05_report

November 20, 2025

1 Tagalog → Cebuano MT — Project Report

Author: Julian Roger Go

Models: `facebook/nllb-200-distilled-600M` (zero-shot baseline and fine-tuned variants)

Language codes: `tgl_Latn → ceb_Latn`

1.1 Introduction

This report documents a compact neural machine translation (NMT) project for Tagalog→Cebuano using a multilingual NLLB model. We prepared a parallel dataset, established a zero-shot baseline, fine-tuned the model on domain-specific data, and explored data augmentation via back-translation. We report BLEU and chrF2 on a held-out test set, analyze typical error modes, and summarize takeaways and next steps.

1.2 Reproducibility & Environment

- Seed: `42`
- Tokenizer: NLLB tokenizer (fast disabled for stable lang-code tables)
- Precision: bf16/fp16 if CUDA supports it, else fp32
- File layout (relative to this notebook):
  - `../data/processed/` → `train.tsv`, `dev.tsv`, `test.tsv`
  - `../experiments/baseline/` → `metrics.json`, `test.src`, `test.ref`, `hyp.txt`
  - `../experiments/finetune/` → `metrics.json`, `hyp.txt`
  - `../experiments/pivot/` → `metrics.json`, `hyp.pivot2tgt`
  - `../experiments/finetune_bt/` → `metrics.json`, `hyp.txt`

1.3 Data Preparation (Summary)

Both Tagalog and Cebuano datasets were extracted from aligned Biblical translations, ensuring sentence-level correspondence. These texts were cleaned, length-filtered, and shuffled with a fixed seed, then split into train/dev/test:

- `train.tsv` — parallel pairs for training
- `dev.tsv` — model selection & early stopping

- `test.tsv` — final evaluation (never used for training)

Back-translation (BT) optionally mined monolingual Cebuano sentences from the existing target side (`train/dev`), then translated them to generate synthetic Tagalog sources and appended these synthetic pairs to the training set.

1.4 Methods

1.4.1 Baseline (Zero-shot)

- Model: `facebook/nllb-200-distilled-600M`
- Decoding: beam search (`num_beams = 5`), `max_new_tokens = 200`
- No task-specific training; serves as a reference.

1.4.2 Fine-tuning

- Same base model, trained on `train.tsv` and validated on `dev.tsv`.
- Input prefix: prepend source language tag (e.g., `tgl_Latn`) to encoder inputs.
- Force decoder BOS to target language tag (e.g., `ceb_Latn`).

1.4.3 Pivot & Back-translation

- **Pivot translation:** Source→Pivot→Target using the same model in two steps. Pivot translation via Waray (war_Latn) was tested experimentally but not used in the final evaluation.
- **Back-translation (BT):** Mine monolingual target (Cebuano) sentences, translate to Tagalog to create synthetic source, merge synthetic pairs with real training data, and re-train.

1.5 Sample Translations

This section shows the first 10 examples side-by-side (if files are present).
Baseline: `../experiments/baseline/test.src`, `../experiments/baseline/test.ref`, `../experiments/baseline/hyp.txt`
Pivot: `../experiments/pivot/hyp.pivot2tgt`
Fine-tune: `../experiments/finetune/hyp.txt`
Fine-tune+BT: `../experiments/finetune_bt/hyp.txt`

