

Wortschatzspiel-Pilot-2026: Repository Overview

Thomas F. Hallmark

Department of Teaching, Learning, and Culture
Texas A&M University, College Station, TX, USA
Email: thomas.hallmark@tamu.edu

Abstract—This document summarizes the structure and purpose of the *Wortschatzspiel-Pilot-2026* repository, which supports a reinforcement learning (RL)-enhanced micro-tutoring system for first-semester German learners and is aligned with the Common European Framework of Reference for Languages (CEFR) A1 proficiency level. The pilot focuses on human-readable YAML (YAML Ain't Markup Language) linguistic rules, adaptive feedback, and RL-driven instructional strategies implemented over a three-week period in Spring 2026. The repository is designed to underpin submissions to the American Society of Engineering Education (ASEE)- Gulf Southwest (GSW), ASEE National 2026, and Junior Researchers of the European Association for Research on Learning and Instruction (EARLI) (JURE) 2026, while also aligning with dissertation and Fulbright Germany research planning.

Index Terms—reinforcement learning, language learning, German vocabulary acquisition, adaptive tutoring systems, micro-tutoring, CEFR A1, computer-assisted language learning, intelligent tutoring systems, reflexive reciprocity theory, educational AI, spaced repetition, feedback scaffolding, cross-cultural education

I. PROJECT SUMMARY

This repository documents the design, implementation, and pilot evaluation of **Wortschatzspiel**, a reinforcement learning (RL)-enhanced micro-tutoring system for German vocabulary acquisition. The system targets first-semester university German learners and adults at the Common European Framework of Reference for Languages (CEFR) A1 level, with a specific focus on American learners transitioning into German-speaking academic or professional contexts.

The pilot study, conducted over three weeks in Spring 2026, investigates how adaptive AI-driven instructional strategies can support reflexive vocabulary learning—a process in which both the learner and the system co-evolve through reciprocal feedback loops. This work operationalizes **Reflexive Reciprocity Theory (RRT)**, examining the dynamic interplay between human pedagogical intuition and algorithmic instructional discovery.

II. CORE COMPONENTS

A. YAML-Encoded Linguistic Rules

The system employs structured YAML files to represent German linguistic patterns, morphological rules, and contextual usage constraints. This human-readable format enables:

- **Transparent rule representation:** Instructors and researchers can inspect, modify, and validate linguistic rules without programming expertise
- **Version control:** Linguistic rule evolution can be tracked across pilot iterations
- **Interoperability:** YAML structures integrate seamlessly with both Python-based RL agents and web-based tutoring interfaces

B. Adaptive Feedback Engine

The tutoring system implements multi-dimensional feedback strategies that adjust to learner performance in real-time:

- **Error-type classification:** Distinguishes between orthographic, morphological, syntactic, and semantic errors
- **Scaffolded hints:** Provides graduated support (implicit cues → explicit examples → direct correction) based on error patterns
- **Temporal spacing:** Applies spaced repetition algorithms informed by RL policy optimization
- **Contextual reinforcement:** Adjusts feedback specificity based on vocabulary difficulty, learner history, and session dynamics

C. Reinforcement Learning–Driven Instruction

The system employs RL algorithms to optimize instructional sequencing and feedback timing:

- **State representation:** Encodes learner knowledge state, session context, vocabulary characteristics, and error history
- **Action space:** Defines instructional moves (present new vocab, review difficult items, provide contextual examples, adjust difficulty)
- **Reward signal:** Balances immediate performance (accuracy) with long-term retention (post-test scores, session-to-session improvement)
- **Policy optimization:** Uses actor-critic methods to discover effective tutoring strategies that emerge from learner interaction data

D. Three-Week Pilot Design

The Spring 2026 pilot involves:

- **Participant recruitment:** 30–40 first-semester German students at a large research university in the U.S.
- **Baseline assessment:** Pre-pilot vocabulary knowledge evaluation (CEFR A1 diagnostic)
- **Intervention period:** 3 weeks of daily 15-minute micro-tutoring sessions (15 sessions total)
- **Control condition:** Traditional flashcard-based vocabulary practice with static sequencing
- **Treatment condition:** RL-enhanced adaptive tutoring with dynamic feedback
- **Post-assessment:** Immediate post-test + 2-week retention follow-up

III. RESEARCH CONTEXT

A. Theoretical Framework: Reflexive Reciprocity Theory

This project extends RRT by examining:

- **Pedagogical intuition:** How instructors' implicit knowledge about effective vocabulary teaching can be formalized into RL reward structures
- **Algorithmic discovery:** How RL agents identify instructional patterns invisible to human tutors (e.g., optimal spacing intervals for specific learner profiles)
- **Co-learning dynamics:** How system-learner interaction produces emergent instructional strategies that neither human designers nor algorithms could generate independently

