technische universität
dortmund

fakultät für informatik

Master-Thesis

Convolutional Neural Networks auf Graphrepräsentationen von Bildern

Matthias Fey

18. Mai 2017

Gutachter:

Prof. Dr. Heinrich Müller

M.Sc. Jan Eric Lenssen

Lehrstuhl Informatik VII
Graphische Systeme
TU Dortmund

Inhaltsverzeichnis

Mathematische Notationen

$\|\cdot\|_2$

1 Einleitung

Homepage[1]
Convolutional Neural Networks (CNNs) CNN $\mathbb{N}$
„wdawd“

1.1 Problemstellung

1.2 Aufbau der Arbeit

[1] https://github.com/rusty1s/embedded_gcnn

2 Grundlagen

2.1 Mathematische Notationen

2.2 Graphentheorie

2.3 Convolutional Neural Networks

3 Graphrepräsentationen von Bildern

3.1 Gitter

3.2 Superpixel

3.2.1 Verfahren

SLIC [1]
Simple Linear Iterative Clustering (SLIC)

Quickshift [15]

Weitere Verfahren [5]

3.2.2 Adjazenzmatrixbestimmung

3.2.3 Merkmalsextraktion

4 Räumliches Lernen auf Graphen

4.1 Räumliche Graphentheorie

Färbung von Knoten awdawd

Isomorphie und kanonische Ordnung awdawd

4.2 Räumliche Faltung

Knotenauswahl awdawd

Nachbarschaftsgruppierung awdawd

Normalisierung awdawd

4.3 Erweiterung auf ebene Graphen

4.4 Netzarchitektur

5 Spektrales Lernen auf Graphen

Das spektrale Lernen auf Graphen bzw. die Formulierung eines spektralen Faltungsoperator auf Graphen basiert auf der spektralen Graphentheorie, d.h. der Betrachtung des Spektrums eines Graphen definiert über dessen Eigenwerte. Merkmale auf den Knoten eines Graphen können über das Spektrum analog zur Fourier-Transformation in dessen Frequenzraum zerlegt und wieder retransformiert werden. Diese Transformation erlaubt damit die fundamentale Formulierung eines Faltungsoperators in der spektralen Domäne des Graphen. Da der so definierte spektrale Faltungsoperator insbesondere rotationsinvariant ist, wird dieser im Verlauf des Kapitels für den Kontext von Graphen im euklidschen Raum modifiziert.

Durch die spektrale Formulierung kann weiterhin ein effizientes Pooling auf Graphen formuliert werden, welches uns erlaubt, Netzarchitekturen auf Graphen völlig analog zu klassischen CNNs auf zweidimensionalen Bildern zu generieren.

5.1 Spektrale Graphentheorie

Die spektrale Graphentheorie beschäftigt sich mit der Konstruktion, Analyse und Manipulation von Graphen. Sie beweist sich dabei als besonders nützlich in Anwendungsgebieten wie der Charakterisierung von Expandergraphen, dem Graphenzeichnen oder dem spektralen Clustering (vgl. [13]). Weiterhin hat die spektrale Graphentheorie z.B. auch Anwendungsgebiete in der Chemie, bei der die Eigenwerte des Spektrums des Graphen mit der Stabilität von Molekülen assoziert werden (vgl. [2]).

Es zeigt sich, dass die Eigenwerte des Spektrums eines Graphen eng mit den Eigenschaften eines Graphen verwandt sind. Als spektrale Graphentheorie versteht man damit insbesondere die Studie über die gemeinsamen Beziehungen dieser beiden Bereiche. Dieses Kapitel gibt eine Einführung in die wichtigsten Definitionen und Intuitionen der spektralen Graphentheorie, die es uns schlussendlich erlauben, die spektrale Faltung auf Graphen zu formulieren.

5.1.1 Eigenwerte und Eigenvektoren reell symmetrischer Matrizen

Das *Eigenwertproblem* einer Matrix $\mathbf{M} \in \mathbb{R}^{N \times N}$ ist definiert als $\mathbf{Mu} = \lambda \mathbf{u}$, wobei $\mathbf{u} \in \mathbb{R}^N$, $\mathbf{u} \neq \mathbf{0}$ *Eigenvektor* und $\lambda \in \mathbb{R}$ der entsprechende *Eigenwert* zu $\mathbf{u}$ genannt werden. Ein Eigenvektor $\mathbf{u}$ beschreibt damit einen Vektor, dessen Richtung durch die Abbildung $\mathbf{Mu}$ nicht verändert, sondern lediglich um den Faktor λ skaliert wird. Zu einem Eigenwert λ gibt es unendlich viele (skalierte) Eigenvektoren $\mathbf{u}$. Wir definieren den Eigenvektor $\mathbf{u}$ eines Eigenwertes λ daher eindeutig über die Bedingung $\|\mathbf{u}\|_2 = 1$. Sei $\mathbf{M}$ weiterhin symmetrisch, d.h. $\mathbf{M} = \mathbf{M}^\top$. Dann gilt für zwei unterschiedliche Eigenvektoren $\mathbf{u}_1$ und $\mathbf{u}_2$, dass diese orthogonal zueinander stehen, d.h. $\mathbf{u}_1 \perp \mathbf{u}_2$, und $\mathbf{M}$ hat genau N reelle Eigenwerte mit $\{\lambda_i\}_{i=1}^N$. Wir definieren demnach zu $\mathbf{M}$ die orthogonale *Eigenvektormatrix* $\mathbf{U} \coloneqq [\mathbf{u}_1, \ldots, \mathbf{u}_n] \in \mathbb{R}^{N \times N}$, d.h. $\mathbf{U}\mathbf{U}^\top = \mathbf{I}$, und dessen Eigenwertdiagonalmatrix $\mathbf{\Lambda} \coloneqq \operatorname{diag}\left([\lambda_1, \ldots, \lambda_N]^\top\right)$, d.h. $\mathbf{\Lambda}_{ii} = \lambda_i$. Dann gilt $\mathbf{MU} = \mathbf{U\Lambda}$ und insbesondere ist $\mathbf{M}$ diagonalisierbar über

$$\mathbf{M} = \mathbf{MUU}^\top = \mathbf{U\Lambda U}^\top.$$

Weiterhin gilt für die k-te Potenz von $\mathbf{M}$, $k \in \mathbb{N}$,

$$\mathbf{M}^k = \left(\mathbf{U\Lambda U}^\top\right)^k = \mathbf{U\Lambda}^k\mathbf{U}^\top. \tag{5.1}$$