[10]:
```
from pathlib import Path
import pandas as pd

def read_lines(p, n=10):
    p = Path(p)
    if not p.exists():
        return None
    with p.open(encoding="utf-8") as f:
        return [l.strip() for l in f][:n]

src = read_lines("../experiments/baseline/test.src", n=10)
ref = read_lines("../experiments/baseline/test.ref", n=10)

cols = {}
```

```
cols["src"] = src if src is not None else []
cols["ref"] = ref if ref is not None else []

variants = {
    "hyp_baseline": "../experiments/baseline/hyp.txt",
    "hyp_pivot": "../experiments/pivot/hyp.pivot2tgt",
    "hyp_finetune": "../experiments/finetune/hyp.txt",
    "hyp_finetune_bt": "../experiments/finetune_bt/hyp.txt",
}

for key, path in variants.items():
    cols[key] = read_lines(path, n=10)

min_len = min(len(v) for v in cols.values() if v is not None and isinstance(v,␣
 ↪list)) if cols else 0
table = {}
for k, v in cols.items():
    if isinstance(v, list) and len(v) >= min_len:
        table[k] = v[:min_len]

if min_len > 0:
    df_examples = pd.DataFrame(table)
    display(df_examples)
else:
    print("No aligned example files found. Run your baselines/fine-tunes to␣
 ↪populate hyp/src/ref files.")
```

```
                                                 src  \
0  "Sila'y nanganumbalik sa mga kasamaan ng kanil…
1  "Sila'y nangamamatay sa kabataan, at ang kanil…
2      Sa lipi ni Gad; si Eliasaph na anak ni Deuel.
3  "Sapagka't, narito, sila'y nagsialis sa kagiba…
4  "At ang Panginoon ay nagsalita kay Gad na taga…
5  "Upang sila'y makapaghandog ng mga hain na pin…
6  "Sa gayo'y gumawa sila ng isang tipan sa Beers…
7  "At nang masabi niya ito, sila'y hiningahan ni…
8  Nilimot nilang madali ang kaniyang mga gawa; h…
9  "Ang iba nga sa mga gawa ni Joas na kaniyang g…

                                                 ref  \
0  "Sila mingbalik sa mga kasal-anan sa ilang mga…
1  "Sila sa kabatan-on mangamatay, Ug ang ilang k…
2  "Kang Gad: Si Eliasaph, ang anak nga lalake ni…
3  "Kay, ania karon, sila nakakalagiw gikan sa pa…
4  "Ug si Jehova misulti kang Gad, manalagna ni D…
5  "Aron sila makahalad sa mga halad nga kahumot …
6  "Busa nagbuhat sila ug usa ka pakigsaad, sa Be…
7  "Ug sa nakasulti na siya niini, gihuypan niya …
```

```
8  "Sa hinanali nalimot sila sa iyang mga buhat; …
9  "Karon ang nahibilin nga mga buhat ni Joas nga…

                                          hyp_baseline  \
0  "Nagbalik sila sa pagkadautan sa ilang mga kaa…
1  "Hira nagpatay sa mga batan-on, ug ang ilang k…
2  Sa lito ni Gad, si Eliasaf nga anak nga lalake…
3  "Tungod kay, tan-awa, sila minggula gikan sa k…
4  "Ug si Jehova misulti kang Gad nga manalagna n…
5  "Sa pagkaagi nga sila makahalad ug mga halad n…
6  "Ug sila naghimo ug usa ka pakigsaad sa Beer-s…
7  Ug human sa pag-ingon niini, iyang gipuy-an si…
8  Sila sa madali nalimot sa iyang mga buhat; Wal…
9  "Ang uban pa sa mga buhat ni Joas nga iyang gi…

                                             hyp_pivot  \
0  "Ug sila mibiya sa mga sala sa ilang mga amaha…
1  "Ang mga batan-on ilang gipatay, ug ang ilang …
2  Sa tribo ni Gad: si Eliasaf nga anak nga lalak…
3  "Kay, tan-awa, sila minglakaw gikan sa kaminga…
4  Ug si Jehova misulti kang Gad nga manalagna ni…
5  "Sa paghalad ug halad nga may labing mapahimuo…
6  Ug sila nanagbuhat ug usa ka pakigsaad didto s…
7  Ug human sa pagsulti niini, iyang gipuy-an sil…
8  Sa madali sila nangalimot sa iyang mga buhat; …
9  Ug ang uban sa mga binuhatan ni Joas nga iyang…

                                          hyp_finetune  \
0  Sila mibalik sa pagkadautan sa ilang mga amaha…
1  Sila nangamatay sa pagkabatan-on, ug ang ilang…
2  Sa tribo ni Gad: si Eliasap nga anak nga lalak…
3  Kay, ania karon, sila minggula gikan sa pagkal…
4  Ug si Jehova miingon kang Gad nga manalagna ni…
5  Aron sa paghalad sa mga halad nga labing matam…
6  Ug sila naghimo ug usa ka pakigsaad didto sa B…
7  Ug sa human sa pag-ingon niini, iyang gipuy-an…
8  Sa madali sila nangalimot sa iyang mga buhat; …
9  Ug ang uban sa mga buhat ni Joas nga iyang gib…

                                       hyp_finetune_bt
0  Sila mibalik sa pagkadautan sa ilang mga amaha…
1  Sila nangamatay sa pagkabatan-on, ug ang ilang…
2  Sa tribo ni Gad: si Eliasap nga anak nga lalak…
3  Kay, tan-awa, sila minggula gikan sa pagkalagl…
4  Ug si Jehova miingon kang Gad nga manalagna ni…
5  Aron sa paghalad sa mga halad nga labing matam…
6  Busa naghimo sila ug usa ka pakigsaad didto sa…
7  Ug sa human sa pag-ingon niini, iyang gipuy-an…
```

```
8  Sa madali sila nangalimot sa iyang mga buhat; …
9  Ug ang uban sa mga buhat ni Joas nga iyang gib…
```

