NEURONALE ANSÄTZE ZUR SEMANTISCHEN ANALYSE HANDSCHRIFTLICHER DOKUMENTENBILDER

Dissertation
zur Erlangung des Grades eines

DOKTORS DER INGENIEURWISSENSCHAFTEN

der Technischen Universität Dortmund
an der Fakultät für Informatik

von

OLIVER TÜSELMANN

Dortmund

2024

Tag der mündlichen Prüfung:	31.10.2024
Dekan:	Prof. Dr. Jens Teubner
Gutachter:	Prof. Dr.-Ing. Gernot A. Fink Prof. Dr. Andreas Fischer

NOTATION

Bei der Verwendung mathematischer Ausdrücke in dieser Arbeit werden alle Variablen, Funktionen und Argumente jeweils an der Stelle definiert, an der sie zuerst erscheinen. Im Folgenden wird eine Übersicht der in dieser Dissertation verwendeten Notationen und Symbole gegeben.

Analysis

$\mathrm{f}(a), \mathrm{f}_{name}(a)$	Funktionswert von f bzw. f_{name} für das Element a
$\mathrm{f} : \mathbf{A} \to \mathbf{B}$	die Funktion f bildet von der Menge $\mathbf{A}$ in die Menge $\mathbf{B}$ ab
$\sum_{i=0}^{n-1}, \sum_{\mathbf{a} \in \mathbf{A}}$	Summe von $i = 0$ bis $n - 1$ bzw. über alle Elemente in der Menge $\mathbf{A}$
$\mathrm{f} * \mathrm{g}$	Faltung der Funktionen f und g
f'	Erste Ableitung der Funktion f
$\frac{\partial \mathrm{f}}{\partial a}$	Partielle Ableitung der Funktion f nach a
$\nabla \mathrm{f}$	Gradient der Funktion f

Arithmetik

a	Skalar
$[a, b]$	Intervall der reellen Zahlen zwischen a und b (einschließlich)
$\sqrt{a}$	Quadratwurzel aus a
a^x	x-te Potenz von a
$\log(a)$	Logarithmus von a zur Basis 2
e	Eulersche Zahl
$a \cdot b$	Skalare Multiplikation
$a \equiv b \pmod{c}$	a und b sind kongruent modulo c
$\lfloor a \rfloor$	Abrundung von a zur nächsten Ganzzahl

Stochastik

$\mathrm{P}(A \mid B)$	Wahrscheinlichkeit von A unter der Voraussetzung B
$\mathrm{Var}(A)$	Standardabweichung der Zufallsvariable A
$\mathrm{Cov}(A, B)$	Kovarianz der Zufallsvariablen A und B
$\mathrm{Std}(A)$	Standardabweichung der Zufallsvariable A

Mengenlehre

$\mathbf{A}$	Menge
$\{a \mid \mathrm{T}(a)\}$	Menge der Elemente a, welche die Bedingung $\mathrm{T}(a)$ erfüllen
$\{a, \ldots, b\}$	Menge aller ganzen Zahlen zwischen a und b (einschließlich)
$\mathbf{A} \cup \mathbf{B}$	Vereinigung der Mengen $\mathbf{A}$ und $\mathbf{B}$
$\mathbf{A} \cap \mathbf{B}$	Durchschnitt der Mengen $\mathbf{A}$ und $\mathbf{B}$
$\mathbb{A}^x, \mathbb{A}^{(x \times y)}$	Vektorräume
$\mathbb{R}, \mathbb{N}, \mathbb{Z}$	Menge der reellen, natürlichen bzw. ganzen Zahlen
$\lvert\mathbf{A}\rvert$	Kardinalität der Menge $\mathbf{A}$
$\mathbf{a}^{(i)}$	i-tes Element der Menge $\mathbf{A} = \left\{\mathbf{a}^{(j)}\right\}_{j=0}^{n-1}$
$\bar{\mathbf{A}}$	Durchschnittswert der Menge $\mathbf{A}$, d. h. $\frac{1}{\lvert\mathbf{A}\rvert} \sum_{i=0}^{\lvert\mathbf{A}\rvert-1} \mathbf{a}^{(i)}$

