# HaSPI – Hate Speech Prevention Through Imitation

Endbericht | Call 19 | Projekt ID 7207



Inhalt

# 1 Einleitung

Das Projekt HaSPI – Hate Speech Prevention Through Imitation (Projekt-ID: 7207, Call #19, Förderjahr 2024) widmete sich der Bekämpfung von Hate Speech im deutschsprachigen Internet. Trotz zahlreicher technischer Ansätze bleibt Hate Speech ein drängendes Problem, insbesondere da viele bestehende Lösungen entweder plattformunabhängig konzipiert oder auf englischsprachige Inhalte beschränkt sind.

Mit HaSPI wurde ein innovativer Ansatz validiert und prototypisch implementiert, der auf Imitation Learning basiert: Anstatt Hate Speech ausschließlich über statisch gelabelte Datensätze zu klassifizieren, ahmt das System menschliche Moderationsentscheidungen nach. Als Datenbasis diente der „One-Million-Posts“-Korpus der Tageszeitung DER STANDARD, der eine reichhaltige Sammlung deutschsprachiger Online-Kommentare samt Moderationsentscheidungen bietet. Durch die Berücksichtigung von Nutzerverhalten und thematischem Kontext wurden Modelle entwickelt, die Hate Speech nicht nur erkennen, sondern deren Entscheidungen auch nachvollziehbar begründet werden können (Explainability).

Als zentrale Ergebnisse liegen ein Open-Source-Framework für Imitation-Learning-basierte Inhaltsmoderation (veröffentlicht unter https://github.com/fhstp/HaSPI, Apache-2.0-Lizenz), eine wissenschaftliche Publikation (iDSC 2025, Preprint: https://arxiv.org/abs/2505.20963) sowie eine im Projektkontext abgeschlossene Masterarbeit vor.

# 2 Projektbeschreibung

Das übergeordnete Ziel von HaSPI war die Entwicklung eines Open-Source-Frameworks, das speziell auf die Moderation deutschsprachiger Inhalte ausgerichtet ist. Neben der Validierung des Imitation-Learning-Ansatzes stand die Erklärbarkeit der Entscheidungen im Fokus, um Vertrauen und Transparenz in automatisierte Moderationssysteme zu fördern. Zielgruppen sind Betreiber deutschsprachiger Online-Plattformen und deren Moderationsteams sowie die Forschungscommunity in den Bereichen NLP und Reinforcement Learning.

Das Projektergebnis umfasst die folgenden Komponenten: eine Datenaufbereitungs-Pipeline für den One-Million-Posts-Korpus (Bereinigung, Train/Test-Split, Vektorrepräsentation via GBERT-Large, Dimensionsreduktion), ein performantes RL-Environment für die automatisierte Textklassifikation sowie JAX-Implementierungen des IQ-Learn-Algorithmus und den getesteten Erweiterungen. Sämtliche Komponenten sind unter https://github.com/fhstp/HaSPI unter Apache-2.0-Lizenz öffentlich verfügbar.

Ergänzend zum Framework entstand im Projektkontext die Masterarbeit von Felix Krejca („Automated Content Moderation for German Newspaper Comments Utilizing Transformer-Based Models with Textual Context“, Masterstudiengang Data Intelligence, University of Applied Sciences St. Pölten, abgeschlossen Juni 2026), die den Einfluss textueller Kontextinformationen auf Transformer-basierte Moderationsmodelle systematisch untersucht und damit die kontextsensitive Ausrichtung des Projekts wissenschaftlich vertieft.

# 3 Verlauf der Arbeitspakete

## 3.1 Arbeitspaket 1 – Detailplanung und Formales am Projektstart

Arbeitspaket 1 wurde bereits mit April 2025 erfolgreich abgeschlossen. Die folgenden Arbeiten wurden in diesem initialen Arbeitspaket durchgeführt:

- der Vertrag wurde unterschrieben,
- der Detailprojektplan wurde erstellt und abgenommen,
- eine detaillierte Liste der geplanten Projektergebnisse mit Lizenz und Ort der öffentlichen Bereitstellung wurde erstellt und abgenommen,
- die Projekt-Website ging in Betrieb, ein erster Blogeintrag wurde erstellt und die erste Förderrate wurde beantragt.

