



TECHNISCHE UNIVERSITÄT
CHEMNITZ

Kolloquium zur Bachelorarbeit
Interaktive, grafische Steuerung des Growing Neural Gas
TU Chemnitz, Fakultät für Informatik

Kolloquium zur Bachelorarbeit

Interaktive, grafische Steuerung des Growing Neural Gas

Tobias Gall

TU Chemnitz, Fakultät für Informatik

20. Oktober 2015



TECHNISCHE UNIVERSITÄT
CHEMNITZ

Gliederung

1. Zielsetzung

2. Growing Neural Gas

3. Anforderungen

4. Implementierung

Zielsetzung

Das Ziel dieser Arbeit:

- ▶ Erarbeitung eines Konzeptes zur interaktiven, grafikbasierten Steuerung des Growing Neural Gas (GNG)
- ▶ Prototypische Umsetzung dieses Konzeptes

Der Zweck:

- ▶ Darstellung und Steuerung des GNG
- ▶ Nachvollziehbarkeit des Trainings des GNG

Growing Neural Gas

- ▶ künstliches neuronales Netz
- ▶ 1995 von Bernd Fritzke entwickelt
- ▶ lernt selbstständig seine Struktur

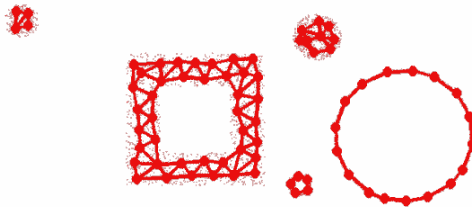


Abbildung : Simulation Growing Neural Gas

Anforderungen

Übersicht

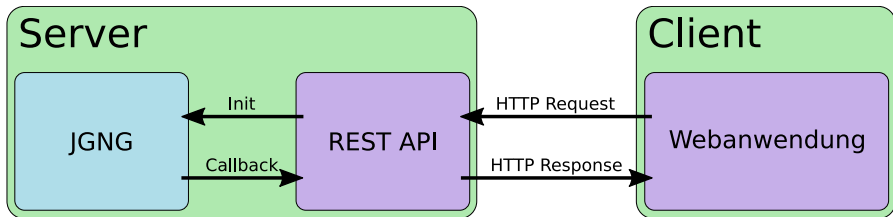


Abbildung : Übersicht Komponenten

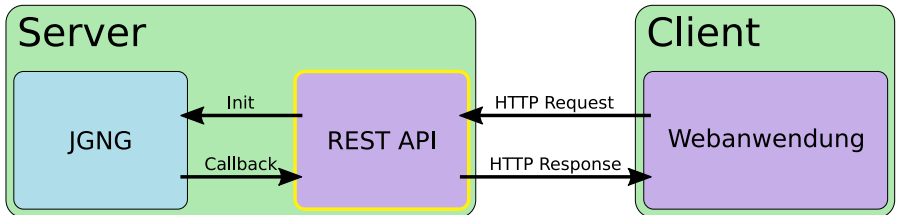
Anforderungen an die API

- ▶ Auswahl des Algorithmus, zum Trainieren des neuronalen Netzes
- ▶ Einstellung aller Parameter des Algorithmus
- ▶ Auswahl eines Datensatzes, zum Trainieren des neuronalen Netzes
- ▶ Festlegung der Anzahl an Durchläufen für eine Trainingseinheit
- ▶ Bereitstellung der topologischen Daten des neuronalen Netzes, die der Algorithmus generiert hat
- ▶ Paralleles Trainieren mehrerer neuronaler Netze

Anforderungen an die Webanwendung

- ▶ Nutzung der Funktionalität der API
- ▶ Übersichtliche Darstellung des Trainingsprozesses des neuronalen Netzes
- ▶ Benutzerfreundliche Möglichkeit zur Parametrisierung des Algorithmus