Lineare Algebra

$\boldsymbol{a}$	Spaltenvektor
$\boldsymbol{A}$	Matrix
$\boldsymbol{a}_i$	i-tes Element des Vektors $\boldsymbol{a}$
$\boldsymbol{A}_{i,j}$	Element der Matrix $\boldsymbol{A}$ in der Zeile i und der Spalte j
$\boldsymbol{A}_i$	i-te Zeile der Matrix $\boldsymbol{A}$
$\boldsymbol{A}_{:,i}$	i-te Spalte der Matrix $\boldsymbol{A}$
$\boldsymbol{a} \cdot \boldsymbol{b}$	Skalarprodukt
$\boldsymbol{A} \cdot \boldsymbol{B}$	Produkt der Matrizen $\boldsymbol{A}$ und $\boldsymbol{B}$
$\boldsymbol{A} \odot \boldsymbol{B}$	Hadamard-Produkt der Matrizen $\boldsymbol{A}$ und $\boldsymbol{B}$
$\boldsymbol{a} \parallel \boldsymbol{b}$	Konkatenation der Vektoren $\boldsymbol{a}$ und $\boldsymbol{b}$
$\lVert\boldsymbol{a}\rVert$	Länge des Vektors $\boldsymbol{a}$
$\boldsymbol{A}^\intercal$, $\boldsymbol{a}^\intercal$	Transponierte Matrix von $\boldsymbol{A}$ bzw. transponierter Vektor von $\boldsymbol{a}$

In einigen Fällen wird eine skalare Funktion $\mathrm{f}(\cdot)$ auf ein Argument in Vektor-, Matrix- oder Mengenform angewendet. Falls nicht anders angegeben, wird in diesen Fällen $\mathrm{f}(\cdot)$ elementweise auf alle Elemente des Vektors, der Matrix oder der Menge angewendet. Zudem können Variablennamen von Matrizen, Vektoren und Mengen zur besseren Lesbarkeit mit einem hochgestellten Symbol oder einer natürlichen Zahl versehen werden, z.B. $\boldsymbol{A}^{(i)}$, $\boldsymbol{a}^{(i)}$ und $\mathbf{A}^{(i)}$. Für skalare Variablen werden zur besseren Lesbarkeit tiefgestellte Symbole verwendet.

INHALTSVERZEICHNIS

1 EINLEITUNG

Seit mehreren Jahrzehnten werden weltweit historische Dokumentenbestände von Institutionen, Unternehmen und Organisationen digitalisiert. Inzwischen wurden bereits mehrere Milliarden Dokumente in eine elektronische Form überführt und damit der breiten Öffentlichkeit zugänglich gemacht [48]. Nach diesem großen Erfolg hinsichtlich der Bewahrung von Informationen ist die Entwicklung geeigneter Technologien für eine effiziente Suche, Indexierung und Exploration dieser Dokumentenmengen erforderlich. Ähnlich wie bei Websuchmaschinen sind die Ansprüche der Nutzer an die Suchfunktionen digitaler Bibliotheken enorm gestiegen. Die Erwartungen der Nutzer gehen weit über eine einfache Stichwortsuche hinaus. Vielmehr wird eine benutzerfreundliche und effiziente Exploration digitaler Bibliotheken mit natürlichsprachlicher Interaktion zur Informationsbeschaffung erwartet. Dies erfordert die Entwicklung von intelligenten Systemen, welche die natürlichsprachlichen Inhalte in den Dokumenten automatisiert erfassen und analysieren. Dabei haben Textanalyseansätze aus dem *Natural Language Processing* (NLP)-Bereich speziell in den letzten Jahren erhebliche Fortschritte erzielt und werden bereits erfolgreich in einer Vielzahl von realen Anwendungen eingesetzt [4, 42, 122].

