Fachhochschule
Nordwestschweiz

RLE Mini-Challenge
Deep Reinforcement Learning für Space Invaders

Nabil Mikhael

Windisch, 29. Dezember 2025

Hochschule für Technik Windisch T nabil.mikhael@students.fhnw.ch
www.fhnw.ch

Inhaltsverzeichnis

1 Einleitung

Dieser Bericht dokumentiert die Entwicklung und Evaluation verschiedener Deep Reinforcement Learning (DRL) Agenten für das klassische Atari-Spiel «Space Invaders». Das Ziel ist es, ausgehend von einer einfachen Baseline über einen initialen DQN-Ansatz systematisch vier Erweiterungen zu implementieren und deren Auswirkungen auf die Spielperformance zu analysieren.

Space Invaders bietet eine ideale Testumgebung für DRL-Algorithmen, da es sowohl reaktives Verhalten (Ausweichen von Projektilen) als auch strategische Entscheidungen (Positionierung, Timing) erfordert. Der Agent muss lernen, aus visuellen Eingaben (Spielframes) optimale Aktionen abzuleiten.

Die Arbeit ist wie folgt aufgebaut: Zunächst wird das experimentelle Setup beschrieben, gefolgt von der Evaluation einer Baseline. Anschliessend werden der initiale DQN-Ansatz sowie vier Erweiterungen (Double DQN, Dueling DQN, Hyperparameter-Tuning und Larger Network) vorgestellt und evaluiert. Abschliessend erfolgt ein Vergleich aller Ansätze sowie ein Fazit mit Ausblick.

Der vollständige Code ist auf GitLab [gitlab-link] und GitHub [github-link] verfügbar. Die Repositories enthalten alle Trainings-Skripte, die gespeicherten Modelle sowie die Evaluationsergebnisse. Die Lernkurven können mit TensorBoard visualisiert werden.

2 Experimentelles Setup

2.1 Trainingsumgebung

Alle Experimente wurden in der Gymnasium-Umgebung «ALE/SpaceInvaders-v5» durchgeführt. Die folgende Tabelle fasst die wichtigsten Umgebungsparameter zusammen:

Parameter	**Wert**
Environment	ALE/SpaceInvaders-v5
Observation Space	84 × 84 × 4 (Grayscale, Frame Stack)
Action Space	6 Aktionen (NOOP, FIRE, RIGHT, LEFT, RIGHTFIRE, LEFTFIRE)
Frame Skip	4 (via MaxAndSkipEnv)
Reward Clipping	[-1, 0, +1]

Der Observation Space (84×84×4) entsteht durch Verkleinerung der Original-Frames (210×160) auf 84×84 Pixel in Graustufen. Dies reduziert die Komplexität erheblich, ohne relevante Informationen zu verlieren. Durch das Frame Stacking von 4 Frames erhält das Modell temporale Information über die Bewegung von Objekten - einzelne Frames reichen nicht aus, um Bewegungsrichtungen von Gegnern oder Projektilen zu erkennen. Die Graustufen-Umwandlung reduziert den Rechenaufwand zusätzlich,

da Farbinformationen für die Entscheidungsfindung nicht relevant sind. Beim Frame Skip trifft der Agent nur alle vier Frames eine neue Entscheidung, wobei die gewählte Aktion für die übersprungenen Frames wiederholt wird. Dies beschleunigt das Training. Das Reward Clipping normalisiert alle Belohnungen auf −1, 0 oder +1 und trägt zur Stabilisierung des Lernprozesses bei.

2.2 Trainingsbudget

Für alle Agenten wurde ein einheitliches Trainingsbudget von 1’000’000 Timesteps festgelegt. Bei einem Frame Skip von 4 entspricht dies insgesamt 4’000’000 Spielframes. Dieses Budget stellt einen Kompromiss zwischen Trainingsdauer (ca. 3–4 Stunden pro Agent auf einem MacBook mit M2-Chip) und Konvergenz dar.

Intermediate Evaluation: Alle 100.000 Timesteps wurde eine kurze Bewertung durchgeführt. Dadurch kann der Lernfortschritt regelmässig beobachtet werden, ohne den Trainingsprozess stark zu unterbrechen.