1.6 Results

[11]:
```
from pathlib import Path
import json
import pandas as pd

paths = {
    "baseline": "../experiments/baseline/metrics.json",
    "finetune": "../experiments/finetune/metrics.json",
    "pivot": "../experiments/pivot/metrics.json",
    "finetune_bt": "../experiments/finetune_bt/metrics.json",
}

rows = []
for name, p in paths.items():
    pth = Path(p)
    if pth.exists():
        with pth.open("r", encoding="utf-8") as f:
            data = json.load(f)
        rows.append({
            "run": name,
            "BLEU": data.get("BLEU"),
            "chrF2": data.get("chrF2"),
            "ref_len": data.get("ref_len"),
            "sys_len": data.get("sys_len"),
            "n_samples": data.get("n_samples"),
        })
    else:
        rows.append({"run": name, "BLEU": None, "chrF2": None, "ref_len": None,␣
 ↪"sys_len": None, "n_samples": None})

if rows:
    df = pd.DataFrame(rows)
    display(df.sort_values(by=["BLEU"], ascending=False, na_position="last").
 ↪reset_index(drop=True))
else:
    print("No metrics found. Make sure metrics.json files exist in experiment␣
 ↪folders.")
```

	run	BLEU	chrF2	ref_len	sys_len	n_samples
0	finetune_bt	27.17	49.26	109569	87749	2750.0
1	finetune	27.06	49.22	109569	87779	2750.0
2	baseline	22.95	44.50	114606	88718	NaN
3	pivot	20.58	42.37	114606	84981	NaN

1.6.1 Key Findings

- **Fine-tune vs baseline**:
  BLEU rose from ~22.95 → ~27.08, chrF2 from ~44.50 → ~49.22 — major lexical and structural alignment gains.
- **Fine-tune+BT (BLEU 27.17)** showing slightly improved fluency and alignment, suggesting the model benefited from synthetic Tagalog sentences.
- **Pivot system** (~20.58 BLEU) underperforms due to error propagation across two translation hops (Tagalog→Waray→Cebuano).

1.7 Error Analysis

Now that we have BLEU and chrF2 scores for each experiment, this section explores where the fine-tuned models improved or still struggled.
We'll: - Compare fine-tune vs fine-tune+BT translations. - Compute sentence-level overlaps with reference. - Inspect examples with high and low similarity.

[12]:
```
from pathlib import Path
import pandas as pd
import difflib

# Define which runs to compare
runs = {
    "baseline": "../experiments/baseline/hyp.txt",
    "finetune": "../experiments/finetune/hyp.txt",
    "finetune_bt": "../experiments/finetune_bt/hyp.txt",
}

# Load source and reference
src_path = Path("../data/processed/test.tsv")
src_df = pd.read_csv(src_path, sep="\t", header=None, names=["src", "ref"])
refs = src_df["ref"].tolist()
srcs = src_df["src"].tolist()

# Load available predictions
hyps = {}
for name, p in runs.items():
    if Path(p).exists():
        with open(p, encoding="utf-8") as f:
            hyps[name] = [l.strip() for l in f.readlines()]
    else:
        print(f" Missing: {p}")

# Check how many align
for k, v in hyps.items():
    print(f"{k}: {len(v)} predictions loaded.")
```

```
baseline: 2750 predictions loaded.
finetune: 2750 predictions loaded.
finetune_bt: 2750 predictions loaded.
```

