

# Projektaufgaben für die PL

## „Künstliche Intelligenz“ WS18/19

---

Die Aufgaben werden jeweils in Einzelarbeit gelöst. Ihre Ideen und Erkenntnisse sollen aber vor allem auch in der Gruppe jeweils in einer Präsentation ausgetauscht werden. So hat jeder Teilnehmer auch etwas von der Arbeit seiner Kommilitonen und reflektiert die eigenen Ergebnisse besser. Wichtig für Ihren Vortrag ist, dass Sie einen eigenen Beitrag leisten, damit Ihre Kommilitonen auch einen breiten Überblick bekommen können und ein Bewusstsein für die Vielfältigkeit wissenschaftlicher Aufgabenstellungen entwickeln.

Die Prüfungsleistung (PL) besteht aus drei Einzelleistungen, die auf diesem Blatt beschrieben werden. Jede Leistung wird bewertet und die Abschlussnote errechnet sich (zu gleichen Gewichten) aus Ihren drei Leistungen. Allerdings ist jede Leistung zu erbringen um die PL zu bestehen.

### 1. Klassifikationslernen

Implementieren Sie eine Komponente eines Systems zum Lernen von Verkehrszeichen. Auf meiner website finden Sie einige Java-Dateien eines Lern-Framework mit Dummy-Klassen. Ersetzen Sie die Dummies so, dass ein lernendes System entsteht, das Bilder von Verkehrszeichen klassifizieren kann (Die Verwendung des Frameworks ist allerdings nicht zwingend vorgeschrieben). Trainings- bzw. Evaluierungsdaten finden Sie reichlich auf I:\Ringwelski\Verkehrszeichen.

In der Datei Farberkennung.java finden Sie Code-Fragmente zur Abbildung von RGB-Werten auf Farben. Dana Müller hat uns diesen Code zur Benutzung im Modul zur Verfügung gestellt.

- 1) Erzeugen von Merkmalsvektoren aus rohen Daten. Im Verzeichnis I:\Ringwelski\Verkehrszeichen finden Sie viele Fotos von Verkehrszeichen. Erzeugen Sie daraus Objekte, die Merkmalsvektoren und Konzepte enthalten und das Interface FeatureVector des Frameworks implementieren. Sammeln Sie die Daten in einer Liste oder Menge und speichern Sie sie in einer Datei ab.
- 2) Realisieren Sie ein Lernverfahren als Implementierung des Interface Learner
  - a) Mit einem Entscheidungsbaum ODER
  - b) Mit einem einschichtigen künstlichen neuronalen Netz
- 3) Realisieren Sie ein Evaluationsverfahren für Ihr Lernverfahren. Dazu ist als Testtreiber die Klasse Evaluator zu vervollständigen, die Merkmalsvektoren aus der Datei einliest, in Trainingsmenge und Testmenge aufteilt, den Algorithmus trainiert und dann evaluiert. Messen Sie den durchschnittlichen Lernerfolg und sein Konfidenzintervall. Variieren Sie dabei willkürliche Festlegungen wie z.B. die Reihenfolge der Trainingsbeispiele um deren Einfluss auf das Ergebnis zu minimieren.
- 4) Formulieren Sie eine wissenschaftliche Fragestellung, die sich durch Experimente mit ihrem code beantworten lassen sollte und stimmen diese spätestens in KW 43 mit dem Dozenten ab. Beispiele: „welchen Einfluss hat die Reihenfolge der Merkmale auf den Lernerfolg?“ „Wie groß muss die Trainingsmenge sein?“ „Wie groß ist der Lernerfolg meines Verfahrens?“...

- 5) Präsentieren Sie eine Antwort auf Ihre Fragestellung in einer schriftlichen Ausarbeitung. Erläutern Sie darin die o.g. Frage, Ihren Versuchsaufbau und die Ergebnisse. Leiten Sie aus den Ergebnissen auch eine allgemeingültige Schlussfolgerung ab.
- 6) Präsentieren Sie Ihre Antwort auf Ihre Fragestellung im Plenum, indem Sie eine Hypothese durch Lernerfolg, Variationen des Testaufbaus, Konfidenzintervalle oder andere geeignete Analysen in einem Vortrag (20 Minuten) belegen.