Finale Evaluation: 100 Episoden pro Agent, um eine statistisch belastbare Aussage über die Performance treffen zu können. Eine grössere Anzahl reduziert die Varianz in der Messung und sorgt für zuverlässigere Vergleichswerte.

Modellspeicherung: Alle 250.000 Timesteps wurden Checkpoints gespeichert, um das Training bei Bedarf fortsetzen zu können.

2.3 Hardware

Alle Experimente wurden auf einem Apple MacBook mit M2-Chip (CPU) durchgeführt.

3 Evaluations-Konzept

Zur Sicherstellung einer fairen und konsistenten Bewertung aller Agenten wurde folgendes Evaluationsprotokoll verwendet:

- Finale Evaluation: 100 Episoden pro Agent mit $\varepsilon = 0.05$
- Intermediate Evaluation: alle 100’000 Timesteps
- Metriken: Mean, Standardabweichung, Minimum, Maximum, Median, Q25, Q75
- Identisches Preprocessing für alle Agenten (84×84 Grayscale, Frame Stack 4, Frame Skip 4, Reward Clipping)

Die Intermediate Evaluations ermöglichen die Analyse des Lernverlaufs über die Zeit hinweg. Durch das einheitliche Vorgehen wird sichergestellt, dass Leistungsunterschiede ausschliesslich auf algorithmische Änderungen zurückzuführen sind.

4 Baseline: Random Agent

Definition

Als Baseline dient ein Random Agent, der in jedem Timestep zufällig eine Aktion aus dem Action Space auswählt. Der Agent lernt nicht und repräsentiert somit die minimale erwartete Performance ohne Lernstrategie.

Evaluation

Metrik	**Wert**
Mean Reward	139.70
Std	95.12
Min	15
Max	430
Median	110
Q25	75
Q75	180

Der Random Agent erreicht einen durchschnittlichen Score von 139.70 Punkten. Die hohe Standardabweichung zeigt die starke Variabilität zufälligen Spielens. Erwartungsgemäss zeigt der Agent keine Leistungsverbesserung über die Zeit. Diese Baseline dient als Referenzpunkt für alle weiteren Ansätze.

5 Initialer Ansatz: DQN

5.1 Begründung der Wahl

Deep Q-Network (DQN) wurde als initialer Ansatz gewählt, da es sich bei Atari-Spielen bewährt hat (Mnih et al., 2015 – «Playing Atari with Deep Reinforcement Learning»). Der Algorithmus kombiniert Q-Learning mit tiefen neuronalen Netzen und nutzt Experience Replay sowie ein Target Network für stabiles Training. Die diskrete Aktionsstruktur von Space Invaders (6 Aktionen) passt sehr gut zum Q-Learning-Ansatz. Aufbauend auf diesem Ansatz werden später Erweiterungen wie Double DQN und Dueling DQN implementiert.

5.2 Netzwerkarchitektur

Die verwendete CNN-Architektur folgt dem klassischen DeepMind-Standard:

Layer	Filter/Neuronen	Kernel	Stride
Conv2d 1	32	8×8	4
Conv2d 2	64	4×4	2
Conv2d 3	64	3×3	1
Fully Connected	512	-	-
Output	6 (Aktionen)	-	-

Die Architektur ist gut geeignet für visuelle Atari-Umgebungen. Grosse Filter in der ersten Schicht extrahieren grobe räumliche Strukturen, während kleinere Filter in den folgenden Schichten feinere Details erfassen. Der Fully-Connected-Layer projiziert diese Merkmale auf Q-Werte für die sechs möglichen Aktionen.