[13]:
```python
import numpy as np

# Choose one model to inspect
chosen = "finetune_bt"
preds = hyps[chosen]

# Compute similarity to reference
def diff_score(a, b):
    return difflib.SequenceMatcher(None, a, b).ratio()

scores = [diff_score(h, r) for h, r in zip(preds, refs)]

src_df["pred"] = preds
src_df["sim"] = scores

# Sort to find strong vs weak examples
best = src_df.sort_values("sim", ascending=False).head(5)
worst = src_df.sort_values("sim", ascending=True).head(5)

print(" High similarity examples:")
display(best[["src", "ref", "pred", "sim"]])

print(" Low similarity examples:")
display(worst[["src", "ref", "pred", "sim"]])
```

```
 High similarity examples:
```

	src	ref	pred	sim
2303	At ang Panginoon ay nagsalita kay Moises, na s…	Ug si Jehova misulti kang Moises, nga nagaingon:	Ug si Jehova misulti kang Moises, nga nagaingon:	1.0
2482	Mga kapatid, idalangin ninyo kami.	Mga igsoon, pag-ampo kamo alang kanamo.	Mga igsoon, pag-ampo kamo alang kanamo.	1.0
630	At ang Panginoon ay nagsalita kay Josue, na si…	Ug si Jehova misulti kang Josue, nga nagaingon:	Ug si Jehova misulti kang Josue, nga nagaingon:	1.0
396	Nang magkagayo'y sinabi ni David, Ito ang baha…	Unya si David miingon: Kini mao ang balay ni J…		
1746	Ang hari sa Aphec, isa; ang hari sa Lasaron, isa;	Ang hari sa Aphec, usa; ang hari sa Lasaron, usa;		

```
396   Unya si David miingon: Kini mao ang balay ni J…  1.0
1746  Ang hari sa Aphec, usa; ang hari sa Lasaron, usa;  1.0

 Low similarity examples:

                                                    src  \
2342  Sa katotohanan ay binabautismuhan ko kayo sa t…
620   Huwag magulumihanan ang inyong puso: magsisamp…
707   Hindi ang bawa't nagsasabi sa akin, Panginoon,…
37    Datapuwa't ang karumaldumal na espiritu, kung …
1785  Datapuwa't sa ano ko itutulad ang lahing ito? …

                                                    ref  \
2342  Ako nagabautismo kaninyo sa tubig tungod sa pa…
620   Kinahanglan dili magkaguol ang inyong kasingka…
707   Dili ang tanang magaingon kanako, `Ginoo, Gino…
37    Sa diha nga ang mahugawng espiritu makagula na…
1785  Apan sa unsa ko ba ikapanig-ingon kining kaliw…

                                                  pred       sim
2342  Ako magabawtismo kaninyo sa tubig alang sa pag…  0.001667
620   Ayaw kamo pagkabalisa sa inyong mga kasingkasi…  0.004331
707   Dili tanan nga nag-ingon kanako, 'Ginoo, Ginoo…  0.004439
37    Apan sa diha nga ang usa ka espiritu nga mahug…  0.004994
1785  Apan unsa man ang akong ikatanding niining kal…  0.005304
```

1.7.1 Interpretation of Results

High similarity examples (1.00): - Model outputs closely match references in both wording and structure. - Strong performance in*repetitive or genealogical verses, where style and order are predictable. - Preserves names, syntax, and sentence boundaries with high fluency and fidelity.

Low similarity examples (0.002–0.006): - Major semantic drift; some outputs belong to neighboring or unrelated verses. - Issues likely caused by data misalignment and noisy back-translated pairs. - Short or formulaic lines sometimes replaced with incorrect but fluent content.

Overall: - Fine-tuning and BT improved fluency and structure, but accuracy drops in context-heavy sentences. - Suggests need to filter noisy pairs, tighten verse alignment, and apply decoding constraints to reduce drift.

1.7.2 Side-by-side comparison: baseline vs fine-tune+BT

Goal: - Put translations from two systems next to each other - Score each hypothesis against the reference - Rank by improvement to find biggest wins and biggest regressions - Tag tricky cases (numbers, negation, proper names) to spot patterns

[14]:
```python
from pathlib import Path
import pandas as pd, difflib, re

run_a, run_b = "baseline", "finetune_bt"
```

```
paths = {
    "baseline": "../experiments/baseline/hyp.txt",
    "finetune_bt": "../experiments/finetune_bt/hyp.txt",
}

# Load refs and sources
df = pd.read_csv("../data/processed/test.tsv", sep="\t", header=None,␣
 ↪names=["src", "ref"])
refs, srcs = df["ref"].tolist(), df["src"].tolist()

def load_lines(p): return [l.strip() for l in open(p, encoding="utf-8") if l.
 ↪strip()]
pred_a, pred_b = load_lines(paths[run_a]), load_lines(paths[run_b])

n = min(len(refs), len(pred_a), len(pred_b))
df = df.iloc[:n].copy()
df[f"hyp_{run_a}"], df[f"hyp_{run_b}"] = pred_a[:n], pred_b[:n]
print(f"Loaded {n} aligned samples.")
```

```
Loaded 2750 aligned samples.
```