Es gab keine Abweichungen vom Plan.

## 3.2 Arbeitspaket 2 – Projektmanagement, Outreach und Dissemination

Dieses Arbeitspaket umfasste die organisatorische Koordination des Projekts sowie die externe Kommunikation und wissenschaftliche Dissemination über die gesamte Projektlaufzeit: Blogposts zum Projektfortschritt, Zwischenbericht (M6–M9), Endbericht (M12) sowie laufende wissenschaftliche Publikationen. Die Projektkoordination erfolgte über regelmäßige Team-Meetings und ein gemeinsames GitHub-Repository zur Versionsverwaltung und Aufgabenkoordination. Der Zwischenbericht wurde fristgerecht erstellt; der vorliegende Endbericht schließt das Arbeitspaket ab.

**Wissenschaftliche Dissemination:** Das Paper „Context-Aware Content Moderation for German Newspaper Comments“ wurde veröffentlicht und auf der iDSC-Konferenz 2025 an der FH Salzburg von Felix Krejca präsentiert (Preprint: https://arxiv.org/abs/2505.20963). Das Projekt wurde zudem am Forschungsforum der österreichischen Fachhochschulen (7.–8. Mai 2025, FH Campus Wien) vorgestellt sowie im **Jänner** 2026 im Data Intelligence Research Seminar der FH St. Pölten und im März 2026 im Gruppenmeeting der Gruppe “Probabilistic and Interactive Machine Learning “ auf der Universität Wien unter dem Titel „Sequence Classification through Reinforcement Learning“ präsentiert. Im Rahmen des Projekts wurde darüber hinaus die Masterarbeit von Felix Krejca zu kontextsensitiver Moderation deutschsprachiger Zeitungskommentare abgeschlossen (siehe Kapitel 2). Eine weitere wissenschaftliche Publikation zur Imitation-Learning-basierten Klassifikation ist derzeit in Arbeit.

**Blogposts zur Projektkommunikation** (https://www.netidee.at/haspi):

- „Introducing HaSPI“ (Jänner 2025)
- „HaSPI at Forschungsforum 2025“ (April 2025)
- „Hate doesn’t only speak English“ (Oktober 2025)
- „Sequence Classification through Reinforcement Learning“ (März 2026)

Ergänzend wurden das Projekt und zentrale Ergebnisse über LinkedIn und Instagram sowie über die offiziellen Kanäle der FH St. Pölten kommuniziert, sowie auf dem “Heurika findet Stadt!” Festival 2026 vorgestellt. Es gab keine wesentlichen Abweichungen vom ursprünglichen Arbeitsplan; alle geplanten Disseminationsmaßnahmen konnten umgesetzt werden.

### 3.3 Arbeitspaket 3 – Datenaufbereitung und RL-Umgebung

Zu Beginn des Arbeitspakets erfolgte eine systematische Literatursuche, um aktuelle Forschungsarbeiten im Bereich Imitation Learning zu identifizieren. Hierfür kamen die Datenbanken Google Scholar, IEEE Xplore, ACM Digital Library und Springer Link zum Einsatz. Berücksichtigt wurden Publikationen ab dem Jahr 2024; die jeweils 50 relevantesten Treffer wurden in Zotero importiert. Nach der Dublettenbereinigung verblieben 149 Einträge, von denen nach Sichtung von Titeln und Abstracts 48 Arbeiten für eine vertiefte Analyse ausgewählt wurden. Ergänzend wurde das IQ-Learn-Paper, auf dessen Konzept das Projekt aufbaut, in die Analyse aufgenommen; über Google Scholar und Semantic Scholar wurden 59 weitere zitierende Publikationen identifiziert und hinsichtlich ihrer Relevanz geprüft. Nach Dublettenbereinigung ergab sich eine finale Literaturliste mit 91 Publikationen, ergänzt um fünf weitere Arbeiten aus den Introduction- und Related-Work-Abschnitten der gesichteten Publikationen.

