Tri-Agent Reinforcement Learning for Low-Resource English→Arabic Translation

Abstract

We propose a tri-agent reinforcement learning framework for translation under low-resource conditions.
The system consists of a **Teacher (B)** that generates English tasks, a **Student (A)** that learns to produce Arabic translations, and a **Judge (C)** that provides evaluative rewards.
Unlike traditional supervised translation models that plateau with limited parallel data, our approach enables self-improvement through interaction and feedback.
A brief supervised warm-up stage grounds the Student in basic word structure, after which learning proceeds purely by reinforcement using the Judge's rewards.
The frozen neural Judge (COMET-22) anchors the reward signal, preventing reward drift and hallucination.
Preliminary one-word translation experiments show stable learning dynamics and support the hypothesis that reinforcement learning can surpass data limitations in low-resource language pairs.

1. Introduction

Modern machine-translation systems rely heavily on supervised learning from large bilingual corpora.
However, for language pairs such as **English↔Arabic**, data scarcity limits achievable performance.
We posit that **reinforcement learning (RL)** can overcome this ceiling by learning through reward and interaction rather than imitation alone.
This intuition parallels the transition from **AlphaGo** (supervised on human games) to **AlphaZero**, which exceeded human performance through self-play.
Supervised models imitate; reinforcement models innovate.

We introduce a **Teacher–Student–Judge (T–S–J)** framework that operationalizes this principle for translation.

- **Teacher (B):** presents English challenges of adaptive difficulty.
- **Student (A):** generates Arabic translations.
- **Judge (C):** evaluates validity and semantic fidelity, issuing scalar rewards.

Our key claim:

In low-resource translation settings, hybrid supervised-reinforcement learning can yield competence that surpasses corpus-bound supervised methods.

1.1 Inspiration: From AlphaGo to AlphaZero

The conceptual foundation of this work is inspired by the transition from **AlphaGo** to **AlphaZero** in reinforcement learning.
AlphaGo was trained through *supervised learning* on a large dataset of expert human games, learning to imitate but ultimately bounded by human strategies.
Its successor, **AlphaZero**, began with the same rules but abandoned human data entirely, learning purely through *self-play reinforcement learning* guided by reward signals from game outcomes.
This shift allowed AlphaZero to surpass both humans and its supervised predecessor, demonstrating that reinforcement learning can outperform imitation when clear rules and goals are available.

We draw a parallel between this evolution and language translation.
In our framework, the **Teacher (B)** defines the rules of the game by presenting English challenges, the **Student (A)** plays the game by producing Arabic translations, and the **Judge (C)** provides the reward signal that replaces human supervision.
Just as AlphaZero achieved mastery through self-play, the proposed **Teacher–Student–Judge** loop enables the system to learn translation dynamics from interaction rather than from large, annotated corpora.

2. Related Work

- **Supervised and Distillation Approaches.**
  Traditional neural machine translation (NMT) systems rely on supervised learning from large bilingual corpora.
  Knowledge-distillation models, such as those described by Hinton *et al.* (2015), train smaller "student" models to imitate "teacher" networks but remain limited by dataset size.

- **Reinforcement Learning for Machine Translation.**
  Reinforcement-based methods (Ranzato *et al.*, 2016; Bahdanau *et al.*, 2017) have applied reward functions such as BLEU or COMET to fine-tune translation models.
  However, these approaches lack **adaptive curricula** and **interactive task generation**, restricting their generalization capacity in low-resource settings.
- **Adversarial and Debate Models.**
  GAN-style frameworks (Goodfellow *et al.*, 2014) and debate models (Irving *et al.*, 2018) inspired the tri-agent structure used here.
  Our framework extends this idea by introducing a **fixed Judge** and a **dynamic Teacher**, creating a stable reward-driven environment.
- **Curriculum Learning.**
  Automatic curriculum mechanisms (Bengio *et al.*, 2009) have been shown to improve training efficiency.
  However, few works integrate curriculum design with an independent evaluation agent.
  Our **Teacher–Student–Judge (T–S–J)** system unifies these ideas within a single reinforcement-learning paradigm.
- **Self-Play Reinforcement Learning (AlphaGo and AlphaZero).**
  The foundational inspiration for this work stems from the **AlphaGo** and **AlphaZero** systems developed by DeepMind (Silver *et al.*, 2016; 2018), where supervised learning gave way to pure self-play reinforcement learning.
  Their results demonstrated that **interactive reward-driven learning** can outperform data-limited imitation, a principle we extend to the domain of language translation.

3. Method

3.1 System Overview

The environment comprises three interacting agents: **Teacher (B)**, **Student (A)**, and **Judge (C)**.
Each episode consists of:

1. B generates an English word or phrase.
2. A outputs an Arabic translation (character-by-character).
3. C scores validity and meaning, returning a reward $r \in [-1,1]$ $r \ in [-1, 1]$r∈[−1,1].
4. A updates its policy; B adapts sampling difficulty.

