report

November 12, 2025

1 Cebuano → Tagalog MT — Project Report

Author: Julian
Date: *(fill in)*
Models: `facebook/nllb-200-distilled-600M` (zero-shot baseline and fine-tuned variants)
Language codes: `ceb_Latn → tgl_Latn`

1.1 Introduction

This report documents a compact neural machine translation (NMT) project for Cebuano→Tagalog 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)

Source and target 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 Tagalog sentences from the existing target side (`train/dev`), then translated them to generate synthetic Cebuano 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., `ceb_Latn`) to encoder inputs.
- Force decoder BOS to target language tag (e.g., `tgl_Latn`).

1.4.3 Pivot & Back-translation

- **Pivot translation:** Source→Pivot→Target using the same model in two steps (optional; use a distinct pivot like `war_Latn` for Waray if desired).
- **Back-translation (BT):** Mine monolingual target (Tagalog) sentences, translate to Cebuano 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`

[2]:
```
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 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…

                                                 ref  \
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…

                                          hyp_baseline  \
0  "Nang bumalik sila sa mga kasamaan ng kanilang…
1  "Ang mga kabataan ay mamamatay, at ang kanilan…
2         At sa Gad: si Eliasap, ang anak ni Dehuel.
3  "Sapagka't, narito, sila'y tumakas mula sa pag…
4  "At sinabi ni Jehova kay Gad na propeta ni Dav…
5  "Sila ay maghahandog ng mga handog na insenso …
6  "At sila'y gumawa ng isang tipan, sa Beer-sheb…
7  At nang sabihin niya ang mga ito, ay inihuhuhu…
8  "Sapagka't sila'y agad na nakalimutan ang kani…
9  "At ang natitirang mga gawa ni Joas na kaniyan…

                                            hyp_pivot  \
0  At sila'y bumalik sa mga kasamaan ng kanilang …
1  "Ang mga kabataan ay mamamatay, at ang kanilan…
2            At sa Gad: si Eliasaf na anak ni Deuel.
3  "Sapagka't, narito, sila'y tumakas mula sa pag…
4   At sinabi ni Jehova kay Gad na propeta ni David,
5  "Maghahandog sila ng mga insenso sa Diyos ng l…
6  At sila'y gumawa ng tipan sa Beer-seba: at tum…
7  Nang magsabi siya ng mga bagay na ito, ay inih…
8  At sila'y agad na nakalimutan ang kaniyang mga…
9  At ang iba sa mga gawa ni Joas, na kaniyang gi…

                                         hyp_finetune  \
0  Nagbalik sila sa mga kasamaan ng kanilang mga …
1  Sila'y mamamatay sa pagkabata, at ang kanilang…
2             Sa Gad: si Eliasap, na anak ni Dehuel.
3  Sapagka't, narito, sila'y tumakas mula sa pagk…
4  At sinabi ni Jehova kay Gad na propeta ni Davi…
5  Upang maghandog sila ng mga handog na inumin s…
6  Kaya't sila'y gumawa ng isang tipan, sa Beer-s…
7  Nang sabihin niya ito, inihinga niya sa kanila…
8  Mabilis nilang nalimot ang kaniyang mga gawa; …
9  At ang natitirang mga gawa ni Joas na kaniyang…

                                      hyp_finetune_bt
0  Nagbalik sila sa mga kasamaan ng kanilang mga …
1  Sila'y mamamatay sa pagkabata, at ang kanilang…
2             Sa Gad: si Eliasap, na anak ni Dehuel.
3  Sapagka't, narito, sila'y tumakas mula sa pagk…
4  At sinabi ni Jehova kay Gad na propeta ni Davi…
5  Upang maghandog sila ng mga handog na inumin s…
6  Kaya't sila'y gumawa ng isang tipan, sa Beer-s…
7  Nang sabihin niya ito, inihinga niya sa kanila…
```

```
8  Mabilis nilang nalimot ang kaniyang mga gawa; …
9  At ang natitirang mga gawa ni Joas na kaniyang…
```