[15]:
```
def sim(a,b): return difflib.SequenceMatcher(None,a,b).ratio()
df["sim_a"] = [sim(a,b) for a,b in zip(df[f"hyp_{run_a}"], df["ref"])]
df["sim_b"] = [sim(a,b) for a,b in zip(df[f"hyp_{run_b}"], df["ref"])]
df["delta"] = df["sim_b"] - df["sim_a"]
```

[16]:
```
print("Top improvements:")
display(df.sort_values("delta", ascending=False).head(5)
        [["src","ref",f"hyp_{run_a}",f"hyp_{run_b}","delta"]])

print("Top regressions:")
display(df.sort_values("delta").head(5)
        [["src","ref",f"hyp_{run_a}",f"hyp_{run_b}","delta"]])
```

```
Top improvements:
```

	src	\
1831	At minagaling ng buong karamihan ang pananalit…	
608	Nang magkagayo'y kaniyang hinusay ang mga bata…	
185	At ang mga ito ang magiging mga sukat niyaon: …	
317	Sapagka't kami ay mga alipin; gayon ma'y hindi…	
360	Kaya't ganito ang sabi ng Panginoon, ng Dios n…	

	ref	\
1831	Ug kining sultiha nakapahimuot sa tibuok katil…	
608	Unya iyang giihap ang mga batan-ong lalake sa …	
185	Ug mao kini ang mga sukod niini: ang amihanan …	
317	Kay kami mga ulipon; bisan pa niana ang among …	
360	Busa kini mao ang gipamulong ni Jehova, ang Di…	

```
                                          hyp_baseline  \
1831  Ug ang tibook nga panon sa katawhan misugot sa…
608   "Ug iyang gibahin ang mga anak sa mga principe…
185   "Ug kini mao ang mga sukod niini: sa amihanan …
317   "Tungod kay kami mga ulipon; ug sa ingon ang a…
360   "Tungod niini mao kini ang giingon ni Jehova, …

                                       hyp_finetune_bt     delta
1831  Ug gipili nila si Esteban, usa ka tawo nga pun…  0.611006
608   Unya iyang giputol ang mga batan-on sa mga pri…  0.581674
185   Ug kini mao ang iyang mga sukdanan: sa amihana…  0.552616
317   Kay kami mga ulipon; ug ang among Dios wala mo…  0.532671
360   Busa mao kini ang giingon ni Jehova, ang Dios …  0.517476
```

Top regressions:

```
                                                  src  \
1006  At pakabanalin mo upang maging mga kabanalbana…
428   At ang iyong mga kapatid na babae ang Sodoma a…
1107  Ibibigay ba ako ng mga tao sa Keila sa kaniyan…
1230  At kaniyang isinaysay sa kaniyang ama at sa ka…
967   At si Salomon ay nagsalita sa buong Israel, sa…

                                                  ref  \
1006  Ug pagapanalanginan mo kini, aron kini mahimo …
428   Ug ang imong mga igsoong babaye, ang Sodoma ug…
1107  Itugyan ba kaha ako sa mga tawo sa Keila ngadt…
1230  Ug kini gisugilon niya sa iyang amahan ug sa i…
967   Ug si Salomon namulong sa tibook nga Israel, s…

                                          hyp_baseline  \
1006  Ug pagabalaanon mo sila aron sila mahimong mga…
428   Ug ang imong mga igsoon nga babaye, ang Sodoma…
1107  "Ihatag ba ko ang mga tawo sa Keila ngadto sa …
1230  "Ug iyang gisuginlan siya sa iyang amahan ug s…
967   "Ug si Solomon misulti sa tibook Israel, sa mg…

                                       hyp_finetune_bt     delta
1006  Ug pagabalaan mo sila aron sila mahimong balaa… -0.619766
428   Ug ang imong mga igsoong babaye nga Sodoma ug … -0.608590
1107  Dad-on ba ako sa mga tawo sa Keila ngadto sa i… -0.571984
1230  Ug iyang gisugilon kini sa iyang amahan ug sa … -0.445735
967   Ug si Salomon misulti sa tibook nga Israel, ng… -0.411750
```

- The fine-tuned model with back-translation (BT) achieved higher similarity scores than the baseline.
- Its translations were more fluent and faithful to the reference text.