B. Dissertation Integration

Wortschatzspiel serves as the second empirical study in a three-paper dissertation:

- 1) **Paper 1 (PedagoReLearn):** RL framework for adaptive cross-cultural competence training
- 2) **Paper 2 (Wortschatzspiel):** RL-enhanced vocabulary learning for language acquisition
- 3) **Paper 3 (Comparative Analysis):** RRT as a unifying framework across domains

C. Fulbright Germany Research Alignment

This pilot prepares for a 2026–2027 Fulbright research stay at Rheinland-Pfälzische Technische Universität (RPTU) Kaiserslautern-Landau, Germany. The German phase will:

- Expand pilot to German university contexts (A2/B1 learners, German students learning academic English)
- Integrate German pedagogical perspectives on AI-enhanced language learning
- Conduct comparative analysis of RL-driven instruction across US and German educational contexts

IV. CONFERENCE SUBMISSION TARGETS

A. ASEE Gulf Southwest (GSW) 2026

Preliminary pilot results, system architecture, and early findings.

B. ASEE National 2026

Complete pilot evaluation, RL vs. traditional comparison, implications for engineering education.

C. JURE 2026

Theoretical contribution (RRT operationalization), cross-cultural considerations, and collaboration potential.

V. TECHNICAL STACK

- **Frontend:** React 18, Next.js 14, Tailwind CSS
- **Backend:** Python 3.11, FastAPI, PostgreSQL
- **RL Framework:** Stable-Baselines3 (PPO/A2C algorithms)
- **NLP Processing:** spaCy (German language models), Hugging Face Transformers
- **Data Storage:** PostgreSQL (learner profiles, session logs), Redis (session state)
- **Analytics:** Pandas, NumPy, Matplotlib, Seaborn, SciPy

VI. KEY RESEARCH QUESTIONS

- RQ1: Do learners using RL-enhanced micro-tutoring demonstrate greater vocabulary acquisition than those using traditional flashcard methods?
- RQ2: Does adaptive feedback improve long-term retention compared to static instructional sequences?
- RQ3: How do learners perceive AI-driven adaptive feedback in terms of helpfulness, transparency, and trust?
- RQ4: What novel tutoring strategies emerge from RL optimization that differ from traditional approaches?
- RQ5: How does the Wortschatzspiel case study inform RRT as a framework for AI-enhanced education?

VII. TIMELINE

- January 2026: Institutional Review Board (IRB) approval, recruitment, system finalization
- February 2026: Pilot launch (3-week intervention)
- March 2026: Data analysis, ASEE GSW submission
- April–May 2026: ASEE National paper preparation
- June–July 2026: ASEE National and JURE presentations
- Fall 2026: Dissertation integration, Fulbright preparation

VIII. SCHOLARLY CONTRIBUTIONS

This project advances:

- **AI in Education:** Practical RL application in authentic contexts
- **Language Learning:** Extends CALL through adaptive AI
- **Engineering Education:** Supports technical language for international work
- **Cross-Cultural Research:** Enables US–Germany comparative pedagogy
- **Theory:** Operationalizes RRT through empirical case studies

IX. INSTALLATION AND SETUP

Detailed setup instructions are provided in README.md. Quick start:

- 1) Clone repository:
`git clone https://github.com/[username]/Wortschatzspiel-Pilot-2026`
- 2) Install dependencies:
`pip install -r requirements.txt`
- 3) Configure environment:
`cp development.env.example development.env`
- 4) Initialize database:
`python -m backend.database.init_db`
- 5) Launch development server:
`docker-compose up`

System requirements: Python 3.11+, Node.js 18+, PostgreSQL 14+, 8GB RAM minimum (16GB recommended for RL training).

X. ETHICS AND DATA MANAGEMENT

This study has received approval from the Texas A&M University Institutional Review Board (IRB Protocol #[PENDING]). All participants provide informed consent prior to enrollment, and personally identifiable in-

formation is stored separately from performance data using anonymization protocols detailed in the repository's `data_collection/privacy/` directory.

Data retention follows university research data policies: raw session logs retained for 3 years post-publication, anonymized datasets available upon reasonable request to qualified researchers, and all data processing complies with FERPA regulations for educational records.

XI. REPOSITORY STRUCTURE

The repository is organized to separate configuration, RL implementation, tutoring interface, data collection, and dissemination assets, as shown in Figs. 1 and 2.