Dieser Zusammenhang lässt sich verdeutlichen, wenn die Potenz ausgeschrieben wird:

$$\left(\mathbf{U\Lambda U}^\top\right)^k = \mathbf{U\Lambda U}^\top\mathbf{U\Lambda U}^\top \prod_{i=1}^{k-2} \mathbf{U\Lambda U}^\top = \mathbf{U\Lambda}^2\mathbf{U}^\top \prod_{i=1}^{k-2} \mathbf{U\Lambda U}^\top = \mathbf{U\Lambda}^k\mathbf{U}^\top.$$

Falls $\mathbf{M}$ weiterhin *schwach diagonaldominant* ist, d.h.

$$\sum_{\substack{j=1 \\ j \neq i}}^{N} |\mathbf{M}_{ij}| \leq |\mathbf{M}|_{ii}, \tag{5.2}$$

und weiterhin $\mathbf{M}_{ii} \geq 0$ für alle $i \in \{1, \ldots, N\}$, dann ist $\mathbf{M}$ *positiv semidefinit*, d.h. $\mathbf{x}^\top\mathbf{Mx} \geq 0$ für alle $\mathbf{x} \in \mathbb{R}^N$. Eigenwerte symmetrischer positiv semidefiniter Matrizen $\lambda_i \in \mathbb{R}^+$ sind positiv reell und es lässt sich folglich auf diesen eine Ordnung definieren mit $0 \leq \lambda_1 \leq \cdots \leq \lambda_N \coloneqq \lambda_{\max}$.

5.1.2 Laplace-Matrix

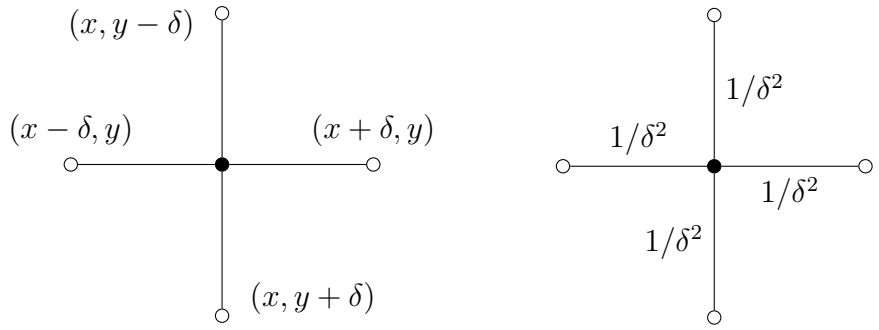
Die Laplace-Matrix ist in der spektralen Graphentheorie eine Matrix, die die Beziehungen der Knoten und Kanten eines beliebigen Graphen $\mathcal{G}$ in einer generalisierten und normalisierten Form beschreibt. Viele der Eigenschaften von $\mathcal{G}$ können durch die Eigenwerte ihrer Laplace-Matrix beschrieben werden, wohingegen dies z.B. für die Eigenwerte der Adjazenzmatrix $\mathbf{A}$ von $\mathcal{G}$ nur bedingt zutrifft und insbesondere nicht verallgemeinbar für beliebge Graphen ist [2]. Dies ist vorallem dem Fakt geschuldet, dass die Eigenwerte der Laplace-Matrix konsistent sind mit den Eigenwerten des Laplace-Beltrami Operators ∇^2 in der spektralen Geometrie [2]. Die Laplace-Matrix ist damit ein geeignetes Mittel zur Betrachtung und Analyse eines Graphen.
Für einen schleifenlosen, ungerichteteten, gewichtet oder ungewichteten Graphen $\mathcal{G}$ und dessen Adjazenzmatrix $\mathbf{A}$ mit Gradmatrix $\mathbf{D}$ ist die *kombinatorische Laplace-Matrix* $\mathbf{L}$ definiert als $\mathbf{L} \coloneqq \mathbf{D} - \mathbf{A}$ [2]. Die *normalisierte Laplace-Matrix* $\tilde{\mathbf{L}}$ ist definiert als $\tilde{\mathbf{L}} \coloneqq \mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2}$ mit der Konvention, dass $\mathbf{D}_{ii}^{-1/2} = 0$ für isolierte Knoten $v_i \in \mathcal{V}$ in $\mathcal{G}$, d.h. $\mathbf{D}_{ii} = 0$ [2]. Daraus ergibt sich die elementweise Definition

$$\tilde{\mathbf{L}}_{ij} \coloneqq \begin{cases} 1, & \text{wenn } i = j, \\ -\frac{w(v_i,v_j)}{\sqrt{d(v_i)d(v_j)}}, & \text{wenn } v_i \sim v_j, \\ 0, & \text{sonst.} \end{cases}$$

Für verbundene Graphen kann $\tilde{\mathbf{L}}$ vereinfacht werden zu $\tilde{\mathbf{L}} \coloneqq \mathbf{I} - \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$ [2]. Jeder Eintrag auf der Diagonalen der normalisierten Laplace-Matrix ist folglich Eins. $\tilde{\mathbf{L}}$ ist damit normalisiert auf den (gewichteten) Grad zweier adjazenter Knoten v_i und v_j. Es ist anzumerken, dass $\mathbf{L}$ und insbesondere $\tilde{\mathbf{L}}$ symmetrisch sind, wohingegen eine Normalisierung der Form $\mathbf{D}^{-1}\mathbf{L}$ dies in der Regel nicht wäre [12]. $\mathbf{L}$ und $\tilde{\mathbf{L}}$ sind desweiteren keine ähnlichen Matrizen, insbesondere sind ihre Eigenvektoren verschieden. Die Nutzung von $\mathbf{L}$ oder $\tilde{\mathbf{L}}$ ist damit abhängig von dem Problem, welches man betrachtet [6]. Wir schreiben $\mathcal{L}$ wenn die Wahl der Laplace-Matrix, ob $\mathbf{L}$ oder $\tilde{\mathbf{L}}$, für die weitere Berechnung irrelevant ist.