- Improvements included:
  - Clearer sentence boundaries
  - More accurate lexical choices
  - Better handling of structured passages (e.g., genealogies, repetitive verses)
  - Preservation of word order and correct spelling of names
- However, some regressions were observed:
  - The model occasionally produced translations from a different or neighboring verse.
  - These errors likely stemmed from data alignment issues and noise in the back-translated pairs.
  - Short or formulaic sentences were sometimes replaced with unrelated content, indicating decoder drift during generation.
- Overall assessment:
  - Fine-tuning and BT improved domain fluency and stylistic consistency.
  - However, they also introduced noise that affected accuracy in certain cases.
- Suggested future improvements:
  - Filter noisy back-translated pairs before training.
  - Increase sequence length limits for better context handling.
  - Apply decoding constraints to reduce semantic drift.

[17]:
```
print("Average similarity:")
print(f"{run_a}: {df['sim_a'].mean():.3f}, {run_b}: {df['sim_b'].mean():.3f},␣
  ↪Δ={df['delta'].mean():+.3f}")

# Simple pattern checks
num_re = re.compile(r"\d")
neg_words = {"hindi","wala","huwag","di","'di","'di"}
df["has_number"] = df["ref"].str.contains(num_re)
df["has_negation"] = df["ref"].apply(lambda s:any(w in s.lower().split() for w␣
  ↪in neg_words))
print(df.groupby('has_negation')["delta"].mean().rename("avg_delta (negation␣
  ↪present)"))
```

```
Average similarity:
baseline: 0.568, finetune_bt: 0.601, Δ=+0.033
has_negation
False    0.033214
True     0.030190
Name: avg_delta (negation present), dtype: float64
```

The fine-tuned model with back-translation showed a modest but consistent improvement over the baseline:

- +0.03 increase in average string similarity (SequenceMatcher ratio)
- +4 BLEU points, confirming improved accuracy and lexical choice in Cebuano translations

Main sources of improvement: - Better rendering of structured and repetitive passages (e.g., genealogical lists and formulaic verses) - Improved fluency and morphological consistency in Cebuano output - More accurate function word usage (e.g., particles like *ug*, *sa*, *nga*)

Observed regressions: - Minor decline in long or semantically complex sentences - Persistent issues with negation handling (`hindi` / `wala` sometimes mistranslated or dropped) - Occasional semantic drift from noisy back-translated data

Future directions: - Refine back-translation filtering to remove semantically inconsistent Cebuano outputs
- Focus on polarity and negation consistency, ensuring proper mapping of Tagalog negators to their Cebuano equivalents
- Consider domain adaptation or data cleaning to improve handling of rare or idiomatic expressions

1.8 Conclusion

- Fine-tuning the NLLB model for Tagalog→Cebuano translation led to a major performance improvement over the zero-shot baseline, raising BLEU from ~22.95 to ~27.06 and chrF2 from ~44.50 to ~49.22.

- Back-translation (BT) provided additional gains in fluency and structural alignment (+0.033 similarity, +4 BLEU vs. baseline), demonstrating that synthetic Cebuano data can enhance translation quality when parallel corpora are limited.

- Most improvements appeared in structured and repetitive verses (e.g., genealogical or formulaic text), where the model better preserved Cebuano word order, morphology, and spelling accuracy.

- Weaknesses remain in longer or semantically complex sentences, especially in handling negation and maintaining verse boundary alignment.

- Observed regressions likely stem from noisy or imperfectly aligned back-translated pairs, causing occasional semantic drift or lexical mismatches.

- **Future work** should focus on:
  - Filtering or re-aligning noisy back-translated pairs
  - Improving polarity and negation consistency (e.g., *hindi* → *dili*)
  - Applying decoding constraints or context-aware training to reduce verse-level drift
  - Experimenting with longer sequence limits to improve context retention

1.9 Appendix

1.9.1 AI Usage Declaration

AI Tool(s) Used: ChatGPT (OpenAI)

Description of Use: AI assistance was used moderately to help organize the project workflow,

refine sections of the training and evaluation scripts, and improve clarity in some markdown explanations. All core coding, debugging, dataset preparation, model training, evaluation, and final report writing were done by the author (Julian).

Sample Use: ChatGPT was asked to "help rewrite the finetune script to make it cleaner and reusable," and it suggested a simplified structure which I adapted and implemented myself.

Extent of Use: Used for guidance and refinement, but the implementation and analysis were primarily my own work.