Die textuellen Inhalte der digitalisierten Dokumente liegen jedoch nicht zwangsläufig in einem maschinenlesbaren Format vor, sondern werden oftmals durch Bilder von handschriftlich verfassten Texten repräsentiert. Die semantische Analyse dieses Eingabeformats ist aufgrund der Kombination von visuellen und textuellen Eigenschaften sowie der hohen Variabilität von Handschriften eine anspruchsvolle Aufgabenstellung. Einen intuitiven Ansatz zur semantischen Analyse von Dokumentenbildern bieten sogenannte HTR-basierte Ansätze, die aus einer sequentiellen Kombination von Handschrifterkennung (engl.: *Handwritten Text Recognition* (HTR)) und einem textuellen aufgabenspezifischen NLP-System bestehen [19, 215]. Bei diesem Verfahren wird der Text im Dokumentenbild zunächst mit einem HTR-Modell in ein maschinenlesbares Format umgewandelt und anschließend ein NLP-System auf den erhaltenen Text angewendet. Die beiden Modelle werden unabhängig voneinander trainiert, wodurch die Korrektur von Texterkennungsfehlern im semantischen Modell erschwert bzw. unmöglich wird [29, 189]. Um das Problem der Fehlerfortpflanzung zu vermeiden, haben sich HTR-freie Modelle etabliert [1, 166, 206]. Diese Ansätze basieren auf neuronalen Ende-zu-Ende-Architekturen und vermeiden eine explizite Texterkennung. Obwohl das Problem der Fehlerfortpflanzung durch HTR-freie Ansätze zumindest technisch abgemildert werden kann, haben sie den grundlegenden Nachteil, dass sie wichtige Fortschritte aus dem NLP-Bereich, wie z.B. vortrainierte semantische Worteinbettungen, nicht ohne weiteres nutzen können.

Beide Ansätze weisen für die semantische Analyse von handgeschriebenen Dokumentenbildern theoretische Vor- und Nachteile auf. Die Wahl eines optimalen Ansatzes für diese Eingabedaten ist daher eine offene Forschungsfrage. Zur Beantwortung dieser Frage werden in der vorliegenden Arbeit sowohl ein HTR-basierter als auch ein HTR-freier Ansatz vorgestellt und anhand einer Reihe von Benchmarks für die semantische Analyse handgeschriebener Dokumentenbilder evaluiert. Ein Hauptproblem der HTR-freien Ansätze ist das Fehlen von vortrainierten semantischen Worteinbettungen. Aus diesem Grund wird in dieser Arbeit ein cross-modaler Ansatz zur Wissensdestillation vorgestellt, mit

dem das in der textuellen Domäne gewonnene semantische Wissen effizient in die visuelle Domäne übertragen werden kann, ohne dabei eine explizite Texterkennung durchzuführen. Ein entscheidender Punkt in diesem Integrationsprozess ist die Abbildung von handgeschriebenen Wortbildern in einen textuell vortrainierten semantischen Worteinbettungsraum unter Verwendung eines neuronalen Faltungsnetzwerks (engl.: *Convolutional Neural Network* (CNN)). Dabei wird in dieser Arbeit detailliert auf die Herausforderungen einer solchen Abbildung eingegangen und verschiedene Optimierungsansätze vorgestellt.

In dieser Dissertation werden vier Forschungsbeiträge auf dem Gebiet der semantischen Analyse von Dokumentenbildern geleistet. Diese Beiträge werden im Abschnitt 1.1 ausführlich beschrieben und es wird angegeben, ob und wo Teile der Beiträge bereits veröffentlicht wurden. Eine detaillierte Übersicht über die Gliederung der Arbeit ist im Abschnitt 1.2 aufgeführt.

1.1 BEITRÄGE

Bei der Entwicklung der in dieser Arbeit vorgestellten Ansätze wurden zahlreiche Beiträge auf dem Gebiet der semantischen Analyse von handschriftlichen Dokumentenbildern geleistet. Alle Beiträge wurden bereits auf etablierten wissenschaftlichen Konferenzen, Workshops oder in Fachzeitschriften veröffentlicht. Im Folgenden werden die Forschungsbeiträge detailliert beschrieben, wobei im Falle gemeinsamer Publikationen die individuellen Beiträge des Autors dieser Arbeit aufgeführt sind.