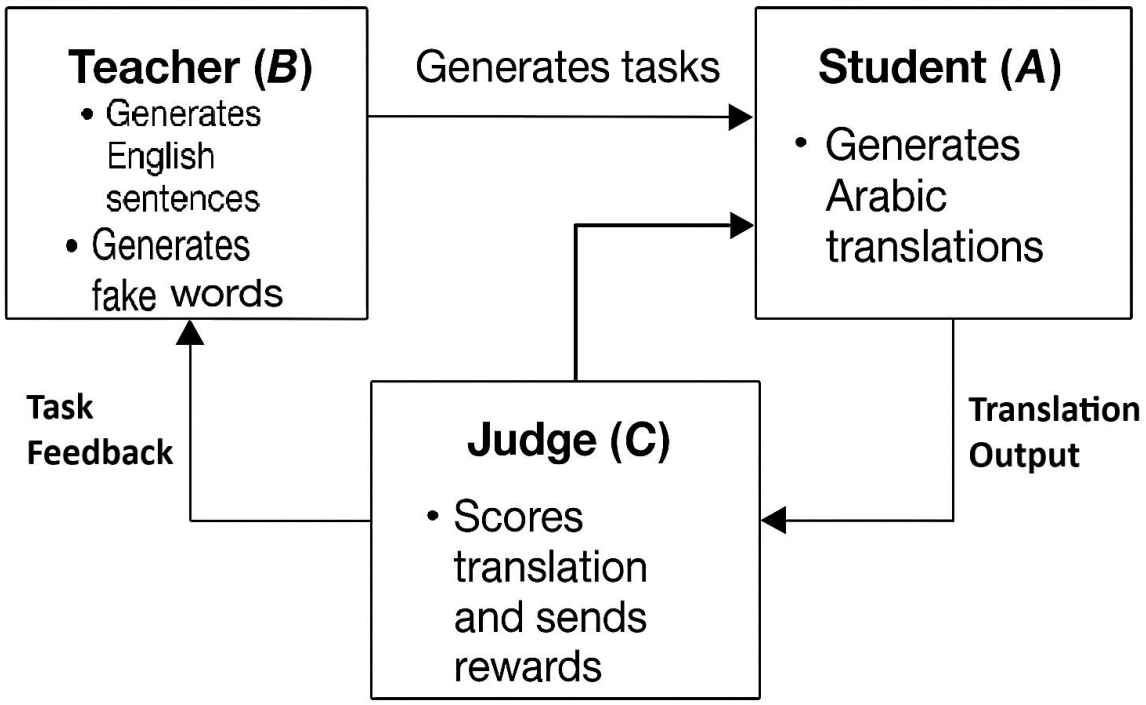

Figure 1. Information and reward flow in the Teacher–Student–Judge tri-agent reinforcement-learning framework.
The Teacher (B) generates English tasks and fake words, the Student (A) produces Arabic translations, and the Judge (C) evaluates them and provides reward feedback to both.

3.2 Student (A)

The Student is the sole learning agent.
Phase-1 employs a **GRU-based recurrent policy** for efficiency, verifying that the framework functions end-to-end on limited hardware.
This model learns to output Arabic letters sequentially, terminating on <eos>.
Training follows a hybrid schedule:

1. **Supervised warm-up:** brief cross-entropy training on a small bilingual lexicon to learn letter combination.
2. **Reinforcement learning:** policy-gradient optimization using Judge rewards plus a light imitation-loss term (0.1×CE).

The long-term plan is to replace the GRU with a **Transformer-based Student** to handle multi-word and syntactic context once Phase-1 validation is complete.

3.3 Teacher (B)

B serves as an **adversarial curriculum generator** whose goal is to cause Student failure within fair linguistic limits.
Initially limited to one-word English tasks, it expands to multi-word phrases as the Student's accuracy surpasses a threshold (≈90 %).
The Teacher maintains difficulty weights and occasionally introduces **fake English words** (≈10 %) to test robustness.
If the Judge deems a prompt invalid, the Teacher receives a penalty.
Future work will replace this adaptive sampler with an English-only language model (e.g., distilled GPT-2 or LLaMA-2-English) for richer phrasing.

3.4 Judge (C)

C is the strongest component and remains **frozen** throughout training.
It anchors the reward function, preventing co-evolution drift.
Implemented using **COMET-22**, it evaluates:

- English validity (does B's word exist and make sense?),
- Arabic validity (are A's outputs grammatical and genuine Arabic?),
- Translation adequacy (semantic equivalence), and
- Correct fake-word tagging.

Rewards combine COMET-22's scalar adequacy with symbolic checks, discretized into {–1, 0, +1}.
C's consistency provides a stable reference signal enabling A and B to co-adapt meaningfully.

4. Prototype Implementation (Phase-1)

Phase-1 experiments focus on single-word English→Arabic translation.
The environment (`RLWordEnv`) encodes English words as bag-of-letters vectors and defines an action space of Arabic letters plus `<eos>`.
Reward shaping blends COMET-22 scores with partial-letter overlap for interpretability.
Training proceeds for several hundred episodes, logging rewards, judge details, and Teacher difficulty adjustments.