5.3 Hyperparameter

Hyperparameter	Wert
Learning Rate	1e-4
Batch Size	32
Buffer Size	100.000
Gamma (Discount)	0.99
Epsilon Start → End	1.0 → 0.01
Exploration Fraction	0.1
Target Network Update	alle 1.000 Steps
Learning Start	50.000 Steps

Die Hyperparameter orientieren sich an der DQN-Literatur (Mnih et al., 2015). Die Learning Rate von 1×10^{-4} ermöglicht stabile Konvergenz ohne Überschwingungen. Mit einem Buffer Size von 100.000 werden genügend diverse Erfahrungen gespeichert. Der Learning Start bei 50.000 Steps stellt sicher, dass der Replay Buffer vor dem ersten Training bereits mit vielfältigen Erfahrungen gefüllt ist. Der Discount Factor $\gamma=0.99$ gewichtet langfristige Belohnungen stark, was für Space Invaders sinnvoll ist - Überleben ist wichtiger als kurzfristige Punkte. Das lineare Epsilon-Annealing von 1.0 auf 0.01 über 100.000 Steps ermöglicht zunächst ausgedehnte Exploration und später fokussierte Ausnutzung des gelernten Wissens.

5.4 Evaluation

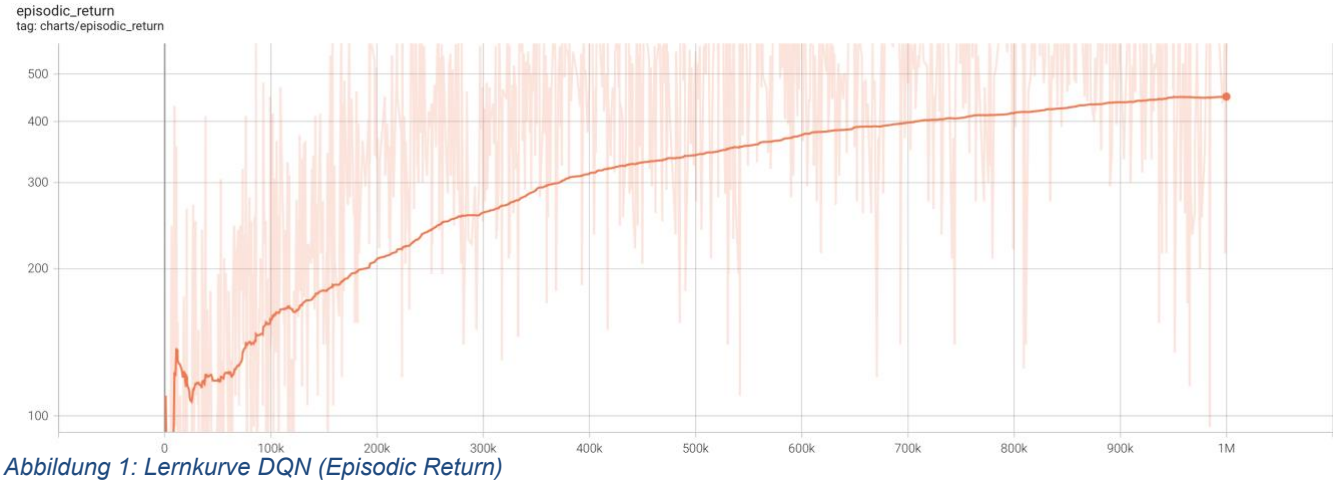


Abbildung 1: Lernkurve DQN (Episodic Return)

Die Lernkurve zeigt einen deutlichen Anstieg der Performance über das Training. In den ersten 100.000 Steps lernt der Agent grundlegende Strategien, danach steigt die Performance kontinuierlich an. Ab ca. 500.000 Steps stabilisiert sich der Agent auf einem Niveau von ~500 Punkten.

Metrik	**Wert**
Mean Reward	486.65
Std	185.77
Min	85
Max	1050
Median	485
Q25	370
Q75	600

Der DQN-Agent erreicht einen Mean Reward von 486.65, was einer Verbesserung von +248% gegenüber der Baseline (139.70) entspricht. Dies zeigt, dass der Agent erfolgreich gelernt hat, Space Invaders zu spielen. Die Lernkurve zeigt einen stetigen Anstieg der Performance während des Trainings.

6 Erweiterungen

6.1 Double DQN

Hypothese
Double DQN adressiert das Problem der Q-Wert-Überschätzung im klassischen DQN. Beim Standard-DQN wird dieselbe Netzwerk sowohl für die Aktionsauswahl als auch für die Bewertung verwendet, was zu systematischer Überschätzung führen kann. Double DQN entkoppelt diese beiden Schritte. Erwartet wird ein stabileres Training und möglicherweise bessere Performance.