1.6 Results

[1]:
```
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	30.08	56.64	85119	89786	2750.0
1	finetune	29.83	56.32	85119	89467	2750.0
2	baseline	1.45	20.83	89797	95072	NaN
3	pivot	1.39	20.30	89797	91058	NaN

1.6.1 Key Findings

- **Fine-tune vs baseline**:
  BLEU rose from ~1.45 → ~29.8, chrF2 from ~20.8 → ~56.3 — major lexical and structural alignment gains.
- **Fine-tune+BT (BLEU 30.1)** slightly improved fluency and alignment with longer outputs (as seen in sys_len), suggesting the model benefited from synthetic Tagalog sentences.
- **Pivot system** (~1.39 BLEU) underperforms due to error propagation across two translation hops (Cebuano→Pivot→Tagalog).

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.

[5]:
```
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: 3093 predictions loaded.
finetune: 2750 predictions loaded.
finetune_bt: 2750 predictions loaded.
```

[6]:
```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
899	Ug pinaagi sa espada iyang gipatay si Santiago…	At pinatay niya sa tabak si Santiago na kapati…	At pinatay niya sa tabak si Santiago na kapati…	1.000000
80	Ug kamo mahimo nga akong katawohan, ug ako mah…	At kayo'y magiging aking bayan, at ako'y magig…	At kayo'y magiging aking bayan, at ako'y magig…	1.000000
1956	Kay ang akong unod tinuod nga kalan-on ug ang …	Sapagka't ang aking laman ay tunay na pagkain,…	Sapagka't ang aking laman ay tunay na pagkain …	0.994012
1870	Ang mga anak nga lalake ni Pares: si Hesron ug…	Ang mga anak ni Phares: si Hesron at si Hamul.		
1232	Ug nahinumdum sila sa iyang mga pulong,	At naalaala nila ang kaniyang mga salita,		

```
1870      Ang mga anak ni Pares: si Hesron at si Hamul.  0.989011
1232           At naalala nila ang kaniyang mga salita,  0.987654

 Low similarity examples:

                                                    src  \
1170  Tungod sa tinapay nga gibutang sa atubangan sa…
942   Ipamati karon ang imong igdulungog, ug bukha a…
707   Dili ang tanang magaingon kanako, `Ginoo, Gino…
620   Kinahanglan dili magkaguol ang inyong kasingka…
1367  si Juan mitubag kanilang tanan, "Kaninyo nagab…

                                                    ref  \
1170  Ukol sa tinapay na handog, at sa palaging hand…
942   Pakinggan ngayon ng iyong tainga, at idilat an…
707   At sinabi sa kaniya, Ang bawa't tao ay unang i…
620   Kung kayo nga, bagaman masasama, ay marurunong…
1367  Ay sumagot si Juan na sinasabi sa kanilang lah…

                                                   pred       sim
1170  Dahil sa tinapay na inilagay sa harap ng Dios,…  0.003008
942   Ngayon, pakinggan mo ang iyong tainga, at buks…  0.005141
707   Hindi lahat ng nagsasabing sa akin, Panginoon…  0.005958
620   Ang inyong puso ay huwag magsisisi. Magtiwala …  0.006054
1367  Sinabi ni Juan sa kanilang lahat, Ako'y nagbub…  0.007519
```

1.7.1 Interpretation of Results

High similarity examples (0.99–1.00): - 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 (0.003–0.007): - 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.

1.8 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

[13]:
```
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.
```

[14]:
```
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"]
```

[15]:
```
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	\
1185	Ang mga anak ni Uzza, ang mga anak ni Phasea, …	
428	Ug ang imong mga igsoong babaye, ang Sodoma ug…	
1870	Ang mga anak nga lalake ni Pares: si Hesron ug…	
1281	Ug si Maasias ang anak nga lalake ni Baruch, a…	
841	Ni managpatalinghug kami sa imong mga alagad n…	

	ref	\
1185	Ang mga anak ni Uzza, ang mga anak ni Phasea, …	
428	At ang iyong mga kapatid na babae ang Sodoma a…	
1870	Ang mga anak ni Phares: si Hesron at si Hamul.	