Erstellung und Veröffentlichung von Datensätzen für die semantische Analyse von handschriftlichen Dokumentenbildern

Die semantische Analyse von handschriftlichen Dokumentenbildern ist ein noch junges Forschungsgebiet, weshalb bisher nur wenige Benchmarks für das Training und die Evaluierung entsprechender Modelle zur Verfügung stehen. Im Forschungsbereich der Erkennung und Verlinkung von benannten Entitäten (engl.: *Named Entities* (NEs)) auf handschriftlichen Dokumentenbildern existieren bisher nur private [22, 202], synthetisch generierte [22] oder teilstrukturierte [161] Datensätze. Die fehlenden Daten behindern den Fortschritt in diesem Bereich massiv, da insbesondere im Zusammenhang mit den neuronalen Modellen annotierte Trainingsdaten von großer Relevanz sind. Darüber hinaus ist die Vergleichbarkeit mit Ansätzen aus der Literatur nur eingeschränkt möglich, da die Evaluierung häufig auf urheberrechtlich geschützten oder unveröffentlichten Datensätzen mit automatisiert erstellten Annotationen erfolgt [166]. Hinzu kommt die Verwendung unterschiedlicher Metriken und Protokolle bei der Evaluierung bereits veröffentlichter Ansätze in diesem Bereich. Aus diesen Gründen wurden im Laufe der Dissertation bekannte Datensätze aus der Dokumentenanalyse manuell mit *Named Entity Recognition* (NER)- als auch *Named Entity Linking* (NEL)-Annotationen [211, 215] annotiert und zusammen mit einer optimierten Verteilung von Trainings-, Test- und Validierungsdaten veröffentlicht. Darüber hinaus wurde ein Evaluierungsprotokoll für die NER- und NEL-Aufgaben festgelegt. Die Datensätze umfassen sowohl moderne als auch historische Bilder von handgeschriebenen Dokumenten, die natürlichsprachlichen Text in englischer Sprache enthalten. Die NER- und NEL-Datensätze wurden bereits in [211, 215] veröffentlicht. In beiden Publikationen war der Autor dieser Dissertation für die Planung und Umsetzung des manuellen An-

notationsprozesses sowie für die Durchführung der Experimente auf diesen Datensätzen verantwortlich.

Entwicklung und Vergleich von HTR-freien und HTR-basierten Ansätzen zur semantische Analyse von handschriftlichen Dokumentenbildern

Für die semantische Analyse von handschriftlichen Dokumentenbildern ist die Wahl eines optimalen Ansatzes nach wie vor eine offene Forschungsfrage. In der Literatur haben sich sogenannte HTR-freie [133, 166, 207] und HTR-basierte [19, 215] Ansätze für diese Aufgabe etabliert, wobei es sowohl Argumente für den einen als auch für den anderen Ansatz gibt. Der HTR-basierte Ansatz besteht aus einem zweistufigen Prozess, bei dem zunächst das Dokumentenbild mit einem Handschrifterkennungsmodell in ein maschinenlesbares Format umgewandelt wird. Anschließend wird ein anwendungsspezifisches NLP-Modell auf diese Ausgabe angewendet. Das HTR-freie Verfahren hingegen vermeidet eine explizite Texterkennung und wandelt die Dokumente in vektorielle statt in textuelle Repräsentationen um, die dann als Eingabe für anwendungsspezifische semantische Modelle dienen. Im Rahmen dieser Arbeit werden sowohl ein HTR-basiertes als auch ein HTR-freies Verfahren entwickelt und hinsichtlich semantischer Benchmarks verglichen. Für die Realisierung des HTR-basierten Verfahrens wird ein geeignetes Handschrifterkennungsmodell reimplementiert. Zudem werden *Question Answering* (QA)-, *Word Spotting* (WS)- und NER-Modelle aus dem NLP-Bereich implementiert und mit dem HTR-Modell in geeigneter Weise für die Analyse handschriftlicher Dokumentenbilder kombiniert. Für den HTR-freien Ansatz wird ein CNN zur Transformation von Wortbildern in vektorielle Repräsentationen vorgestellt und geeignete semantische Modelle entwickelt, die auf vektorieller statt textueller Eingabe basieren. Vorläufige Versionen und Ergebnisse der vorgestellten Ansätze wurden bereits in Konferenzbeiträgen veröffentlicht. In [215] wird das HTR-basierte Modell für die NER-Aufgabe und in [213] das HTR-freie Modell für die QA-Aufgabe vorgestellt. In beiden Veröffentlichungen war der Autor dieser Dissertation für den Entwurf der Architekturen und die Durchführung der Experimente verantwortlich.