Änderung
Die beste Aktion wird mit dem Online-Network ausgewählt, aber mit dem Target-Network bewertet: `Q_target = r + γ · Q_target(s', argmax_a Q_online(s', a))`.
Alle anderen Hyperparameter bleiben identisch zum initialen DQN.

Evaluation

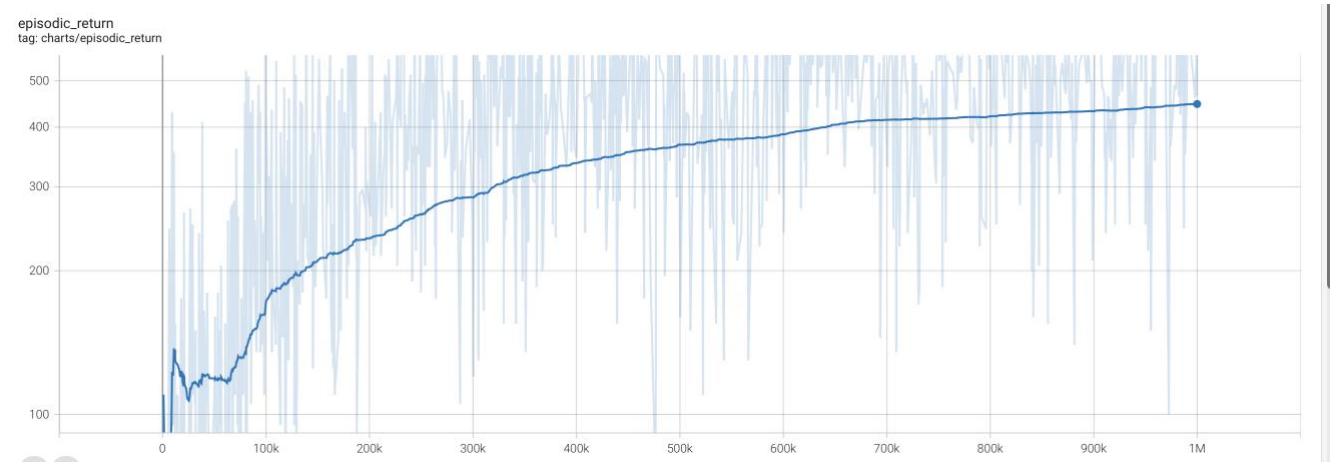


Abbildung 2: Lernkurve Double DQN (Episodic Return)

Der Agent weist auch in diesem Fall eine positive Tendenz auf. Im Vergleich zum DQN lässt sich jedoch keine klare Leistungssteigerung feststellen.

Metrik	**Wert**
Mean Reward	484.80
Std	175.12
Min	75
Max	1025
Median	452.5
Q25	373.75
Q75	600

Double DQN erreicht mit 484.80 praktisch identische Performance wie das Basis-DQN (486.65). Die leicht geringere Standardabweichung (175.12 vs. 185.77) deutet auf etwas stabileres Verhalten hin. Die Hypothese konnte nicht bestätigt werden - Q-Wert-Überschätzung scheint bei Space Invaders kein dominantes Problem zu sein, oder 1M Steps reichen nicht aus um den Unterschied sichtbar zu machen.

6.2 Dueling DQN

Hypothese

Die Dueling-Architektur trennt das Netzwerk in zwei Streams: einen Value-Stream V(s) und einen Advantage-Stream A(s,a). Dies soll eine bessere Generalisierung ermöglichen, da der Agent lernt, welche Zustände wertvoll sind, unabhängig von der gewählten Aktion.