This stage verifies the feasibility of the tri-agent loop before scaling to multi-word translation.

5. Datasets and Splits

A small handcrafted bilingual lexicon (≈10 common nouns) forms the supervised warm-up set:
{book, sun, moon, sky, car, water, tree, house, day, night, fire}.
Words are split 70 / 20 / 10 % for warm-up, validation, and test reference.
The environment also generates **out-of-vocabulary (OOV)** fake words for robustness.
Because the main goal is reinforcement learning rather than corpus fitting, dataset scale remains intentionally minimal; future phases will incorporate larger public resources (Tatoeba, OPUS).

6. Evaluation Metrics & Protocol

We evaluate both learning dynamics and translation quality using:

- **Average Episode Reward** and **Reward Variance** (learning stability)
- **Exact-Match Accuracy (EMA)** vs. lexicon
- **Arabic Character Validity Rate (ACVR)** (syntactic correctness)
- **COMET-22 Mean Score** (semantic adequacy)
- **Fake-Word Detection Accuracy (FWDA)** (robustness)

Learning curves of reward and COMET-22 score track progression, while qualitative examples illustrate improved generation structure over time.

7. Discussion

The results highlight how even a lightweight recurrent Student can acquire stable translation behavior through reinforcement alone when guided by a frozen, high-quality Judge.
This setup avoids the instability typical of dual-learning agents by grounding rewards in COMET-22's fixed semantic metric.
Resource limitations motivated the GRU prototype; nevertheless, the system

architecture naturally extends to a Transformer Student for multi-word learning once computational capacity permits.

Resource Considerations.
The initial GRU implementation was chosen for pragmatic efficiency, enabling rapid iteration without large GPUs.
This focus on verifying dynamics precedes scaling to richer Transformer architectures.

8. Limitations

While the proposed framework establishes a proof-of-concept for tri-agent reinforcement learning in translation, it currently faces several limitations.

Computational Constraints.
All experiments were conducted on limited local hardware without access to high-end GPUs.
This restriction required the use of a lightweight GRU-based Student instead of a Transformer, smaller batch sizes, and reduced training iterations.
As a result, the prototype focuses on validating system dynamics rather than achieving optimal translation accuracy.
Future work will revisit these experiments under more capable computational conditions to explore deeper models, longer sequences, and multi-agent scaling.

Scope Limitation.
The current phase is restricted to single-word English→Arabic translation.
Multi-word or sentence-level translation remains future work once the computational budget allows for Transformer-based models and extended reward modelling.

9. Broader Impact

This tri-agent formulation opens a path toward **self-learning translation systems** that require little parallel data.
Beyond translation, similar Teacher–Student–Judge frameworks could inform adaptive language tutoring, cross-lingual reasoning, and low-resource education technology.

10. Conclusion

We introduced a tri-agent reinforcement framework (Teacher–Student–Judge) for low-resource translation.
Phase-1 prototypes confirm feasibility and training stability.
Future work will extend to multi-word and sentence-level translation with Transformer Students, larger datasets, and multi-judge ensembles.
This approach demonstrates that reinforcement learning can break the data-dependence barrier of conventional supervised translation.

11. Acknowledgements

The author occasionally used **OpenAI's ChatGPT (GPT-5)** to assist with phrasing, idea clarification, and proofreading. All conceptual development, analysis, and final writing decisions were made by the author.

12. References

Hinton, G., Vinyals, O., & Dean, J. (2015). *Distilling the Knowledge in a Neural Network*. arXiv:1503.02531

Ranzato, M. et al. (2016). *Sequence Level Training with Recurrent Neural Networks*. arXiv:1511.06732

Bahdanau, D., et al. (2017). *Actor-Critic for Sequence Prediction*. arXiv:1607.07086

Goodfellow, I., et al. (2014). *Generative Adversarial Nets*. arXiv:1406.2661

Irving, G., Christiano, P., et al. (2018). *AI Safety via Debate*. arXiv:1805.00899

Bengio, Y., Louradour, J., Collobert, R., & Weston, J. (2009). *Curriculum Learning*. *Proceedings of ICML 2009*

Silver, D., et al. (2016). *Mastering the Game of Go with Deep Neural Networks and Tree Search*. *Nature*, 529(7587), 484–489.

Silver, D., et al. (2018). *A General Reinforcement Learning Algorithm that Masters Chess, Shogi, and Go through Self-Play*. *Science*, 362(6419), 1140–1144.

Rei, R., et al. (2022). *COMET-22: A Reference-Free Evaluation Metric for Machine Translation*. arXiv:2209.15649

Sweilmeen, M. A. (2025). *Tri-Agent Reinforcement Learning for Low-Resource English→Arabic Translation* https://github.com/Mohswalm545/Mohswalm545-Tri-Agent-Reinforcement-Learning-for-Low-Resource-English-Arabic-Translation