REST API



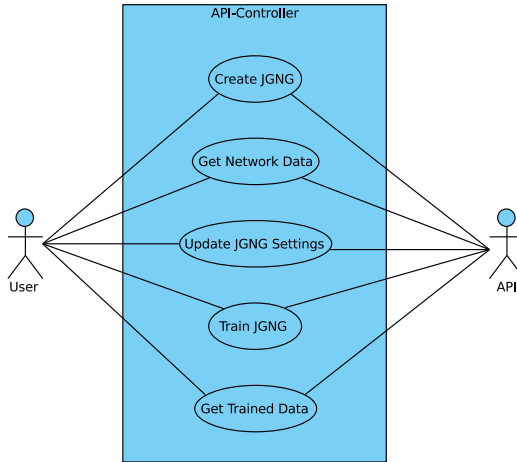
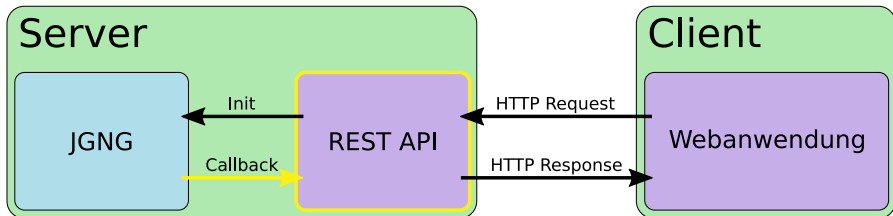


Abbildung : Usecasediagramm API

Aspekte der Implementierung

- ▶ Initialisierung des JGNG
- ▶ Implementierung des Delegationsinterfaces(Callback)
- ▶ Datenstruktur für Neuronen und Kanten
- ▶ Verarbeiten der Nutzeranfragen und Bereitstellung der Daten

JGNG Callback



- ▶ Callback wird als Interface vom Algorithmus bereitgestellt und von der REST API implementiert
- ▶ Er empfängt alle Änderungen am neuronalem Netz
- ▶ **Problem:** NeuronenID sind nur Listindices
→ Mapping auf eindeutige IDs durch atomare Zählvariable
- ▶ Speicherung der Neuronen und Kanten

Datenstruktur

- Speicherung der Neuronen und Kanten als Objekte
→ Verknüpfung mit zusätzlichen Attributen
- Zusammenfassung der Neuronen und Kanten in zwei verschiedenen Objekten

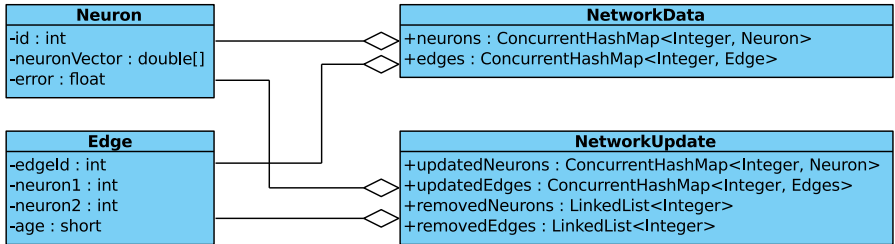
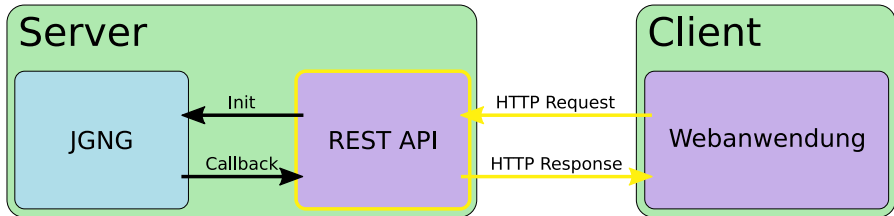
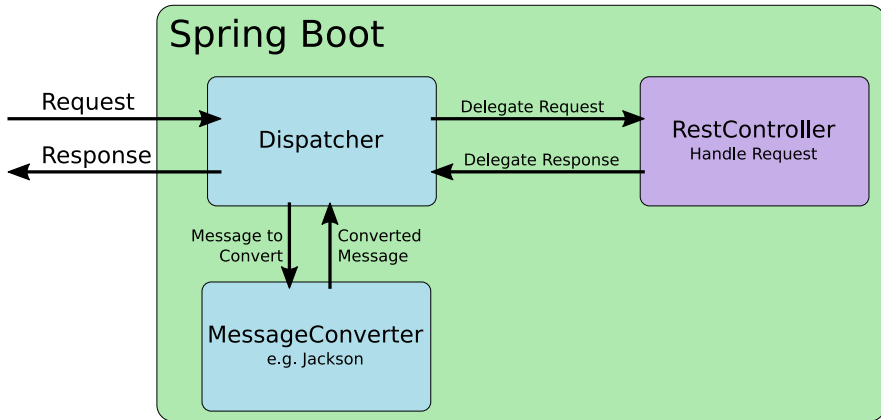