**Abgabetermin** für die schriftliche Ausarbeitung und Beginn der Vorträge ist **KW 45**.

## 2. Lernverfahren

Realisieren Sie ein weiteres Lernverfahren in dem „Verkehrszeichenprojekt“ aus Aufgabe 1. Setzen Sie dazu aber jetzt eines der folgenden Lernverfahren ein:

- Backpropagation: Lernen mit Mehrschichtigen künstliches neuronalen Netzen
- K-Nearest-Neighbor Klassifikation: Lernen mit Abstandsmaßen
- Naive-Bayes Klassifikator: Minimieren der Fehlerwahrscheinlichkeit
- C4.5: Fortgeschrittenes Entscheidungsbaumverfahren

Evaluieren Sie dann diesen neuen Klassifikator genauso wie ihren eignen in Aufgabe 1 im Framework und vergleichen Sie die Ergebnisse. Dazu müssen Sie den Algorithmus in das framework integrieren, es ist aber nicht erforderlich, den Algorithmus selbst zu implementieren, sie können auf Lösungen aus dem Internet zurückgreifen.

Erstellen Sie eine **Präsentation als Lehrveranstaltung** (45 min) für Ihre Kommilitonen, in der sie das eingesetzte Verfahren erklären, es mit den bereits gesehenen Verfahren vergleichen und am Verkehrszeichen-Beispiel evaluieren. Achten Sie darauf, dass Ihre Kommilitonen in Ihrer Präsentation lernen können, wie das von Ihnen verwendete Verfahren im Allgemeinen und im Projekt funktioniert. (Eine schriftliche Ausarbeitung ist nicht gefordert)

Die Präsentationen finden ab **KW 49** statt.

### 3. Suchverfahren

Implementieren Sie ein Suchverfahren (z.B. Simulated Annealing, einen genetischen Algorithmus oder auch BestFirstSearch/BnB) zum Lösen des Graph Coloring Problems (GCP). Verwenden Sie die Probleminstanzen aus [sites.google.com/site/graphcoloring/vertex-coloring](https://sites.google.com/site/graphcoloring/vertex-coloring), die auf meiner website bereits in einzelnen Dateien zur Verfügung stehen: gcpProblems.zip. Auf der genannten website finden Sie zu jeder Probleminstanz auch die chromatische Zahl, die das optimale Ergebnis darstellt. Versuchen Sie mit Einsatz angemessener Rechenzeit an diesen Wert möglichst nah heran zu kommen.

Beim Graph Coloring Problem werden die Knoten eines Graphen so eingefärbt, dass zwei Knoten, die durch eine Kante verbunden sind, unterschiedliche Farben haben. Dabei sollen möglichst wenig unterschiedliche Farben benutzt werden. Die Anzahl der benötigten Farben wird „chromatische Zahl“ genannt. Die chromatische Zahl ist bei diesem Problem zu minimieren. Ein bekannter Spezialfall ist das Färbeproblem für Landkarten, bei dem benachbarte Länder unterschiedlich gefärbt werden müssen. Für Landkarten reichen dabei drei verschiedene Farben aus. Hier ist die sog. chromatische Zahl also 3. In allgemeinen Graphen kann das komplexer sein, so dass oft auch zweistellige chromatische Zahlen herauskommen (vgl. Tabelle auf der o.g. website).

Bearbeiten Sie folgenden Aufgaben:

- a) Implementieren Sie ein Suchverfahren Ihrer Wahl zur Lösung der o.g. GCP -Instanzen.
- b) Testen (also implementieren Sie wiederholbare Tests) Sie die Korrektheit Ihres Verfahrens? Sind die gefundenen Lösungen wirklich zulässig? Wenn nicht, welchen Einfluss hat das auf die Qualitätsbewertung? Die Optimalität braucht natürlich nicht geprüft zu werden, lediglich die o.g. Randbedingungen.
- c) Evaluieren Sie Ihren Solver bzgl Laufzeit und Optimierungsergebnis (die chromatische Zahl) anhand unterschiedlicher Benchmark-Instanzen. Achten Sie darauf, Tests von nicht-deterministischen Algorithmen hinreichend oft durchzuführen und Konfidenzintervalle sowohl zur Laufzeit als auch zur Qualität der gefundenen Lösung anzugeben.
- d) Beschreiben Sie den von Ihnen eingesetzten Algorithmus, Ihren Versuchsaufbau zur Evaluierung und die Ergebnisse in einer kurzen schriftlichen Ausarbeitung.
- e) Präsentieren Sie Ihre Lösung sowie dessen Evaluierung in einem Vortrag (20 min). Gehen Sie beim Algorithmus eher auf Ihre speziellen Einstellungen und Modifikationen ein, Sie brauchen nicht die Algorithmen aus der Vorlesung zu wiederholen.

**Abgabetermin** für die schriftliche Ausarbeitung und Beginn der Präsentationen ist **KW03**.