Interpretation Sei $f\colon \mathcal{V} \to \mathbb{R}$ bzw. $\mathbf{f} \in \mathbb{R}^N$ mit $f(v_i) = \mathbf{f}_i$ eine Funktion bzw. ein Signal auf den Knoten eines Graphen $\mathcal{G}$. Dann kann für die kombinatorische Laplace-Matrix $\mathbf{L}$ verifiziert werden, dass sie die Gleichung

$$(\mathbf{L}\mathbf{f})_i = \sum_{i \sim j} w(v_i, v_j)(\mathbf{f}_i - \mathbf{f}_j)$$



(a) Gitterrepräsentation (b) Graphrepräsentation

Abbildung 5.1: Illustration des 5-Punkte-Sterns in zwei Dimensionen mit gleicher Approximation des ∇^2 Operators (bei umgekehrtem Vorzeichen), einmal mit der 5-Punkte-Stern Approximation auf regulären Gittern (a) und einmal mit der kombinatorischen Laplace-Matrix $\mathbf{L}$ auf Graphen (b).

erfüllt [6]. Sei $\mathcal{G}$ nun ein Graph, der aus einem (unendlichen) zweidimensionalen regulärem Gitter entstanden ist, d.h. jeder Knoten v_i besitzt genau vier achsenparallele rechtwinklige Nachbarn mit gleichen Kantengewichten $1/\delta^2$, wobei $\delta \in \mathbb{R}$ eine beliebige Konstante. Zur einfacheren Veranschaulichung benutzen wir dabei für die Signalstärke $\mathbf{f}_i$ eines Knoten v_i an Position (x, y) die Indexnotation $\mathbf{f}_{x,y}$. Dann beschreibt

$$(\mathbf{Lf})_{x,y} = \frac{4\mathbf{f}_{x,y} - \mathbf{f}_{x+1,y} - \mathbf{f}_{x-1,y} - \mathbf{f}_{x,y+1} - \mathbf{f}_{x,y-1}}{h^2}$$

die *5-Punkte-Stern* Approximation $\nabla^2 f$ (bei umgekehrtem Vorzeichen) definiert auf den Punkten $\{(x, y), (x+\delta, y), (x-\delta, y), (x, \delta+h), (x, y-\delta)\}$ [6] (vgl. Abbildung 5.1). Ähnlich zu einem regulären Gitter lässt sich ein Graph $\mathcal{G}$ auch über beliebig viele Abtastpunkte einer differenzierbaren Mannigfaltigkeit konstruieren. Es zeigt sich, dass mit steigender Abtastdichte und geeigneter Wahl der Kantengewichte die normalisierte Laplace-Matrix $\tilde{\mathbf{L}}$ zu dem kontinuierlichem Laplace-Beltrami Operator ∇^2 konvergiert [6]. Damit kann $\tilde{\mathbf{L}}$ als die diskrete Analogie des ∇^2 Operators auf Graphen verstanden werden. Der Laplace-Beltrami Operator $\nabla^2 f(p)$ misst dabei, in wie weit sich eine Funktion f an einem Punkt p von dem Durchschnitt aller Funktionspunkte um einen kleinen Bereich um p unterscheidet. Die Laplace-Matrix operiert dabei völlig analog, in dem sie misst, wie sehr sich eine (diskrete) Funktion um einen Knoten im Vergleich zu seinen Nachbarknoten unterscheidet.

Eigenwerte und Eigenvektoren von $\mathcal{L}$ helfen uns dabei, die lineare Transformation einer Funktion $\mathbf{f}$ (mehrfach) angewendet auf $\mathcal{L}$ besser zu verstehen. Wir können

dafür $\mathbf{f}$ als Linearkombination der Eigenbasis $\sum_i c_i \mathbf{u}_i$ schreiben und erhalten

$$\mathcal{L}^k \mathbf{f} = \sum_i c_i \mathcal{L}^k \mathbf{u}_i = \sum_i c_i \lambda_i^k \mathbf{u}_i.$$

Somit können Eigenschaften von $\mathcal{L}$ und damit des Graphen selber durch dessen Eigenwerte und Eigenvektoren beschrieben werden.

Eigenschaften $\mathcal{L} \in \mathbb{R}^{N \times N}$ ist eine reell symmetrische, positiv semidefinite Matrix [2]. Folglich besitzt $\mathcal{L}$ nach Kapitel 5.1.1 genau N positiv reelle Eigenwerte $\{\lambda_i\}_{i=1}^N$ mit Ordnung $0 \leq \lambda_1 \leq \cdots \leq \lambda_N$ und N korrespondierenden orthogonalen Eigenvektoren $\{\mathbf{u}_i\}_{i=1}^N$.
Die kombinatorische Laplace-Matrix $\mathbf{L}$ ist nach (5.2) weiterhin schwach diagonaldominant. Insbesondere summiert sich jede Reihen- und Spaltensumme von $\mathbf{L}$ zu Null auf, d.h. $\sum_{j=1}^N \mathbf{L}_{ij} = \sum_{j=1}^N \mathbf{L}_{ji} = 0$. Daraus folgt unmittelbar, dass $\lambda_1 = 0$, da $\mathbf{u}_1 = 1/\sqrt{N}[1, \ldots, 1]^\top \in \mathbb{R}^N$ Eigenvektor von $\mathbf{L}$ mit $\mathbf{L}\mathbf{u}_1 = \mathbf{0}$ [13]. $\tilde{\mathbf{L}}$ hingegen ist nicht zwingend schwach diagonaldominant. Es lässt sich jedoch zeigen, dass auch für $\tilde{\mathbf{L}}$ gilt, dass $\lambda_1 = 0$ [2].
Eine der interessantesten Eigenschaften eines Graphen ist dessen Konnektivität. Die Laplace-Matrix $\mathcal{L}$ bzw. dessen Eigenwerte stellen ein geeignetes Mittel zur Untersuchung dieser Eigenschaft dar. So gilt z.B. für einen verbundenen Graphen $\mathcal{G}$, dass $\lambda_2 > 0$. Falls $\lambda_i = 0$ und $\lambda_{i+1} \neq 0$, dann besitzt $\mathcal{G}$ genau i verbundene Komponenten [2]. Damit ist die Anzahl der Null-Eigenwerte äquivalent zu der Anzahl an Komponenten, die ein Graph besitzt. Für $\tilde{\mathbf{L}}$ lässt sich weiterhin zeigen, dass $\lambda_{\max} \leq 2$ eine obere Schranke ihrer Eigenwerte ist [2].
Aus der Laplace-Matrix können ebenso Rückschlüsse über die kürzeste Pfaddistanz zweier Knoten gewonnen werden. So gilt für $\mathcal{L}^k$ mit $k \in \mathbb{N}$, dass $\mathcal{L}_{ij}^k = 0$ genau dann, wenn $s(v_i, v_j) > k$ [6]. Damit beschreibt $\mathcal{L}_i^k$ bildlich gesprochen die Menge an Knoten, die maximal k Kanten von i entfernt liegen.