Aufbauend auf dieser Literaturrecherche wurde ein prototypisches Reinforcement-Learning-Environment für die automatisierte Textklassifikation entwickelt, das vor allem dazu diente, Anforderungen an Modellarchitektur und Trainingsumgebung besser einschätzen zu können. Der IQ-Learn-Algorithmus wurde zunächst in PyTorch implementiert, um das Environment zu testen. Dabei wurden Speicherrestriktionen im Replay-Buffer identifiziert, die die Verwendung der Daten im Vollumfang erschwert hätten. Um diese Einschränkungen zu adressieren, wurde der Soft Actor-Critic (SAC)-Algorithmus in JAX neu implementiert und mit der PyTorch-Version verglichen. Die Experimente zeigten einen rund 40-fachen Geschwindigkeitszuwachs in typischen Benchmark-Umgebungen bei Verwendung von JAX. Aufgrund dieser deutlichen Effizienzsteigerung erfolgte eine vollständige Re-Implementierung sowohl des Text-Environments als auch des IQ-Learn-Algorithmus in JAX. Diese Umstellung brachte unter Realbedingungen eine Beschleunigung um den Faktor 16 (ca. 1250 statt ca. 75 Trainingsschritte pro Sekunde auf einer Nvidia RTX 3090) sowie eine verbesserte Kontrolle über den Speicherverbrauch, wodurch JAX als bevorzugtes ML-Framework des Projekts etabliert wurde.

Parallel dazu wurde eine Datenaufbereitungs-Pipeline für den One-Million-Posts-Korpus erstellt: Die Userkommentare werden bereinigt und in Trainings- und Testdatensätze unterteilt; anschließend werden vortrainierte Open-Source-Transformer-Modelle (z. B. GBERT-Large) eingesetzt, um Vektorrepräsentationen der Kommentare zu generieren. Zur effizienten Weiterverarbeitung im Imitation Learning wird zudem eine Dimensionsreduktion dieser Vektoren durchgeführt. Als Vorbereitung auf den Einsatz rekurrenter Architekturen (LSTM/xLSTM) wurde außerdem die Lambda-Discrepancy für das zugrundeliegende Lernproblem implementiert. Erste Ergebnisse blieben hinter den Erwartungen zurück,

jedoch wurden kritische Implementierungsaspekte identifiziert, die in Arbeitspaket 4 aufgegriffen wurden.

Größere Abweichungen vom ursprünglichen Projektplan traten nicht auf. Die Umstellung auf JAX war nicht vorgesehen, stellte sich jedoch als strategisch sinnvolle Anpassung heraus: Sie trug wesentlich dazu bei, die Forschungsziele effizienter zu erreichen und die Entwicklung rekurrenter Agenten in Arbeitspaket 4 auf eine solide technische Basis zu stellen.

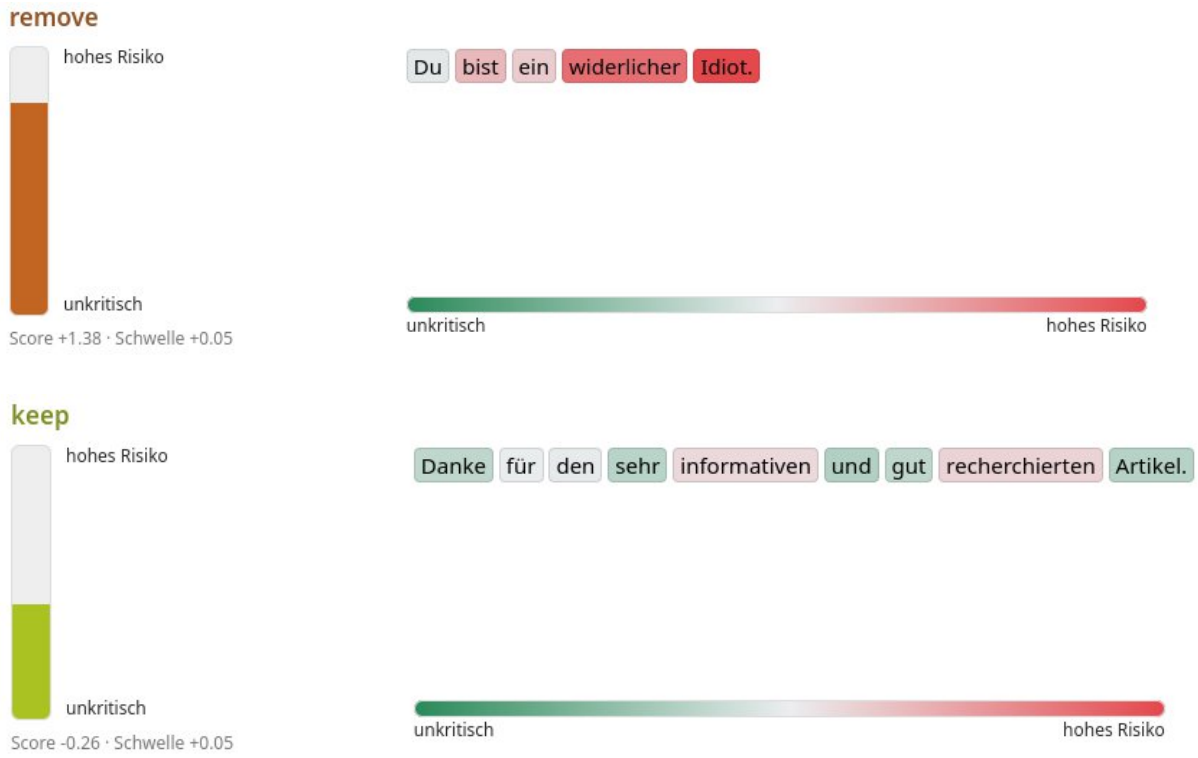
### 3.4 Arbeitspaket 4 – Implementierung und Evaluierung