XII. LICENSE AND USAGE

This repository is released under the MIT License, permitting free use, modification, and distribution with attribution. The linguistic rule files (`linguistic_rules/`) are additionally available under Creative Commons BY-SA 4.0 to encourage community contributions from language educators and researchers.

The complete license text is included in the `LICENSE` file in the repository root.

Researchers wishing to replicate or extend this work should cite this repository as:

Hallmark, T. F. (2026). *Wortschatzspiel-Pilot-2026: Reinforcement Learning-Enhanced Micro-Tutoring for German Vocabulary Acquisition*. GitHub.

ACKNOWLEDGMENTS

This research is supported by the U.S. Department of Veterans Affairs Vocational Rehabilitation and Employment (VR&E) program. The author gratefully acknowledges doctoral advisors Dr. Karen Rambo-Hernández and Dr. Ali Bicer at Texas A&M University, and international mentor Prof. Dr. Leo van Waveren at RPTU Kaiserslautern-Landau, Germany, for their guidance in developing the Reflexive Reciprocity Theory framework.

Fulbright Germany support for the 2026–2027 research stay is pending and will enable expansion of this pilot to German university contexts. The author thanks the Department of Teaching, Learning, and Culture at Texas A&M University for providing research facilities and Mr. Petr Kandidatov and Dr. David Brenner from the the Department of Global Languages & Cultures for pilot participant recruitment support.

The author further acknowledges the open-source communities maintaining Stable-Baselines3, spaCy, React, Next.js, and FastAPI, which form the technical foundation of this system.

DATA AVAILABILITY

Anonymized pilot data will be made available via the Open Science Framework (OSF) repository upon publication of the primary research findings. Raw data containing personally identifiable information will be retained securely at Texas A&M University in accordance with IRB protocols and institutional data retention policies.

COMPETING INTERESTS

The author declares no competing financial or non-financial interests that could inappropriately influence this research.

CONTACT INFORMATION

For questions, collaboration inquiries, or technical documentation, contact:

Thomas F. Hallmark

PhD Candidate, Curriculum & Instruction
(Engineering Education)
Texas A&M University
Email: thomas.hallmark@tamu.edu
ORCID: 0009-0002-3124-8140
GitHub: [aggiewissenschaftler](#)

```

Wortschatzspiel-Pilot-2026/
|
+-- README.md                                     # Project overview and setup
+-- LICENSE                                         # Open-source license
+-- .gitignore                                       # Git exclusions
+-- requirements.txt                                # Python dependencies
+-- docker-compose.yml                            # Container orchestration
|
+-- linguistic_rules/                           # YAML-encoded German rules
|   +-- vocab/                                 # A1-level vocabulary (500 words)
|       +-- cefr_a1_core.yaml      # German-English cognates
|       +-- cognates.yaml        # Topic-based groupings
|       \-- thematic_clusters.yaml
|   +-- grammar/                               # Present tense conjugation
|       +-- verbs_present.yaml    # Perfect tense (sein/haben)
|       +-- verbs_perfect.yaml   # Können, müssen, wollen, etc.
|       +-- modal_verbs.yaml     # Word order rules (V2, VL)
|       +-- sentence_structure.yaml # Negation
|       +-- imperative.yaml      # Command forms
|       \-- cases_intro.yaml      # Nominative/accusative intro
|   \-- phonology/                             # Ä, Ö, Ü pronunciation
|       +-- umlaute.yaml        # Stress in compounds
|       \-- compound_stress.yaml
|
+-- rl_agent/                                    # Reinforcement learning core
|   +-- environment.py          # OpenAI Gym-compatible env
|   +-- state_representation.py # Learner state encoding
|   +-- action_space.py         # Tutoring action definitions
|   +-- reward_functions.py    # Reward signal computation
|   +-- policies/               # A2C/PPO implementation
|       +-- actor_critic.py    # Random policy (control)
|       \-- baseline_expert.py # Rule-based expert policy
|   +-- training/                # RL training loop
|       +-- train_agent.py     # Learning rate, discount, etc.
|       \-- checkpoints/        # Saved model weights
|   \-- utils/                      # TensorBoard integration
|       +-- logging.py          # Policy evaluation metrics
|
+-- tutoring_interface/                         # Web-based learner UI
|   +-- frontend/
|       +-- components/
|           |   +-- VocabCard.jsx      # Flashcard component
|           |   +-- FeedbackPanel.jsx # Adaptive hint display
|           |   +-- ProgressTracker.jsx # Session progress bar
|           |   \-- GrammarExplainer.jsx # Rule visualization
|       +-- pages/
|           |   +-- index.jsx        # Landing page
|           |   +-- session.jsx     # Active tutoring session
|           |   +-- profile.jsx      # Learner dashboard
|           |   \-- post-test.jsx    # Assessment interface
|       \-- styles/                  # Tailwind configuration
|           \-- globals.css
|
+-- backend/                                     # Session management
|   +-- api/
|       +-- session.py          # Vocabulary retrieval
|       +-- vocab.py            # Adaptive feedback logic
|       \-- rl_integration.py   # RL agent API wrapper
|   +-- database/                             # SQLAlchemy models
|       +-- models.py           # PostgreSQL schema
|       +-- schema.sql          # Alembic migrations
|       \-- migrations/
|   \-- nlp/                                  # Learner response parsing
|       +-- error_detection.py  # German NLP processing
|       +-- spacy_pipeline.py   # Response-target comparison
|
\-- config/                                     # Dev environment variables
    +-- development.env      # Production config
    \-- production.env
|
|