Änderung

Nach den Conv-Layers wird das Netzwerk in zwei Streams aufgeteilt: V(s) und A(s,a). Die Q-Werte werden berechnet als: `Q(s,a) = V(s) + (A(s,a) - mean(A(s,·)).`
Alle anderen Hyperparameter bleiben identisch

Evaluation

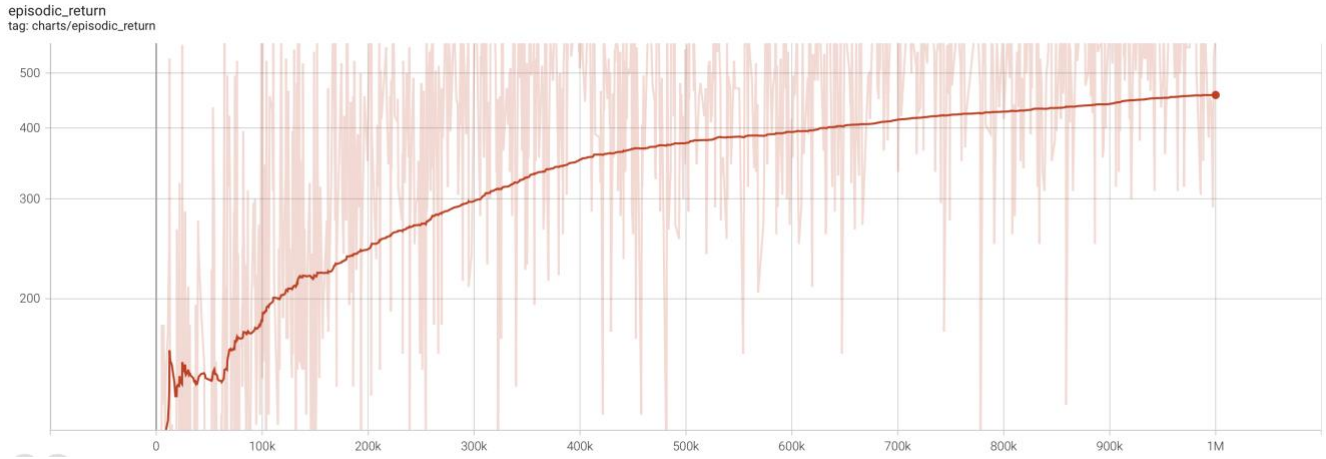


Abbildung 3: Lernkurve Dueling DQN (Episodic Return)

Auch hier ist eine klare positive Tendenz erkennbar, wobei sich der Score im gleichen Bereich wie bei den bisherigen Agenten bewegt.

Metrik	Wert
Mean Reward	448.40
Std	179.73
Min	50
Max	890
Median	480
Q25	330
Q75	571.25

Dueling DQN erreicht mit 448.40 eine etwas schlechtere Performance als das Basis-DQN (486.65). Die Hypothese konnte nicht bestätigt werden. Mögliche Gründe: Die Value/Advantage-Trennung bringt bei Space Invaders keinen Vorteil, da fast jede Aktion relevant ist (ständiges Ausweichen und Schiessen). Die Architektur könnte bei Spielen mit mehr «Wartezuständen» besser funktionieren.

6.3 Hyperparameter-Tuning

Hypothese

Durch Anpassung der Hyperparameter (höhere Learning Rate, grössere Batch Size, mehr Exploration) könnte die Performance verbessert werden.

Änderungen

Parameter	Original	Neu
Learning Rate	1e-4	2.5e-4
Batch Size	32	64
End Epsilon	0.01	0.05