Aufbauend auf den in Arbeitspaket 3 gewonnenen Erkenntnissen wurde die Imitation-Learning-Pipeline vollständig implementiert. Der Fokus lag dabei auf einer effizienten Nutzung der neuen internen Server-Infrastruktur der FH St. Pölten, die auch Finetuning und Training von LLMs ermöglicht. Parallel dazu wurden Grundlagen für eine xLSTM-Architektur erforscht, um die Leistungsfähigkeit rekurrenter Modelle in komplexen Imitation-Learning-Szenarien als Alternative zu Transformern zu evaluieren. Zudem wurde die theoretische Grundlage zur Anwendung von Imitation Learning für Klassifikationsaufgaben ausgearbeitet – als Vorbereitung auf die wissenschaftliche Publikation der Projektergebnisse, die sich derzeit in Arbeit befindet. Zwischenergebnisse wurden im März 2026 im Data Intelligence Research Seminar der University of Applied Sciences St. Pölten präsentiert und als Blogbeitrag öffentlich dokumentiert.

Die Kerntätigkeit im Arbeitspaket war der systematische Test der Komponenten. Die Analyse lieferte ein starkes neues Verständnis für Imitation Learning als Grundlage für Klassifikationen generell, sowie im Speziellen für den Use Case der automatischen Content-Moderation und textuellen Daten. Optimiert wurde unter anderem auch in Hinblick auf benötigte Hardware-Ressourcen. Insgesamt hat sich gezeigt, dass ein reines Reward-Training (ohne Actor-Training!) im Durschnitts-Reward Reinforcement Learning Setting auf einen performanten, aber eingefrorenem vortrainierten Transformer (LeoLM) die besten Resultate liefert. Dies wurde in ein leicht zu verwendendes PyPI-Paket verarbeitet. Dies bietet insbesondere den Vorteil, dass dabei im Gegensatz zu “reinem” IQLearn kein generativer Hate-Speech Bot als Zwischenresultat entsteht, wodurch das im Antrag angeführte Dual-Use Risiko methodisch gänzlich verhindert werden konnte.

Im Rahmen des Arbeitspakets wurde zudem die Masterarbeit von Felix Krejca abgeschlossen („Automated Content Moderation for German Newspaper Comments Utilizing Transformer-Based Models with Textual Context“, FH St. Pölten, Mai 2026). Die Arbeit untersucht auf Basis des One-Million-Posts-Korpus systematisch, wie sich fünf Stufen kontextueller Information – von isolierten Kommentaren über Artikelmetadaten bis hin zu hierarchischen Diskussionsverläufen – auf die Moderationsleistung auswirken, und vergleicht parameter-effizient nachtrainierte Encoder-Modelle (ModernGBERT) mit generativen Large Language Models (u. a. Llama 3 und Gemma). Die Ergebnisse zeigen, dass der Nutzen von Kontextinformation und Retrieval-Augmented Generation (RAG) stark von Modellgröße und Integrationsstrategie abhängt und dass feinjustierte Encoder-Modelle mit deutlich größeren generativen Modellen konkurrenzfähig sind. Diese Erkenntnisse fließen direkt in die kontextsensitive Ausrichtung des HaSPI-Frameworks ein.