Entwicklung eines effizienten Ansatzes zur Integration von vortrainierten semantischen Informationen aus der Textdomäne in das HTR-freie Modell

HTR-freie Modelle erzielen im Vergleich zu HTR-basierten Ansätzen auf den meisten semantischen Benchmarks in der Literatur geringere Leistungen, obwohl sie das Problem der Fehlerfortpflanzung von HTR-basierten Modellen beheben. In dieser Arbeit wird die Hypothese vertreten, dass dieser Leistungsunterschied hauptsächlich auf die Nichtberücksichtigung von vortrainierten semantischen Worteinbettungen in HTR-freien Modellen zurückzuführen ist. Ein wesentlicher Beitrag dieser Arbeit ist die Entwicklung eines cross-modalen Destillationsansatzes, mit dem semantische Informationen aus vortrainierten textuellen Worteinbettungsmodellen effizient in HTR-freie Modelle integriert werden können. Das Destillationsverfahren basiert auf einem Lehrer-Schüler-Ansatz, bei dem eine Abbildung von handschriftlichen Wortbildern in einen vortrainierten semantischen Worteinbettungsraum mit einem CNN erlernt wird. Dabei besteht das Lehrermodell aus einem textuell vortrainierten semantischen Worteinbettungsmodell und das Schülermodell aus einem zu-

fällig initialisierten CNN. Das Training basiert auf manuell annotierten handschriftlichen Wortbildern, wobei das Wortbild als Eingabe für das CNN und dessen textuelle Annotation zur Generierung der Zielrepräsentationen mit dem Lehrermodell verwendet wird. Nach dem Training des CNNs können semantische Wortbildrepräsentationen mit dem Schülermodell vorhergesagt werden, ohne dass eine explizite Texterkennung durchgeführt werden muss. In dieser Arbeit werden zudem Ansätze zur optimalen Integration des Schülermodells in das HTR-freie Verfahren vorgestellt und evaluiert. Der Ansatz der cross-modalen Wissensdestillation und die zugehörigen Evaluationen wurde bereits in [216] veröffentlicht. Der Autor dieser Dissertation war sowohl für die Entwicklung der Methodik als auch für die Durchführung der Experimente verantwortlich.

Entwicklung und Analyse von Strategien zur Optimierung einer robusten semantischen Repräsentation für handschriftliche Wortbilder

Das in dieser Arbeit vorgestellte Verfahren zur cross-modalen Wissensdestillation basiert auf einer robusten Abbildung von handschriftlichen Wortbildern in einen vortrainierten semantischen Worteinbettungsraum. Ein zentraler Parameter dieses Verfahrens ist die Wahl eines geeigneten Modells zur semantischen Wortrepräsentation. Daher werden in dieser Arbeit semantische Worteinbettungsmodelle aus dem NLP-Bereich hinsichtlich ihrer Eignung zur semantischen Repräsentation von handschriftlichen Wortbildern evaluiert. Ein zentrales Problem des Destillationsverfahrens ist die unzureichende Repräsentation von semantischen Eigenschaften für Wörter, die nicht im Trainingsprozess der Destillation vorkamen. Insbesondere die geringe Anzahl von manuell annotierten Trainingsdaten in der Handschriftdomäne ist in diesem Zusammenhang problematisch. Zur Lösung dieses Problems werden die Auswirkungen der Hinzunahme synthetisch generierter Wortbilder während des Destillationsprozesses evaluiert. Ein weiterer Ansatz zur Erhöhung der Robustheit von Wortbildrepräsentationen besteht in der Entwicklung einer geeigneten Kombination aus syntaktischen und semantischen Worteinbettungen. Die Idee dieser Kombination basiert auf der hohen Robustheit für die Vorhersage der syntaktischen Repräsentation auf handschriftlichen Wortbildern, sodass auch bei einer fehlerhaften Vorhersage der semantischen Repräsentation die Information über die Syntax des Wortes für nachfolgende Modelle zur Verfügung steht. Vorläufige Versionen und Ergebnisse der vorgestellten Ansätze wurden bereits in [210], [212] und [214] veröffentlicht. Dabei werden in [214] die Herausforderungen einer semantischen Wortbildrepräsentation identifiziert und ein optimierter Ansatz zur semantischen Schlüsselwortsuche sowie eine neue Bewertungsmetrik vorgestellt. In [210] wird das Verfahren zur geeigneten Kombination semantischer und syntaktischer Worteinbettungen und in [212] die Analyse semantischer Worteinbettungsverfahren aus dem NLP-Bereich hinsichtlich ihrer Eignung zur semantischen Repräsentation von handschriftlichen Wortbildern präsentiert. In den Veröffentlichungen war der Autor dieser Dissertation sowohl für die Konzeption der Ansätze als auch für die Durchführung der Experimente verantwortlich.

