.nl.po

2. Generate a list of glossary candidates

Bonus: Inconsistent translations are indicated

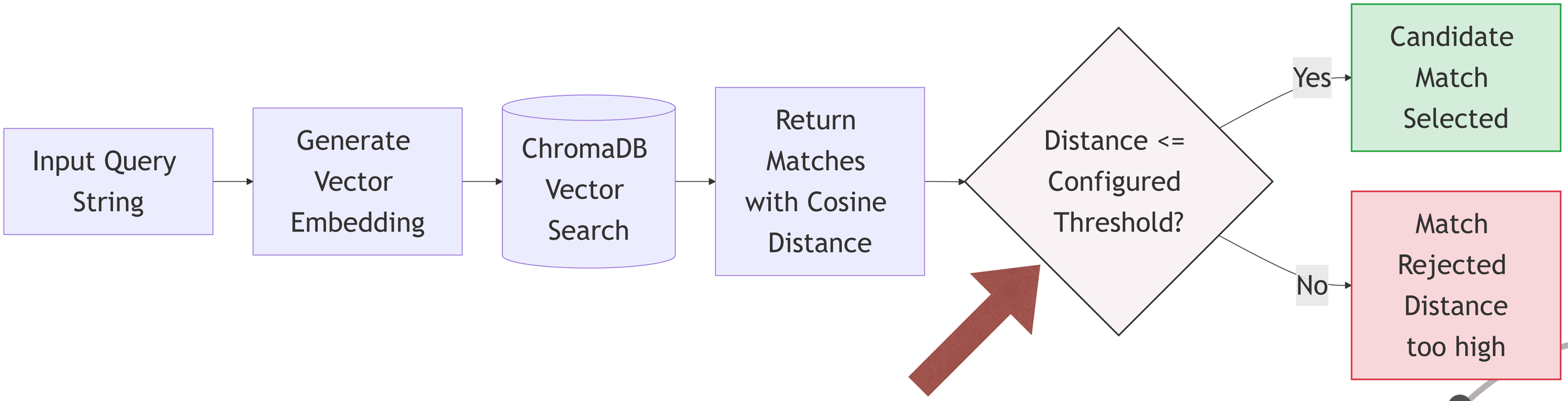
Glossary Generator: example

Source	Context	Target	Total Occurrences	Consistency	Alternatives
And		および	43	79.6%	AND (7); かつ (4)
Exposed		外部設置	34	79.1%	表示 (9)
Or		または	79	90.8%	OR (8)
Parent		親	31	79.5%	上位 (8)
Width		幅	23	67.6%	横幅 (11)

Additional features

- Token counter / cost estimator
- Prompt override per language
- Glossary generator (from translation memory)
- **RAG fine-tuning helper**
- Translation evaluation (LLM-as-a-Judge)
- Post processing framework (accepts custom plugins)

RAG fine-tuning helper



Fine-tuning = setting the distance threshold

RAG fine-tuning helper (excerpts)

```
--- 💡 Recommended Settings ---

Based on 1300 accepted matches:

- GLOSSARY_THRESHOLD:  0.37

- TM_THRESHOLD:        0.30
```

RAG fine-tuning helper (excerpts)

```
--- 🔍 Synonym Guardrail Analysis ---

  Distance Range                                Count  Status

  0.00-0.15 (within strict threshold)              24  ✅ ACCEPTED (below strict threshold)

  0.15-0.17 (borderline — review these)            13  ⚠️  REJECTED — potential synonyms?

 ~~~~~~~~~~~~~~

  Borderline Examples:

     e.g. 'Resolved' vs 'Fixed' (dist: 0.1623)

     e.g. 'field_breadcrumb_title' vs 'Title field' (dist: 0.1645)
```

Additional features

- Token counter / cost estimator
- Prompt override per language
- Glossary generator (from translation memory)
- RAG fine-tuning helper
- **Translation evaluation (LLM-as-a-Judge)**
- Post processing framework (accepts custom plugins)

LLM-as-a-Judge: Summary report

- 🏆 EVALUATION RESULTS SUMMARY
- Completed: 2026-04-12 15:35
- ==========
- JUDGE MODEL ⚖ : Kimi K2.5 (kimi-k2.5)
- Total Evaluated: **296**
- Wins (With RAG): **116**
- Wins (Without RAG): **34**
- Ties: **146**

- Files Compared:
-   - With RAG: en-ja_mistral-large-2402-v1_rag_2026-04-12_16-12-12.po (in with_rag/)
-   - Without RAG: en-ja_mistral-large-2402-v1_norag_2026-04-12_16-26-57.po (in without_rag/)

(continued)

```
----------------------------------------
Comparative Metrics (vs Non-RAG):
  - Win Ratio: 3.41x (RAG is 3.4x more likely to win)
  - Relative Win Rate: 77.3% (Preference in decided cases)
  - Win Lead: +241.2% (More 'Best' translations produced)
  - Contextual Error Reduction: 52.5% (Closing the gap to perfection)
  - Sub-optimal Rate Reduction: +63.2% (Reduction in scores < 4.0)
  - Net Improvement (Delta): +27.7% (Total win-rate difference)
  - Score Improvement: +13.3% (Average context score boost)
----------------------------------------
Average Context Adherence Score (Max 5):
  - With RAG: 4.52
  - Without RAG: 3.99
Average Accuracy & Fluency Score (Max 5):
  - With RAG: 4.46
  - Without RAG: 4.28
```

Additional features

- Token counter / cost estimator
- Prompt override per language
- Glossary generator (from translation memory)
- RAG fine-tuning helper
- Translation evaluation (LLM-as-a-Judge)
- **Post processing framework (accepts custom plugins)**

Post-processing Framework

LLMs may disregard granular instructions.

But the post-processing plugins allow you to apply the format you desire.

For example:

古い URL から新しい URL にリダイレクトします。

Whitespaces

Add your own post-processing plugins

For example, a **quotation mark localisation plugin**:

- **French / Spanish:** "text" ➔ « text »

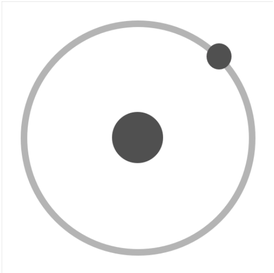
Or a **number/currency formatting plugin**:

- **German / Spanish:** $1,234.56 ➔ 1.234,56 $

…and some more features :)

- Multi-LLM provider gateway (Local / external)
- Custom embedding model configuration
- ChromaDB backup & restore (model mismatch guardrail)
- ChromaDB Admin UI

Thank you! / Dank je wel!



Mori Sugimoto

circular-orb.it