Evaluation

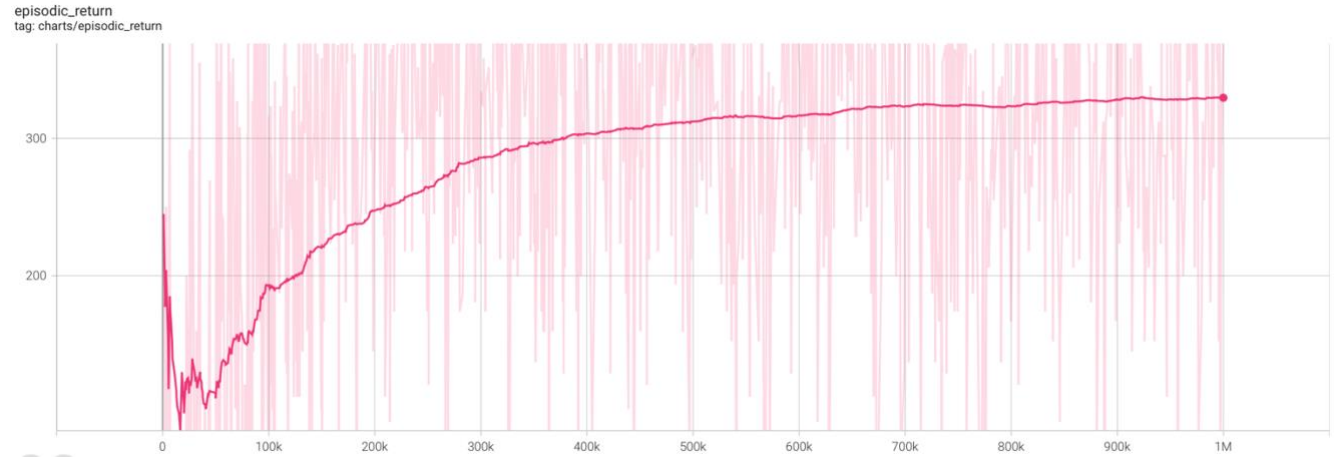


Abbildung 4: Lernkurve Hyperparameter Tuning (Episodic Return)

Auch hier ist eine klare positive Tendenz erkennbar, wobei sich der Score unter den bisherigen Agenten bewegt.

Metrik	**Wert**
Mean Reward	337.7
Std	148.55
Min	105
Max	870
Median	305
Q25	233.75
Q75	420

Das Hyperparameter-Tuning führte zu einer deutlichen Verschlechterung (337.70 vs. 486.65, **-31%**). Die Lernkurve zeigt zwar eine positive Tendenz, aber der Agent erreicht nur ~350 statt ~500 Punkte. Die höhere Learning Rate (2.5e-4) führte vermutlich zu instabilerem Training, und das höhere End-Epsilon (0.05 statt 0.01) bedeutet mehr zufällige Aktionen während der Exploitation-Phase. Dies zeigt, dass die Standard-Hyperparameter aus der DQN-Literatur bereits gut optimiert sind.

6.4 Larger Network

Hypothese

Ein grösseres Netzwerk mit mehr Parametern könnte komplexere Muster lernen und dadurch bessere Performance erzielen.

Änderungen

Layer	Original	Neu
Conv3 Filter	64	128
FC Layer	512	1024 → 512

Evaluation

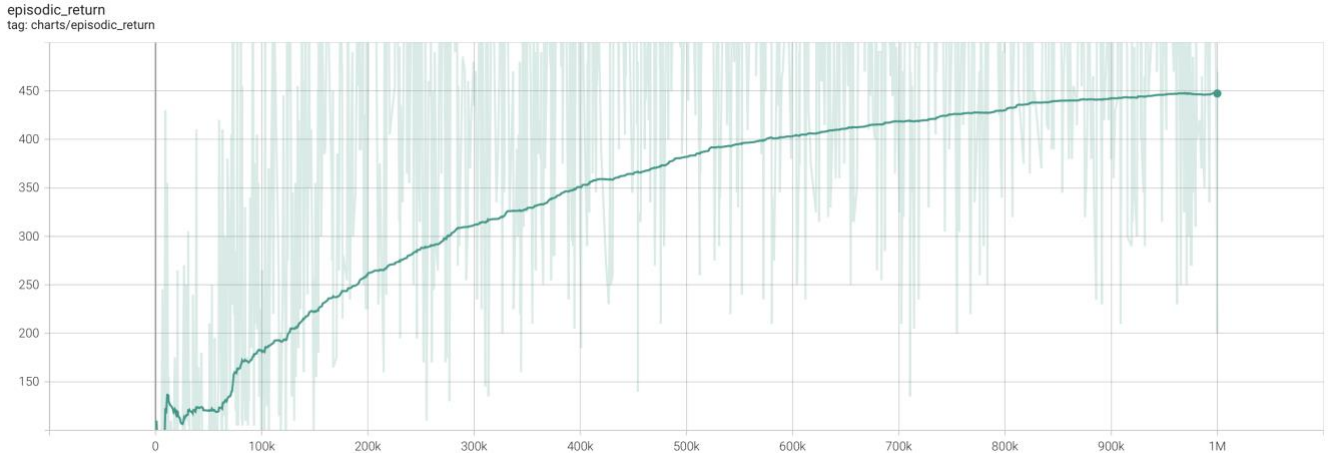


Abbildung 5: Lernkurve Larger Network (Episodic Return)

Auch hier ist eine klare positive Tendenz erkennbar, wobei sich der Score unter den bisherigen Agenten bewegt.