```

Wortschatzspiel-Pilot-2026 Repository Tree continues on the next page →

Fig. 1: Directory structure of the Wortschatzspiel-Pilot-2026 repository (top-level through tutoring interface).

```

Wortschatzspiel-Pilot-2026 Repository Tree (continued)
|
|
|--- data_collection/                      # Pilot study data
|   |--- raw/
|   |   |--- session_logs/                 # Timestamped interactions
|   |   |--- learner_responses/           # Text input, accuracy
|   |   |--- rl_decisions/                # Agent action log
|   |   \--- system_metrics/              # Response times, errors
|   |--- processed/
|   |   |--- aggregated_sessions.csv    # Session-level summaries
|   |   |--- learner_trajectories.csv   # Learning curves
|   |   \--- error_patterns.csv         # Error type frequencies
|   \--- privacy/
|       |--- anonymization.py          # PII removal scripts
|       \--- consent_records/          # Signed consent forms (local)
|
|--- evaluation_metrics/                  # Assessment instruments
|   |--- pre_test/
|   |   |--- vocab_diagnostic.pdf      # AI vocabulary assessment
|   |   |--- grammar_diagnostic.pdf    # Basic grammar test
|   |   \--- scoring_rubric.md        # Grading guidelines
|   |--- post_test/
|   |   |--- immediate_posttest.pdf    # Day 21 assessment
|   |   |--- retention_followup.pdf    # 2-week delayed test
|   |   \--- transfer_tasks.pdf        # Novel sentence generation
|   |--- surveys/
|   |   |--- user_experience.pdf      # System usability scale
|   |   |--- cognitive_load.pdf       # NASA-TLX adapted
|   |   \--- motivation_questionnaire.pdf # Intrinsic motivation
|   \--- scoring_scripts/
|       |--- auto_grade.py            # Automated scoring
|       \--- inter_rater_reliability.py # Cohen's kappa calculation
|
|--- pilot_protocols/                   # IRB and study procedures
|   |--- irb_application.pdf          # Texas A&M IRB submission
|   |--- informed Consent.pdf        # Participant consent form
|   |--- recruitment_flyer.pdf       # Study advertisement
|   |--- participant_instructions.pdf # Session guidelines
|   |--- debriefing_script.md        # Post-study explanation
|   |--- timeline_gantt.png          # 3-week schedule
|   |--- participant_flowchart.png   # Enrollment diagram
|   \--- data_management_plan.md      # Storage, security, retention
|
|--- analysis_scripts/                 # Statistical analysis
|   |--- descriptive_stats.py        # Means, SDs, distributions
|   |--- learning_curves.py          # Session-by-session progress
|   |--- hypothesis_testing.py       # t-tests, ANOVA, effect sizes
|   |--- retention_analysis.py      # Forgetting curves
|   |--- rl_policy_visualization.py # Action frequency heatmaps
|   |--- qualitative_coding.py       # Survey response themes
|   \--- figures/
|       |--- learning_efficiency.png # Treatment vs. control
|       |--- retention_comparison.png # 2-week follow-up
|       |--- rl_strategy_emergence.png # Policy evolution
|       \--- learner_experience.png   # Survey results
|
\--- conference_papers/               # Manuscript drafts
    |--- ASEE_GSW_2026/
    |   |--- draft.tex                # LaTeX manuscript
    |   |--- figures/                 # Paper-specific figures
    |   \--- submission_checklist.md   # Formatting requirements
    |--- ASEE_National_2026/
    |   |--- draft.tex                # Conference presentation
    \--- JURE_2026/
        |--- draft.tex
        |--- figures/
        \--- submission_guidelines.pdf

```

Fig. 2: Directory structure of the Wortschatzspiel-Pilot-2026 repository (continued from Fig. 1).