5.2 Spektraler Faltungsoperator

Sei $\mathbf{f} \in \mathbb{R}^N$ ein Signal auf den Knoten eines Graphen $\mathcal{G}$, welches abhängig von der Struktur des Graphen weiter verarbeitet werden soll. Es ist jedoch nicht selbstverständlich, wie recht einfache, dennoch fundamentale Signalverarbeitungsprozesse wie Translation oder Filterung und die daraus entstehende Faltung in der Domäne des Graphen definiert werden können [13]. So kann z.B. ein analoges Signal $f(t)$ mittels

$f(t-3)$ um 3 nach rechts verschoben werden. Es ist hingegen völlig unklar, was es bedeutet, ein Graphsignal auf den Knoten um 3 nach rechts zu bewegen (vgl. [13]). Die spektrale Graphentheorie bietet uns dafür einen geeigneten Weg, in dem Eingabesignale in das Spektrum des Graphen zerlegt bzw. abgebildet, modifiziert und wieder retransformiert werden können.

5.2.1 Graph-Fourier-Transformation

Das Spektrum eines Graphen $\mathcal{G}$ bilden die Eigenwerte $\{\lambda_i\}_{i=1}^N$ der Laplace-Matrix $\mathcal{L}$ von $\mathcal{G}$. Diese werden deshalb auch oft als die *Frequenzen* von $\mathcal{G}$ betitelt. In der spektralen Domäne können wir ein Eingabeignal $\mathbf{f}$ über $\mathcal{G}$ dann analog wie ein zeitdiskretes Abtastsignal in der Fourier-Domäne behandeln.

Klassische Fourier-Transformation Die Fourier-Transformation $\hat{f}$ einer Funktion $f(t)$ ist definiert als [13]

$$\hat{f}(\omega) \coloneqq \left\langle f, e^{2\pi i\omega t} \right\rangle = \int_{\mathbb{R}} f(t) e^{-2\pi i\omega t} \, \mathrm{d}t.$$

Die komplexen Exponentiale $e^{2\pi i\omega t}$ beschreiben dabei die Eigenfunktionen des eindimensionalen Laplace-Beltrami Operators [13]

$$-\nabla^2 e^{2\pi i\omega t} = -\frac{\partial^2}{\partial t^2} e^{2\pi i\omega t} = (2\pi\omega)^2 e^{2\pi i\omega t}. \tag{5.3}$$

$\hat{f}$ kann damit als die Ausdehnung von f in Bezug auf die Eigenfunktionen des Laplace-Beltrami Operators ∇^2 verstanden werden [6].

Analog lässt sich die *Graph-Fourier-Transformation* einer Funktion $f\colon \mathcal{V} \to \mathbb{R}$ bzw. $\mathbf{f} \in \mathbb{R}^N$ auf den Knoten eines Graphen $\mathcal{G}$ als Ausdehnung von f in Bezug auf die Eigenvektoren $\{\mathbf{u}_i\}_{i=1}^N$ der Laplace-Matrix $\mathcal{L}$ definieren [13]:

$$\hat{f}(\lambda_i) \coloneqq \langle \mathbf{f}, \mathbf{u}_i \rangle \qquad \text{bzw.} \qquad \hat{\mathbf{f}} \coloneqq \mathbf{U}^\top \mathbf{f}. \tag{5.4}$$

Die inverse Graph-Fourier-Transformation ergibt sich dann als [13]

$$f(v_i) = \sum_{j=1}^{N} \hat{f}(\lambda_j)(\mathbf{u}_j)_i \qquad \text{bzw.} \qquad \mathbf{f} = \mathbf{U}\hat{\mathbf{f}}. \tag{5.5}$$

In der klassischen Fourier-Analyse sind für die Eigenwerte $\left\{(2\pi\omega)^2\right\}_{\omega\in\mathbb{R}}$ in (5.3) na-

he bei Null die korrespondieren Eigenfunktionen kleine, weich schwingende Funktionen, wohingegen für größere Eigenwerte bzw. Frequenzen die Eigenfunktionen sehr schnell und zügig anfangen zu oszillieren. Bei der Graph-Fourier-Transformation ist dies ähnlich. So ist für $\mathbf{L}$ der erste Eigenvektor $\mathbf{u}_1 = 1/\sqrt{N}[1, \ldots, 1]^\top$ zum Eigenwert $\lambda_1 = 0$ konstant und an jedem Knoten gleich. Generell zeigt sich, dass die Eigenvektoren geringer Frequenzen nur geringfügig im Graph variieren, wohingegen Eigenvektoren größerer Eigenwerte immer unähnlicher werden (vgl. [13]).

Die Graph-Fourier-Transformation (5.4) und ihre Inverse (5.5) bieten uns eine Möglichkeit ein Signal in zwei unterschiedlichen Domänen zu repräsentieren, nämlich der Knotendomäne, d.h. das unveränderte Signal auf der Knotenmenge $f(v_i)$, und der spektralen Domäne, d.h. das transformierte Signal in das Spektrum des Graphen $\hat{f}(\lambda_i)$. Diese Transformation erlaubt uns die Formulierung fundamentaler Signalverarbeitungsoperationen.