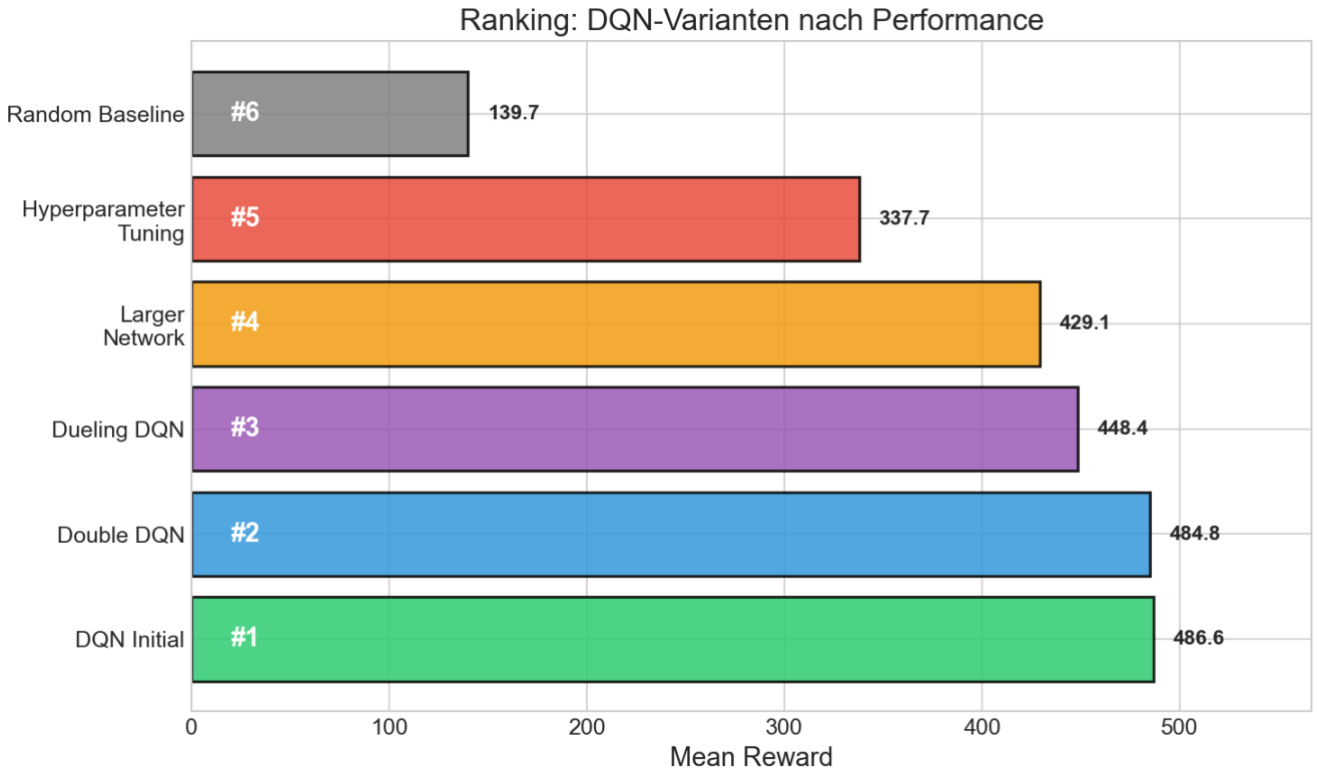
Metrik	Wert
Mean Reward	429.15
Std	143.71
Min	35
Max	870
Median	427.5
Q25	343.75
Q75	507.5

Das grössere Netzwerk erreicht 429.15 - etwas schlechter als das Basis-DQN (486.65, **-12%**). Die Hypothese konnte nicht bestätigt werden. Mögliche Gründe: Das grössere Netzwerk benötigt mehr Trainingszeit für Konvergenz (1M Steps reichen möglicherweise nicht), oder Space Invaders erfordert schlicht keine höhere Modellkapazität - das Standard-Netzwerk ist bereits ausreichend komplex für dieses Spiel.