Unsere finalen Evaluierungsergebnisse zeigen, dass unsere Methode vergleichbar zum State-of-the-Art funktioniert (in unserer Evaluierung etwas besser, allerdings nicht signifikant), während sie aber intrinsische Erklärungen liefert. Diese Erklärungen haben wir auch in einer Demo-Oberfläche visualisiert, siehe Screenshots unten. (Weitere Beispiele finden sich in der Entwickler:innen Dokumentation.)




### 3.5 Arbeitspaket 5 – Dokumentation und Formales am Projektende

Der im Projekt entwickelte Quellcode – Datenaufbereitungs-Pipeline, RL-Environment sowie die JAX-Implementierungen von IQ-Learn und Varianten– wurde in einem öffentlichen GitHub-Repository unter der Apache-2.0-Lizenz veröffentlicht: https://github.com/fhstp/HaSPI. Das Repository enthält die technische Dokumentation zur Nachnutzung und Weiterentwicklung durch Dritte.

Der vorliegende Endbericht sowie die Abschlussabrechnung schließen das Projekt formal ab; die letzte Förderrate wurde beantragt. Es gab keine Abweichungen vom Plan.

# 4 Umsetzung Förderauflagen

In der Fördervereinbarung sind keine speziellen Förderauflagen vorgesehen; entsprechend waren keine gesonderten Maßnahmen umzusetzen.

# 5 Liste Projektendergebnisse

Die folgende Tabelle fasst die erreichten Projektendergebnisse mit Lizenz und Ort der öffentlichen Bereitstellung zusammen:

| Nr. | Projektergebnis | Lizenz | Bereitstellung |
|---|---|---|---|
| 1 | Projektzwischenbericht | CC BY-SA 4.0 | www.netidee.at/haspi |
| 2 | Projektendbericht | CC BY-SA 4.0 | www.netidee.at/haspi |
| 3 | Entwickler:innen-Dokumentation | CC BY-SA 4.0 | haspi.readthedocs.io |
| 4 | Anwender:innen-Dokumentation | CC BY-SA 4.0 | https://github.com/fhstp/HaSPI/blob/main/README.md |
| 5 | Veröffentlichungsfähiger Einseiter / Zusammenfassung | CC BY-SA 4.0 | www.netidee.at/haspi |
| 6 | Dokumentation Externkommunikation (als Teil des Endberichts) | CC BY-SA 4.0 | www.netidee.at/haspi |
| 7 | Prototypische Softwareimplementierung:<br>Datenaufbereitungs-Pipeline für den One-Million-Posts-Korpus | Apache-2.0 | https://github.com/fhstp/haspi |
| 8 | Paket für Endanwender:innen API auf PyPi:<br>* Interface für Hate Speech Detection<br>* Interface für Erklärungen der Entscheidungen | Apache-2.0 | https://pypi.org/project/haspi/ |
| 9 | Code für Endanwender:innen API: HaSPI-Framework: Imitation-Learning-Pipeline für Hate-Speech-Erkennung (JAX-Implementierungen von IQ-Learn und Varianten; RL-Environment für Textklassifikation) | Apache-2.0 | https://github.com/fhstp/haspi |
| 10 | Wissenschaftliche Publikation: „Context-Aware Content Moderation for German Newspaper Comments“ (iDSC 2025) | CC BY-SA 4.0 | arxiv.org/abs/2505.20963 |

# 6 Verwertung der Projektergebnisse in der Praxis

Das HaSPI-Framework steht als Proof of Concept unter Apache-2.0-Lizenz öffentlich zur Verfügung und kann von Plattformbetreibern, Moderationsteams und Forschungseinrichtungen nachgenutzt werden. Die Datenaufbereitungs-Pipeline und die JAX-Implementierungen (IQ-Learn und Varianten) sind modular aufgebaut und auch unabhängig vom Moderationskontext als Bausteine für Reinforcement- und Imitation-Learning-Anwendungen einsetzbar.