5.2.2 Spektrale Filterung

In der Signalverarbeitung versteht man unter der Frequenzfilterung die Transformation eines Eingabesignals in die Fourier-Domäne und der verstärkenden oder dämpfenden Veränderung der Amplituden der Frequenzkomponenten. Formal betrachtet ergibt dies

$$\hat{f}_{\text{out}}(\omega) \coloneqq \hat{f}_{\text{in}}(\omega)\hat{g}(\omega) \tag{5.6}$$

mit dem Filter $\hat{g}\colon \mathbb{R} \to \mathbb{R}$. Shuman u. a. zeigen, dass die Filterung in der Fourier-Domäne äquivalent zu einer Faltung in der Zeitdomäne ist, d.h.

$$(f_{\text{in}} \star g)(t) \coloneqq \int_{\mathbb{R}} f_{\text{in}}(\tau)g(t-\tau)\,\mathrm{d}\tau = f_{\text{out}}(t). \tag{5.7}$$

Wir können die Filterung der Frequenzen in der Fourier-Domäne analog zu (5.6) für die spektrale Domäne auf Graphen über

$$\hat{f}_{\text{out}}(\lambda_i) \coloneqq \hat{f}_{\text{in}}(\lambda_i)\hat{g}(\lambda_i) \qquad \text{bzw.} \qquad \hat{\mathbf{f}}_{\text{out}} \coloneqq \hat{\mathbf{f}}_{\text{in}} \odot \hat{\mathbf{g}}$$

beschreiben, wobei $\odot$ das elementweise Hadamard-Produkt ist [13]. $\hat{\mathbf{g}} \in \mathbb{R}^N$ ist damit ein *nicht-parametrischer* Filter, d.h. ein Filter, dessen Werte für alle Frequenzen $\{\lambda_i\}_{i=1}^N$ frei wählbar sind [3]. Daraus ergibt sich analog zu (5.7) der *spektrale Faltungsoperator* auf Graphen in der Knotendomäne mit Hilfe der Graph-Fourier-

Transformation (5.4) und ihrer Inversen (5.5) als [3, 13]

$$\mathbf{f}_{\text{in}} \star \hat{\mathbf{g}} \coloneqq \mathbf{U}\Big(\mathbf{U}^\top \mathbf{f}_{\text{in}} \odot \hat{\mathbf{g}}\Big) = \mathbf{f}_{\text{out}}. \tag{5.8}$$

5.2.3 Polynomielle Approximation

Es zeigt sich, dass die Benutzung des spektralen Faltungsoperators in (5.8) im Kontext eines CNNs auf Graphen mehrere Schwächen aufweist. So ist z.B. die Auswertung von $\mathbf{f}_{\text{out}} \star \hat{\mathbf{g}}$ extrem berechnungsintensiv ist, denn die Multiplikation mit der dichtbesetzten Eigenvektormatrix $\mathbf{U}$ liegt in $\mathcal{O}(N^2)$. Zudem muss $\mathbf{U}$ zuerst bestimmt werden — ein kostspieliger Aufwand für Graphen mit möglicherweise weit mehr als hundert Knoten [9]. Desweiteren führt ein Filter $\hat{\mathbf{g}} \in \mathbb{R}^N$ der Größe N zu einem Lernaufwand in $\mathcal{O}(N)$, d.h. der Dimensionalität der Eingabedaten [3]. Ebenso kann $\hat{\mathbf{g}}$ so nicht für das Lernen auf Graphen mit variierendem N verwendet werden. Um die oben genannten Schwächen zu umgehen kann $\hat{g}(\lambda_i)$ über ein Polynom

$$\hat{g}(\lambda_i) \approx \sum_{k=0}^{K} c_k \lambda_i^k =: \hat{g}'(\lambda) \tag{5.9}$$

vom Grad K mit Koeffizienten $c_0, \dots, c_K \in \mathbb{R}$ approximiert werden [3, 6]. Die Filtergröße von $\hat{g}'$ sinkt somit auf einen konstanten Faktor K mit Lernaufwand $\mathcal{O}(K)$, dem gleichen Aufwand klassischer zweidimensionaler CNNs [3]. $\mathbf{f}_{\text{in}} \star \hat{\mathbf{g}}$ ergibt dann nach (5.1), (5.8) und (5.9) approximiert durch [3]

$$\mathbf{f}_{\text{in}} \star \hat{\mathbf{g}} \approx \sum_{k=0}^{K} c_k \mathbf{U}\mathbf{\Lambda}^k\mathbf{U}^\top \mathbf{f}_{\text{in}} = \sum_{k=0}^{K} c_k \mathcal{L}^k \mathbf{f}_{\text{in}}. \tag{5.10}$$

Insbesondere ist die spektrale Faltung damit nicht mehr abhängig von der Berechnung der Eigenwerte bzw. Eigenvektoren von $\mathcal{L}$. Mittels Kapitel 5.1.2 kann $\mathbf{f}_{\text{in}} \star \hat{\mathbf{g}}$ in der Knotendomäne nun als eine *lokaliserte lineare Transformation* interpretiert werden. So sammelt ein Summand $\mathcal{L}^k \mathbf{f}_{\text{in}}$ des spektralen Filters an einem Knoten v genau die Signale von Knoten auf, die maximal k Kanten von v entfernt liegen [6]. Eine Faltung über $f_{\text{in}}(v_i)$ wird damit als Linearkombination

$$f_{\text{out}}(v_i) \approx b_{ii} f_{\text{in}}(v_i) + \sum_{v_j \in \mathcal{N}_K(v_i)} b_{ij} f_{\text{in}}(v_j)$$

über dessen k-lokalisierte Nachbarschaft $\mathcal{N}_K(v_i) \coloneqq \Big\{v_j \in \mathcal{V}, i \neq j \mid \mathcal{L}_{ij}^K > 0\Big\}$ beschrieben mit $b_{ij} \coloneqq \sum_{k=0}^{K} c_k \mathcal{L}_{ij}^k$ [13].