1.2 AUFBAU DER ARBEIT

Der Hauptteil dieser Arbeit ist in sieben Kapitel gegliedert, wobei das vorliegende Kapitel als Einleitung und Motivation dient. Die übrigen Kapitel sind wie folgt strukturiert:

Kap. 2: METHODISCHE GRUNDLAGEN
Die in dieser Arbeit verwendeten Ansätze basieren auf dem maschinellen Lernen mit neuronalen Modellen. Folglich werden in diesem Kapitel sowohl die Grundlagen neuronaler Ansätze als auch die für diese Arbeit relevanten neuronalen Architekturen vorgestellt. Dabei wird die Entwicklung von einem einzelnen Perzeptron bis zum Training eines mehrschichtigen Perzeptrons beschrieben. Zudem wird das Konzept der Faltungsnetzwerke sowie eine spezielle Architekturen zur effizienten Verarbeitung von Bilddaten vorgestellt. Abschließend werden Modelle zur Verarbeitung sequentieller Daten eingeführt, wobei insbesondere auf rekurrente Ansätze und *Transformer*-Architekturen eingegangen wird.

Kap. 3: SEMANTISCHE DOKUMENTENBILDANALYSE
Dieses Kapitel bietet einen Überblick über das relativ neue Forschungsgebiet der semantischen Dokumentenbildanalyse. Dabei handelt es sich um ein interdisziplinäres Forschungsfeld der Dokumentenbildanalyse und des NLP. Das Kapitel beginnt mit einer Einführung in die beiden Disziplinen und bietet anschließend einen detaillierten Literaturüberblick über die Anwendungsgebiete der semantischen Dokumentenbildanalyse. Aufgrund der großen Anwendungsvielfalt in diesem Bereich beschränkt sich der Literaturüberblick auf drei repräsentative semantische Aufgaben. Bei diesen Aufgaben handelt es sich um das *Information Retrieval*, die NER und das QA. Zunächst werden die Aufgaben ausführlich definiert und beschrieben. Anschließend wird die Literatur nach dem erwarteten Eingabeformat gegliedert und die wichtigsten Fortschritte in den jeweiligen Anwendungsgebieten vorgestellt. Dabei werden sowohl Ansätze für bildbasierte als auch für maschinenlesbare Dokumente berücksichtigt. Am Ende des Kapitels werden die Vor- und Nachteile der betrachteten Ansätze zusammengefasst und eine Motivation für die in dieser Arbeit vorgestellte Methodik gegeben.

Kap. 4: NEURONALE MODELLE ZUR SEMANTISCHEN DOKUMENTENBILDANALYSE
Für die semantische Analyse von handgeschriebenen Dokumentenbildern werden in diesem Kapitel zwei Ansätze vorgestellt. Der erste Ansatz basiert auf der sequentiellen Anwendung eines Texterkenners und eines anwendungsspezifischen NLP-Modells. Der zweite Ansatz vermeidet eine explizite Texterkennung und wandelt das Dokumentenbild zunächst in eine Sequenz von vektoriellen Repräsentationen um, die anschließend von einem anwendungsspezifischen semantischen Modell verarbeitet werden. In diesem Kapitel werden für beide Ansätze die einzelnen Komponenten detailliert vorgestellt. Hierbei wird speziell auf die Texterkennung beziehungsweise die Wortbildeinbettung sowie auf die Modelle für die semantische Schlüsselwortsuche, die NER und das QA eingegangen.