```
1281  At si Maasias na anak ni Baruch, na anak ni Co…
841   Na hindi man kami nangakinig sa iyong mga ling…

                                          hyp_baseline  \
1185  "Sapagka't narito, ako'y nagsisimula sa pagpap…
428   "Hatapos pa ang halagang tumbaga na ginawa ni …
1870  Sapagka't ngayon ay aking sinasaktan ang aking…
1281  "Walang isa sa inyo ang magsilapit sa sinumang…
841   Ang bayan na ito ay hindi magiging inyong pana…

                                       hyp_finetune_bt     delta
1185  Ang mga anak ni Uzza, ang mga anak ni Pasea, a…  0.923067
428   At ang iyong mga kapatid na babae, ang Sodoma …  0.850690
1870      Ang mga anak ni Pares: si Hesron at si Hamul.  0.838541
1281  At si Maasias ay anak ni Baruc, anak ni Colboz…  0.836601
841   At kami'y hindi makinig sa iyong mga lingkod n…  0.823293
```

Top regressions:

```
                                                  src  \
2342  Ako nagabautismo kaninyo sa tubig tungod sa pa…
875   Ug ang iyang anak nga magulang didto sa uma; u…
2373                       Ug si Job mitubag ug miingon:
1250  Diha kanimo gitamay nila ang amahan ug inahan;…
2467  Ug magabuhat ka ug usa ka binakbak nga lunsay …

                                                  ref  \
2342  Gayon man kung inyong ganapin ang kautusang ha…
875   Katotohanang sinasabi ko sa inyo, Ang sinomang…
2373  At sinabi ng Panginoon kay Satanas, Saan ka na…
1250  Sa iyo'y kanilang niwalang kabuluhan ang ama't…
2467  At gagawa ka ng isang laminang taganas na gint…

                                          hyp_baseline  \
2342  At nang agad ay nalaman ni Jesus sa kaniyang e…
875   "Sapagka't gaya ng gulay na lumilitaw mula sa …
2373  At iyong nakita ang kasamaan ng ating mga magu…
1250  at iligtas ang lahat ng tao na sa buong buhay …
2467  At ang lahat ng mga matanda sa Israel ay nagti…

                                       hyp_finetune_bt     delta
2342  Ako'y nagbubunyag sa inyo sa tubig dahil sa pa… -0.306447
875   At ang kaniyang magulang na anak ay nasa paran… -0.269860
2373                        At sumagot si Job at sinabi: -0.206497
1250  Sa iyo'y kanilang pinabayaan ang ama at ang in… -0.206033
2467  At ikaw ay maggagawa ng isang pinutol na puron… -0.201625
```

- 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.

[16]:
```
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: 0.229, finetune_bt: 0.588, Δ=+0.359
has_negation
False    0.371081
True     0.319358
Name: avg_delta (negation present), dtype: float64
```

- The fine-tuned model with back-translation showed a **large performance improvement** over the baseline:
  - **+0.36** increase in average similarity
  - **+29 BLEU points**, indicating more accurate and fluent translations

- **Main sources of improvement:**
  - Better handling of **structured and repetitive verses**
  - Improved **fluency** and **lexical accuracy**
- **Observed regressions:**
  - Some decline in **longer or semantically complex sentences**
  - Persistent weakness in **negation handling**
- **Future directions:**
  - Focus on improving **polarity consistency** (correctly translating negation)
  - **Filter noisy back-translated pairs** to reduce semantic drift and misalignment

1.8.1 Conclusion

- **Fine-tuning** the NLLB model led to a **major performance jump** over the zero-shot baseline, raising BLEU from ~1.45 to ~29.8 and chrF2 from ~20.8 to ~56.3.
- **Back-translation (BT)** further improved fluency and structural alignment (+0.36 similarity, +29 BLEU vs. baseline), showing that synthetic data can enhance translation quality when parallel data is limited.
- Most gains occurred in **structured and repetitive verses** (e.g., genealogies, formulaic text), where the model preserved word order and spelling accuracy.
- **Weaknesses remain** in long or semantically complex sentences, especially in handling **negation** and **verse boundary alignment**.
- **Observed regressions** likely stem from noisy or imperfectly aligned back-translated pairs, causing occasional semantic drift.
- Overall, fine-tuning with BT **enhanced fluency, lexical accuracy, and domain style**, but introduced some noise.
- **Future work** should focus on:
  - Filtering or re-aligning noisy BT pairs
  - Improving polarity and negation consistency
  - Applying decoding constraints or context-aware training to reduce verse-level drift
  - Experimenting with longer sequence limits for improved context retention