Tschebyschow-Polynome Obwohl der Aufwand zur Bestimmung von $\mathbf{f}_{\text{in}} \star \hat{\mathbf{g}}$ durch die polynomielle Approximation und insbesondere durch den Wegfall der Berechnung der Eigenvektormatrix $\mathbf{U}$ deutlich reduziert wurde, ist diese immer noch recht teuer aufgrund der Berechnung der K-ten Potenz von $\mathcal{L}$. So ist $\mathcal{L}$ zwar eine dünnbesetzte Matrix mit $|\mathcal{E}|+N \ll N^2$, $N \leq |\mathcal{E}|$, Einträgen, $\mathcal{L}^K$ ist dies jedoch zwangsläufig nicht. Eine Lösung zu diesem Problem ist die Benutzung eines Polynoms mit einer rekursiven Formulierung. Ein rekursives Polynom, dass dafür üblicherweise genutzt wird, ist das *Tschebyschow-Polynom*, da sich dieses zusätzlich durch einen sehr günstigen Fehlerverlauf auszeichnet (vgl. [6]). Tschebyschow-Polynome bezeichnen eine Menge von Polynomen $T_k(x)\colon \mathbb{R} \to \mathbb{R}$ mit dem rekursiven Zusammenhang

$$T_k(x) = 2x\, T_{k-1}(x) - T_{k-2}(x)$$

mit $T_0(x) = 1$ und $T_1(x) = x$ [6]. Ein Tschebyschow-Polynom T_k ist ein Polynom k-ten Grades und gilt im Intervall $[-1, 1]$ als numerisch stabil, d.h. $T_k(x) \in [-1, 1]$ für $x \in [-1, 1]$ [6]. Die spektrale Faltung mittels Tschebyschow-Polynomen ergibt sich dann nach (5.10) als

$$\mathbf{f}_{\text{in}} \star \hat{\mathbf{g}} \approx \sum_{k=0}^{K} c_k \mathbf{U} T_k\Big(\tilde{\mathbf{\Lambda}}\Big)\mathbf{U}^\top \mathbf{f}_{\text{in}}, \tag{5.11}$$

wobei $c_0, \ldots, c_K \in \mathbb{R}$ nun Koeffizienten der Tschebyschow-Polynome $T_k\Big(\tilde{\mathbf{\Lambda}}\Big) \in \mathbb{R}^{N \times N}$ angewendet auf die skalierten Eigenwerte $\tilde{\mathbf{\Lambda}} \coloneqq 2\mathbf{\Lambda}/\lambda_{\max} - \mathbf{I}$ mit $\tilde{\mathbf{\Lambda}} \in [-1, 1]^{N \times N}$ [3]. Dann kann (5.11) mit $\tilde{\mathcal{L}} \coloneqq \mathbf{U}\tilde{\mathbf{\Lambda}}\mathbf{U}^\top = 2\mathcal{L}/\lambda_{\max} - \mathbf{I}$ weiter vereinfacht werden zu [3]

$$\mathbf{f}_{\text{in}} \star \hat{\mathbf{g}} \approx \sum_{k=0}^{K} c_k T_k\Big(\tilde{\mathcal{L}}\Big)\mathbf{f}_{\text{in}}. \tag{5.12}$$

Für $\mathcal{L}$ kann $\lambda_{\max} \leq 2$ auf dessen obere Schranke, d.h. $\lambda_{\max} \coloneqq 2$, gesetzt werden um die Berechnung von $\lambda_{\max}$ zu vermeiden ohne die numerische Stabilität von T_k im Intervall $[-1, 1]$ zu verletzen.

Der rekursive Zusammenhang von T_k hilft uns dabei, $\mathbf{f}_{\text{in}} \star \hat{\mathbf{g}}$ effizient zu bestimmen. Berechne dafür $\bar{\mathbf{f}}_k \coloneqq T_k\Big(\tilde{\mathcal{L}}\Big)\mathbf{f}_{\text{in}} \in \mathbb{R}^N$ für alle $k \in \{0, \ldots, K\}$ rekursiv mit $\bar{\mathbf{f}}_0 = \mathbf{f}_{\text{in}}$, $\bar{\mathbf{f}}_1 = \tilde{\mathcal{L}}\,\mathbf{f}_{\text{in}}$ und $\bar{\mathbf{f}}_k = 2\tilde{\mathcal{L}}\,\bar{\mathbf{f}}_{k-1} - \bar{\mathbf{f}}_{k-2}$. Dann ergibt sich $\mathbf{f}_{\text{out}} = \Big[\bar{\mathbf{f}}_0, \bar{\mathbf{f}}_1, \ldots, \bar{\mathbf{f}}_K\Big]\mathbf{c} \in \mathbb{R}^N$ mit $\mathbf{c} \coloneqq [c_0, c_1, \ldots, c_K]^\top \in \mathbb{R}^{K+1}$ (vgl. [6]). $\mathbf{f}_{\text{out}}$ lässt sich damit über $K + 1$ Multiplikationen einer dünnbesetzen Matrix mit einem Vektor und einer abschließenden Vektormultplikation beschreiben. Mit $N \leq |\mathcal{E}|$ ergibt dies eine finale Laufzeit der spektralen Faltung von $\mathcal{O}(K|\mathcal{E}|)$ [3].

Implementierung Für gewöhnlich besteht eine CNN-Schicht nicht nur aus einem, sondern aus M_{in} vielen Signalen bzw. Merkmalen pro Knoten mit jeweils unterschiedlichen Filtern bzw. Gewichten pro Ein- und Ausgabekarte. In klassichen zweidimensionalen CNNs werden diese Merkmale auf M_{out} viele Merkmale abgebildet, in dem für jede Ausgabekarte über jede Eingabekarte gefaltet und dessen Ergebnisse sukzessive aufsummiert werden (vgl. Kapitel 2.3). Modellieren wir diesen Fall für den spektralen Faltungsoperator, dann erhalten wir rekursiv für eine Merkmalsmatrix $\mathbf{F}_{\text{in}} \in \mathbb{R}^{N \times M_{\text{in}}}$ gefaltet auf eine Merkmalsmatrix $\mathbf{F}_{\text{out}} \in \mathbb{R}^{N \times M_{\text{out}}}$ über den N Knoten eines Graphen $\mathcal{G}$ mittels des Filtertensors $\mathbf{W} \in \mathbb{R}^{K+1 \times M_{\text{in}} \times M_{\text{out}}}$

$$\mathbf{F}_{\text{out}} \coloneqq \sum_{k=0}^{K} \bar{\mathbf{F}}_k,$$

wobei $\bar{\mathbf{F}}_k = \Big(2\tilde{\mathcal{L}}\,\bar{\mathbf{F}}_{k-1} - \bar{\mathbf{F}}_{k-2}\Big)\mathbf{W}_k \in \mathbb{R}^{N \times M_{\text{out}}}$ mit $\bar{\mathbf{F}}_0 = \mathbf{F}_{\text{in}}\mathbf{W}_0$ und $\bar{\mathbf{F}}_1 = \tilde{\mathcal{L}}\,\mathbf{F}_{\text{in}}\mathbf{W}_1$.

