1 Worked Example: Bengali–English Translation with Revised TATN

We illustrate the revised TATN algorithm with a concrete Bengali→English example. Consider the sentence:

> **Input (Bengali):** আমি কল বন্ধ করলাম।
> **Ambiguous token:** কল (can mean *tap/faucet* or *phone call*)

The algorithm processes this token in two distinct phases: *Phase A (Training)* and *Phase B (Inference)*.

1.1 Phase A: Training Example

1. **Encoding.** The contextual embedding for token "কল" is

$$h_j = [0.6,\, 0.8,\, 0.0], \quad \|h_j\| = 1.$$

2. **Prototypes.** Two prototypes for type "কল" already exist:

$$c_{w,1} = [0.8,\, 0.6,\, 0], \quad c_{w,2} = [-0.6,\, 0.8,\, 0].$$

3. **Cosine Similarity.**

$$s_{j,1} = \cos(h_j, c_{w,1}) = 0.96, \quad s_{j,2} = 0.28.$$

4. **Sense Probabilities (softmax, $T = 0.8$).**

$$p_j = [0.7006,\, 0.2994], \quad \hat{y}_j = \text{tap}.$$

5. **Uncertainty Signals.**

$$\begin{aligned} H_j &= -\sum_i p_{j,i} \ln p_{j,i} = 0.610, \\ \text{Var}_j &= 0.00013 \quad \text{(MC-dropout variance)}, \\ \sigma_j &= \exp(0.5 u_j) = 1.073, \\ d_{\text{min}} &= \min_i (1 - s_{j,i}) = 0.04. \end{aligned}$$

   Aggregated:

$$U_j = H_j + \text{Var}_j + \sigma_j + d_{\text{min}} \approx 1.72.$$

6. **Flagging & Gating.** Since $U_j > \delta_{\text{inst}} = 0.5$, token j is flagged. Gate value:

$$g_j = \sigma(U_j - b_g) = \sigma(1.72 - 0.8) = 0.715.$$

7. **Attention Boost.** If $a_j^{(0)} = 0.05$ (base attention), then

$$\tilde{a}_j = a_j^{(0)} \cdot (1 + \gamma g_j), \quad \gamma = 2.0 \;\Rightarrow\; \tilde{a}_j = 0.1215.$$

8. **Sense-Augmented Embedding.**

$$h'_j = h_j + c_{w,1} = [1.4,\ 1.4,\ 0].$$

9. **ASBN (Adversarial).** Gradient reversal strengths:

$$\lambda_{k,j} = \text{clip}\big(\bar{\lambda}_k \cdot p_{\text{max}}(1 - U_j) g_j,\ 0,\ \lambda_{\text{max}}\big).$$

   Since $(1 - U_j) < 0$, all $\lambda_{k,j}$ clip to 0: no adversarial forcing is applied here.

10. **Rationale Extraction (TRG).** Evidence collected:

    - Prototype match: $c_{w,1}$ (tap, distance 0.04).
    - Context cue: verb "বন্ধ" (close/turn off).
    - Alternatives: tap (0.70), call (0.30).

    Template rationale R^T: *"Because the verb 'বন্ধ' (turn off) appears in context and the nearest prototype corresponds to a faucet, translate 'কল' as **tap**."*

During training, R^T is stored in a silver dataset for the generator–verifier (TRG) module.

1.2 Phase B: Inference Example

At inference, the system repeats the forward pass but does not update parameters:

1. Compute $p_j = [0.7006, 0.2994]$, $U_j = 1.72$.
2. Since $p_{\text{max}} = 0.70 < 0.9$ and $U_j > 0.2$, a generated rationale is required.
3. Generator G outputs candidates, e.g.:
   - R1: "Because the verb 'বন্ধ' appears in context and the prototype matches faucet, translate as tap."
   - R2: "Because it looks like a phone event, translate as call."
4. Verifier V accepts R1 (faithful to sense="tap"), rejects R2.
5. Final translation:

   Output: "I turned off the tap."

   with rationale R1.

1.3 Summary

This worked example shows how:

- **DSCD** computes sense distributions, uncertainty, and prototypes.
- **ASBN** selectively suppresses shortcut cues via adaptive GRL.
- **TRG** generates and verifies rationales, ensuring explanations are faithful.

Thus, the ambiguous token "কল" is correctly interpreted as *tap* in this context, with a faithful natural language rationale provided.