Kap. 5: CROSS-MODALE WISSENSDESTILLATION
In diesem Kapitel wird eine cross-modale Destillationsstrategie zur Integration von semantischem Wissen aus textuell vortrainierten Worteinbettungsmodellen in das HTR-freie Verfahren vorgestellt. Der Ansatz basiert auf einer robusten Abbildung von handschriftlichen Wortbildern in einen vortrainierten semantischen Worteinbettungsraum, ohne eine explizite Texterkennung durchzuführen. Dazu werden zunächst vielversprechende textuelle Worteinbettungsmodelle aus dem NLP-Bereich präsentiert. Anschließend wird eine geeignete Kombination aus semantischer und syntaktischer Worteinbettung als robuste semantische Wortbildrepräsentation vorgeschlagen. Um das Problem von Datensätzen mit wenigen annotierten Trainingsdaten zu lösen, wird eine annotationsfreie Destillationsstrategie vorgestellt, die auf synthetisch generierten Wortbildern basiert. Das Kapitel schließt

mit der Beschreibung von drei Strategien zur Integration des semantischen Einbettungsmodells in den HTR-freien Ansatz.

Kap. 6: EXPERIMENTELLE EVALUATION
In diesem Kapitel wird eine qualitative und quantitative Bewertung der HTR-freien und HTR-basierten Ansätze anhand von drei verschiedenen semantischen Aufgaben vorgenommen. Dazu wird die Leistungsfähigkeit der Modelle anhand bewährter Benchmarks gemessen und mit Ansätzen aus der Literatur verglichen. Die Benchmarks und Bewertungskriterien werden zu Beginn des Kapitels präsentiert. Anschließend werden die konzeptionellen Designentscheidungen des vorgestellten Ansatzes zur Wissensdestillation untersucht und die Auswirkungen auf die Leistung des HTR-freien Ansatzes analysiert. In einer abschließenden Diskussion werden weiterführende Experimente zum Vergleich der Robustheit von HTR-freien und HTR-basierten Ansätzen vorgestellt.

Kap. 7: FAZIT
In diesem Kapitel werden die wichtigsten Erkenntnisse und Resultate aus den Experimenten dieser Arbeit zusammengefasst. Zudem wird ein Ausblick über potentiell zukünftige Forschungsarbeiten im Bereich der HTR-freien semantischen Analyse von handschriftlichen Dokumentenbildern gegeben.

Anhang

Nach dem Hauptteil der Arbeit folgt ein Anhang, der eine ausführliche Liste der Publikationen des Autors, weitere Ergebnisse der experimentellen Auswertungen, das Literaturverzeichnis und eine Liste mit Definitionen aller verwendeten Akronyme enthält.

2 METHODISCHE GRUNDLAGEN

In diesem Kapitel werden die methodischen Grundlagen dieser Arbeit vorgestellt. Die in der vorliegenden Arbeit verwendeten Ansätze basieren auf dem maschinellen Lernen mit neuronalen Netzwerken. Daher werden im Abschnitt 2.1 zunächst die grundlegenden Begriffe und Verfahren des maschinellen Lernens beschrieben. Anschließend wird ein detaillierter Überblick über neuronale Netzwerke und die für diese Arbeit relevanten neuronalen Architekturen gegeben. Im Abschnitt 2.2 wird die Entwicklung von künstlichen neuronalen Netzwerken ausgehend von einem einzelnen Perzeptron bis hin zum Training eines mehrschichtigen Perzeptrons dargestellt. Im Abschnitt 2.3 wird das allgemeine Konzept der Faltungsnetzwerke und im Abschnitt 2.4 eine spezielle Architektur zur effizienten Verarbeitung von Bilddaten vorgestellt. Abschließend werden Modelle zur Verarbeitung sequentieller Daten erläutert, wobei speziell auf rekurrente neuronale Netzwerke (siehe Abschnitt 2.5) und Transformer-Architekturen (siehe Abschnitt 2.6) eingegangen wird.