5.3 Graph Convolutional Networks

Kipf und Welling motivieren einen weiteren Ansatz zur Faltung auf Graphen, genannt *Graph Convolutional Network (GCN)*, der auf der Methodik des spekralen Faltungsoperators aus Kapitel 5.2 aufbaut und dabei wie eine „differenzierbare und parametriesierte Generalisierung des eindimensionalen Weisfeiler-Lehman Algorithmus auf Graphen“ fungiert [9].

5.3.1 Faltungsoperator

Sei $\mathbf{f}_{\text{in}} \star \hat{\mathbf{g}} \approx \sum_{k=0}^{K} c_k T_k\Big(\tilde{\mathcal{L}}\Big)\mathbf{f}_{\text{in}}$ der in (5.12) definierte spektrale Faltungsoperator mit $K = 1$. Dann ist $\mathbf{f}_{\text{in}} \star \hat{\mathbf{g}}$ eine lineare Funktion bzgl. $\mathcal{L}$ und damit eine lineare Funktion auf dem Spektrum des Graphen [9]. Mit $K = 1$ betrachtet der spektrale Faltungsoperator nur noch die lokale Nachbarschaft eines jeden Knoten (vgl. 5.2.3). Es ist anzumerken, dass dies in der Regel keinen Nachteil darstellt. So hat es sich bei gegenwärtigen „State-of-the-Art“-CNNs auf Bildern ebenfalls eingebürgert, nur noch über minimale 3×3 Receptive-Fields zu falten und stattdessen Merkmale weit entfernterer Knoten über die mehrfache Aneinanderreihung der Faltungsschichten mittels tieferer Netze zu gewinnen (vgl. [7, 9, 14]). Unter dieser Restriktion vereinfacht sich $\mathbf{f}_{\text{in}} \star \hat{\mathbf{g}}$ zu

$$\mathbf{f}_{\text{in}} \star \hat{\mathbf{g}} \approx c_0\,\mathbf{f}_{\text{in}} + c_1\Big(\frac{2}{\lambda_{\max}}\mathcal{L} - \mathbf{I}\Big)\mathbf{f}_{\text{in}} \tag{5.13}$$

mit zwei freien Parametern c_0 und c_1 [9]. Für $\tilde{\mathbf{L}}$ auf einem verbundenen Graphen $\mathcal{G}$ gilt dann nach (5.13) weiter

$$\mathbf{f}_{\text{in}} \star \hat{\mathbf{g}} \approx c_0\, \mathbf{f}_{\text{in}} + c_1\Big(\tilde{\mathbf{L}} - \mathbf{I}\Big)\mathbf{f}_{\text{in}} = c_0\, \mathbf{f}_{\text{in}} - c_1 \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}\mathbf{f}_{\text{in}}, \tag{5.14}$$

wobei $\lambda_{\max} \coloneqq 2$ auf dessen oberste Schranke gesetzt wird [9]. Um die Gefahr des Overfittings und die Anzahl an Berechnungen pro Schicht weiter zu beschränken, reduziert sich (5.14) mit einem einzigen Parameter $c \coloneqq c_0$ mit $c = -c_1$ zu [9]

$$\mathbf{f}_{\text{in}} \star \hat{\mathbf{g}} \approx c\Big(\mathbf{I} + \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}\Big)\mathbf{f}_{\text{in}}.$$

Die skalierten Eigenwerte von $\tilde{\mathbf{\Lambda}}$ liegen auf Grund der Addition mit $\mathbf{I}$ nun im Intervall $[0, 2]$ (vgl. [9]). Demnach können wiederholte Anwendungen des Faltungsoperators zu „numerischen Instabilitäten und folglich zu explodierenden oder verschwindenen Gradienten“ führen [9]. Kipf und Welling führen zur Behebung dieses Problems die folgende Renormalisierung durch: $\mathbf{I}+\mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2} \to \tilde{\mathbf{D}}^{-1/2}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-1/2}$ mit $\tilde{\mathbf{A}} \coloneqq \mathbf{A}+\mathbf{I}$ und $\tilde{\mathbf{D}}_{ii} \coloneqq \sum_{j=1}^{N} \tilde{\mathbf{A}}_{ij}$. Der entgültige Faltungsoperator des GCNs ergibt sich dann als

$$\mathbf{f}_{\text{in}} \star \hat{\mathbf{g}} \approx c\, \tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}\mathbf{f}_{\text{in}}$$

auf einem einzigen freien Parameter $c \in \mathbb{R}$.

Implementierung Die Faltung des GCNs auf Merkmalsmatrizen lässt sich analog zur Tensorimplementierung des spektralen Faltungsoperators in Kapitel 5.2.3 beschreiben, mit dem Unterschied, dass wir aufgrund der Festlegung von $K = 1$ keinen Gewichtstensor, sondern lediglich eine Gewichtsmatrix $\mathbf{W} \in \mathbb{R}^{M_{\text{in}} \times M_{\text{out}}}$ nutzen. Die Faltung einer Eingabemerkmalsmatrix $\mathbf{F}_{\text{in}} \in \mathbb{R}^{N \times M_{\text{in}}}$ auf eine Ausgabemerkmalsmatrix $\mathbf{F}_{\text{out}} \in \mathbb{R}^{N \times M_{\text{out}}}$ ergibt sich dann als

$$\mathbf{F}_{\text{out}} \coloneqq \tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}\,\mathbf{F}_{\text{in}}\mathbf{W}$$

mit Faltungsaufwand $\mathcal{O}(M_{\text{in}}M_{\text{out}}|\mathcal{E}|)$, weil $\tilde{\mathbf{A}}\mathbf{F}_{\text{in}}$ effizient mit der Multiplikation einer dünnbesetzten mit einer dichtbesetzten Matrix implementiert werden kann [9].