Aus dem Projektverlauf lassen sich mehrere Erkenntnisse für die Praxis ableiten: Der nicht geplante, aber strategisch richtige Umstieg von PyTorch auf JAX brachte einen ca. 16-fachen Speedup unter Realbedingungen (bis zu 40-fach in Benchmarks) und eine deutlich bessere Kontrolle über den Speicherverbrauch – eine wesentliche Voraussetzung, um den One-Million-Posts-Korpus im Vollumfang verarbeiten zu können.

Für eine Produktivsetzung empfehlen wir, die Evaluierung auf weitere deutschsprachige Plattformen und Korpora auszudehnen und gemeinsam mit realen Moderationsteams zu validieren.

# 7 Öffentlichkeitsarbeit/ Vernetzung

Im Rahmen der Öffentlichkeitsarbeit wurden über die gesamte Projektlaufzeit Maßnahmen zur Sichtbarmachung des Projekts HaSPI umgesetzt:

- **Wissenschaftliche Dissemination:** Paper „Context-Aware Content Moderation for German Newspaper Comments“, präsentiert auf der iDSC 2025, FH Salzburg (Preprint: https://arxiv.org/abs/2505.20963); Projektpräsentation am Forschungsforum der österreichischen Fachhochschulen, 7.–8. Mai 2025, FH Campus Wien (https://www.netidee.at/haspi/haspi-forschungsforum-2025); Seminarvortrag „Sequence Classification through Reinforcement Learning“, Data Intelligence Research Seminar, FH St. Pölten, März 2026; Masterarbeit Felix Krejca (FH St. Pölten, 2026). Eine weitere Publikation zur Imitation-Learning-basierten Klassifikation ist in Arbeit.
- **Blogbeiträge auf der netidee-Projektseite (https://www.netidee.at/haspi):** „Introducing HaSPI“ (Jänner 2025), „HaSPI at Forschungsforum 2025“ (April 2025), „Hate doesn’t only speak English“ (Oktober 2025), „Sequence Classification through Reinforcement Learning“ (März 2026).
- **Social Media:** Das Projekt und zentrale Ergebnisse wurden laufend über LinkedIn und Instagram sowie über die offiziellen Kanäle der FH St. Pölten kommuniziert.

# 8 Eigene Projektwebsite

Während der Projektlaufzeit diente die netidee-Projektseite (https://www.netidee.at/haspi) als zentrale öffentliche Anlaufstelle; die Software-Entwicklung erfolgte zunächst in einem privaten GitHub-Repository zur internen Koordination und Versionsverwaltung. Mit Projektabschluss wurde das Repository unter https://github.com/fhstp/HaSPI öffentlich gestellt und dient gemeinsam mit der enthaltenen Dokumentation als technische Projektwebsite.

# 9 Geplante Aktivitäten nach netidee-Projektende

Nach Projektende wird die wissenschaftliche Verwertung der Ergebnisse fortgesetzt: Die Publikation zur Imitation-Learning-basierten Klassifikation befindet sich derzeit in Arbeit und soll bei einer einschlägigen Konferenz bzw. Fachzeitschrift eingereicht werden. Das GitHub-Repository wird weiter gepflegt; darüber hinaus ist geplant, die im Projekt entwickelten Methoden in Folgeforschung an der USTP – etwa in Abschlussarbeiten und weiterführenden Projektanträgen – aufzugreifen.

# 10 Anregungen für Weiterentwicklungen durch Dritte

Für Dritte ergeben sich mehrere Anknüpfungspunkte: Das Framework kann auf weitere deutschsprachige Plattformen und Korpora übertragen werden, um die Generalisierbarkeit des Imitation-Learning-Ansatzes zu prüfen. Die JAX-Implementierungen von IQ-Learn, SAC und xLSTM sind als eigenständige Komponenten für andere Sequenzklassifikations- und RL-Aufgaben nachnutzbar. Weiters bieten sich Studien zur Validierung mit realen Moderationsteams sowie die Integration des Frameworks in bestehende Moderationsworkflows (z. B. als Vorfilter mit menschlicher Letztentscheidung) an.