Abbildung : Klassendiagramm Datenstruktur

Verwaltung der Anfragen



Einsatz des Spring Boot Framework



REST Controller

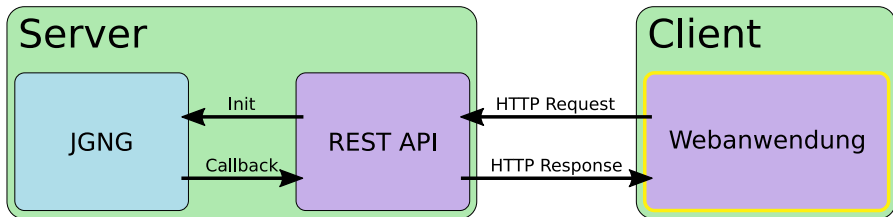
Der REST Controller übernimmt folgende Aufgaben:

- ▶ Initialisierung des JGNG
 - ▶ Kapselung des JGNG in einem Wrapper mit eindeutiger ID
→ mehrere JGNG parallel verfügbar
 - ▶ Übergabe eines Callback und Datenhaltung
- ▶ Entgegennehmen der Einstellungen für das JGNG
- ▶ Zurückgeben des neuronalen Netzes
- ▶ Zurückgeben einer Trainingseinheit

Optional:

- ▶ Auflisten aller neuronalen Netze
- ▶ Löschen eines neuronalen Netzes

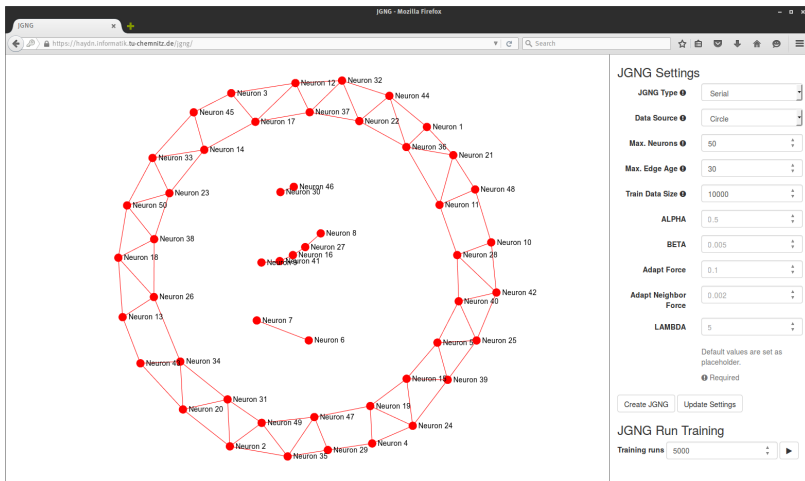
Webanwendung



Layout und Umsetzung

- ▶ fixiertes Layout in der Aufteilung von 75% (Graph) zu 25% (Einstellungen) der Breite des Fensters
- ▶ Verwendung von **Bootstrap** (CSS) für die grundlegende Gestaltung
- ▶ Verwendung von **jQuery** (JS) zur Kommunikation mit der API mittels AJAX und JSON
- ▶ Verwendung des **Sigma** (JS) für die Darstellung des Graphen

Ergebnis



Vielen Dank.