Weisfeiler-Lehman Analogie

5.3.2 Erweiterung auf ebene Graphen

B-Spline-Kurven

Faltungsoperator

5.4 Pooling auf Graphen

5.4.1 Graphvergröberung

Clustering von Knoten

5.4.2 Erweiterung auf ebene Graphen

5.5 Netzarchitektur

6 Evaluation

6.1 Versuchsaufbau

6.1.1 Datensätze

MNIST [11]

Cifar-10 [10]

Pascal VOC [4]

6.1.2 Metriken

6.1.3 Parameterwahl

Vorstellung aller Parameter Superpixelalgorithmen Parameterwahl

6.2 Merkmalsselektion

6.3 Ergebnisse

Vergleich mit anderen Implementierungen

6.4 Laufzeitanalyse

Vergleich mit anderen Implementierungen

6.5 Diskussion

7 Ausblick

Weitere Anwendungsgebiete

Entfernung irrelevanter Knoten nicht nur bei MNIST auch bei pascal voc slic? (z.B. Regelmäßigkeiten erkennen)

Augmentierung von Graphen

Spatial-Pyramid-Pooling

Attention-Algorithmus

A Weitere Informationen

Abbildungsverzeichnis

Algorithmenverzeichnis

Literaturverzeichnis

[1] Achanta, Radhakrishna; Shaji, Appu; Smith, Kevin; Lucchi, Aurelien; Fua, Pascal; Susstrunk, Sabine: SLIC Superpixels Compared to State-of-the-Art Superpixel Methods. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2012), S. 2274–2282

[2] Chung, Fan .R.K.: *Spectral Graph Theory*. American Mathematical Society, 1997

[3] Defferrard, Michaël; Bresson, Xavier; Vandergheynst, Pierre: Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering. In: *Advances in Neural Information Processing Systems 29*. Curran Associates, Inc., 2016, S. 3844–3852

[4] Everingham, Mark; Eslami, S.M. Ali; Van Gool, Luc; Williams, Christoper K. I.; Winn, John; Zisserman, Andrew: The Pascal Visual Object Classes Challenge: A Retrospective. In: *International Journal of Computer Vision* (2015), S. 98–136

[5] Felzenszwalb, Pedro F.; Huttenlocher, Daniel P.: Efficient Graph-Based Image Segmentation. In: *International Journal of Computer Vision* (2004), S. 167–181

[6] Hammond, David K.; Vandergheynst, Pierre; Gribonval, Réne: Wavelets on Graphs via Spectral Graph Theory. In: *Applied and Computational Harmonic Analysis* (2011), S. 129–150

[7] He, Kaiming; Zhang, Xiangyu; Ren, Shaoqing; Sun, Jian: Deep Residual Learning for Image Recognition. In: *Computer Vision and Pattern Recognition* (2016), S. 83–98

[8] Huszár, Ferenc: *How Powerful are Graph Convolutions?* `http://www.inference.vc/how-powerful-are-graph-convolutions-review-of-kipf-welling-2016-2/`. 2016

[9] Kipf, Thomas N.; Welling, Max: Semi-Supervised Classification with Graph Convolutional Networks. In: *Computing Research Repository* (2016)

[10] Krizhevsky, Alex: *Learning Multiple Layers of Features from Tiny Images*, Department of Computer Science, University of Toronto, Diplomarbeit, 2009

[11] LeCun, Yann; Cortes, Corinna; Burges, Christopher J.C.: The MNIST Database of Handwritten Digits. (2010)

[12] Reuter, Martin; Biasotti, Silvia; Giorgi, Daniela; Patanè, Guiseppe; Spagnuolo, Michela: Discrete Laplace-Beltrami Operators for Shape Analysis and Segmentation. In: *Computers & Graphics* (2009), S. 381–390

[13] Shuman, David I.; Narang, Sunil. K.; Frossard, Pascal; Ortega, Antonio; Vandergheynst, Pierre: The Emerging Field of Signal Processing on Graphs: Extending High-Dimensional Data Analysis to Networks and Other Irregular Domains. In: *IEEE Signal Processing Magazine* (2013), S. 83–98

[14] Simonyan, Karen; Zisserman, Andrew: Very Deep Convolutional Networks for Large-Scale Image Recognition. In: *Computing Research Repository* (2014)

[15] Vedaldi, Andrea; Soatto, Stefano: Quick Shift and Kernel Methods for Mode Seeking. In: *European Conference on Computer Vision*, 2008, S. 705–718

Eidesstattliche Versicherung

_____ _____

Name, Vorname Matr.-Nr.

Ich versichere hiermit an Eides statt, dass ich die vorliegende Bachelorarbeit/Masterarbeit* mit dem Titel

_____

_____

_____

selbstständig und ohne unzulässige fremde Hilfe erbracht habe. Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie wörtliche und sinngemäße Zitate kenntlich gemacht. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

_____ _____

Ort, Datum Unterschrift

*Nichtzutreffendes bitte streichen

Belehrung:

Wer vorsätzlich gegen eine die Täuschung über Prüfungsleistungen betreffende Regelung einer Hochschulprüfungsordnung verstößt, handelt ordnungswidrig. Die Ordnungswidrigkeit kann mit einer Geldbuße von bis zu 50.000,00 € geahndet werden. Zuständige Verwaltungsbehörde für die Verfolgung und Ahndung von Ordnungswidrigkeiten ist der Kanzler/die Kanzlerin der Technischen Universität Dortmund. Im Falle eines mehrfachen oder sonstigen schwerwiegenden Täuschungsversuches kann der Prüfling zudem exmatrikuliert werden. (§ 63 Abs. 5 Hochschulgesetz - HG -)

Die Abgabe einer falschen Versicherung an Eides statt wird mit Freiheitsstrafe bis zu 3 Jahren oder mit Geldstrafe bestraft.

Die Technische Universität Dortmund wird gfls. elektronische Vergleichswerkzeuge (wie z.B. die Software „turnitin“) zur Überprüfung von Ordnungswidrigkeiten in Prüfungsverfahren nutzen.

Die oben stehende Belehrung habe ich zur Kenntnis genommen:

_____ _____

Ort, Datum Unterschrift