7 Finaler Vergleich

Die folgende Tabelle fasst alle Ergebnisse zusammen:

Agent	Mean	Std	Min	Max	vs. Baseline
DQN Initial	486.6	185.8	85.0	1050.0	+248%
Double DQN	484.8	175.1	75.0	1025.0	+247%
Dueling DQN	448.4	179.7	50.0	890.0	+221%
Larger Network	429.1	143.7	35.0	870.0	+207%
Hyperparameter Tuning	337.7	148.5	105.0	870.0	+142%
Random Baseline	139.7	86.5	15	530	-



- **DQN Initial** erreicht mit einem Mean von 486.6 die beste Performance aller getesteten Ansätze (+248% vs. Baseline). Der hohe Max-Wert von 1050 zeigt das Potenzial des Agenten, während die Standardabweichung von 185.8 auf eine gewisse Variabilität hinweist.
- **Double DQN** liefert mit 484.8 praktisch identische Ergebnisse, zeigt aber mit einer niedrigeren Standardabweichung (175.1 vs. 185.8) etwas stabileres Verhalten. Die Bias-Korrektur durch die Entkopplung von Aktionswahl und -bewertung scheint bei Space Invaders keinen messbaren Vorteil zu bringen.
- **Dueling DQN** erreicht einen niedrigeren Mean (448.4) und zeigt mit dem niedrigsten Min-Wert (50) mehr katastrophale Episoden. Die Trennung in Value- und Advantage-Stream bringt bei Space Invaders keinen Vorteil, da fast jede Aktion relevant ist - es gibt wenig «Wartezuständen» wo nur der State-Value zählt.
- **Larger Network** zeigt mit 429.1 eine schwächere Performance, hat aber die niedrigste Standardabweichung (143.7) aller DQN-Varianten. Das grössere Netzwerk ist konsistenter,

erreicht aber weder die Spitzenwerte (Max: 870 vs. 1050) noch den Durchschnitt des Basis-DQN. Dies deutet darauf hin, dass 1M Steps nicht ausreichen, um das volle Potenzial des grösseren Netzwerks auszuschöpfen.
- **Hyperparameter-Tuning** schneidet am schlechtesten ab (337.7). Bemerkenswert ist der höhere Min-Wert (105 vs. 85), was auf weniger katastrophale Episoden hindeutet - aber zu Kosten der Spitzenleistung.
- **Kernbeobachtung:** Mehr Komplexität führt nicht automatisch zu besseren Ergebnissen. Die Standard-Hyperparameter aus der DQN-Literatur sind für Space Invaders bereits nahezu optimal.

8 Fazit

In dieser Arbeit wurden verschiedene DQN-Varianten für Space Invaders implementiert und evaluiert. Das Standard-DQN erreicht mit 486.6 Punkten die beste Performance (+248% vs. Baseline). Keine der vier Erweiterungen konnte eine Verbesserung erzielen.
Die konsistente Evaluationsmethodik (100 Episoden, identisches Preprocessing, gleiches Trainingsbudget von 1M Steps) ermöglichte einen fairen Vergleich aller Ansätze.

9 Ausblick

Basierend auf den Ergebnissen dieser Arbeit wären folgende Schritte sinnvoll:

- **Längeres Training:** Das Larger Network zeigte die niedrigste Varianz aber schwächere Performance - mit 5-10M Timesteps könnte es sein volles Potenzial entfalten.
- **Rainbow DQN:** Die Kombination mehrerer Erweiterungen (Double + Dueling + Prioritized Experience Replay + Noisy Nets) könnte synergistische Effekte zeigen, die bei einzelnen Erweiterungen nicht sichtbar waren.
- **Andere Spiele:** Dueling DQN könnte bei Spielen mit mehr «Wartezuständen» (besser funktionieren, da dort die Value/Advantage-Trennung relevanter ist.
- **Policy-basierte Methoden:** PPO oder A3C als Alternative zu value-basierten Methoden - möglicherweise besser geeignet für die kontinuierliche Aktion in Space Invaders.