2.1 GRUNDLAGEN DES MASCHINELLEN LERNENS

Das maschinelle Lernen ist ein Teilgebiet der Künstlichen Intelligenz, das sich mit der Entwicklung von Algorithmen und Techniken befasst, die es computergestützten Systemen ermöglichen, aus Daten zu lernen [63, S.8]. Das allgemeine Ziel ist die Entwicklung von Verfahren, mit denen die Lösung eines vorliegenden Problems aus gegebenen Daten automatisch erlernt werden kann, ohne dass ein System explizit für das Problem programmiert werden muss [175]. Diese Ansätze werden vor allem bei komplexen Problemen eingesetzt, bei denen herkömmliche algorithmische Ansätze nur schwer realisierbar sind. Die Anwendungsgebiete des maschinellen Lernens sind vielfältig und konnten insbesondere in den letzten Jahren nahezu alle Branchen und Disziplinen nachhaltig beeinflussen [250, S.26]. Beispiele hierfür sind das maschinelle Sehen, die automatische Sprachverarbeitung und das autonome Fahren [250, S.2].

Beim maschinellen Lernen gibt es verschiedene Arten von Lernverfahren, die von den zur Verfügung stehenden Daten und dem Ziel des Lernprozesses abhängen. Für jedes dieser Verfahren existieren spezifische Techniken, Algorithmen und Anwendungen, die für unterschiedliche Probleme und Datentypen geeignet sind. Im Wesentlichen können drei Paradigmen des maschinellen Lernens unterschieden werden: überwachtes, unüberwachtes und bestärkendes Lernen [250, S.7-20]. Das überwachte Lernen (engl.: *Supervised Learning*) basiert auf annotierten Daten, wobei für jedes Element der Trainingsmenge neben der Eingabe auch die zugehörige Ausgabe gegeben ist. Das Lernziel besteht darin, eine Abbildungsfunktion zu entwickeln, welche die Beziehung zwischen den Eingaben und den Ausgaben modelliert, um Vorhersagen für neue, noch nicht gesehene Daten zu ermöglichen [250, S.7-8]. Beim unüberwachten Lernen (engl.: *Unsupervised Learning*) verfügt das Modell nur über die Eingabedaten ohne die zugehörigen Ausgaben. Das Modell lernt Muster und Strukturen in den Daten zu erkennen und kann zur Gruppierung, Segmentierung oder Datenreduktion verwendet werden [17, S.3]. Das bestärkende Lernen (engl.: *Reinforcement Learning*) unterscheidet sich grundlegend von den beiden vorhergehenden Ansätzen. Bei diesem Verfahren interagiert das Modell mit seiner Umwelt, indem es Aktionen aus-

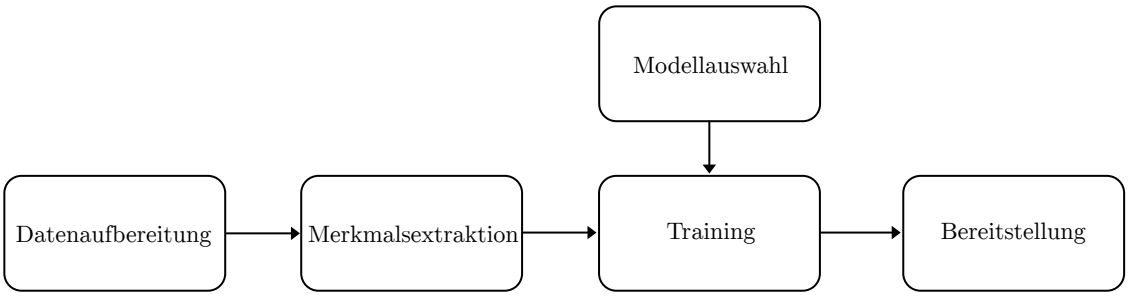


Abbildung 2.1: Eine Visualisierung der klassischen Vorgehensweise zur Modellbildung mit überwachten Lernverfahren aus dem Bereich des maschinellen Lernens.

führt und dafür Belohnungen erhält oder Bestrafungen erfährt [17, S.3]. Auf diese Weise lernt das System Verhaltensregeln in Bezug auf Aktionen, die in verschiedenen Zuständen ausgeführt werden sollten, um die kumulative Belohnung zu maximieren. Typische Anwendungen dieses Lernverfahrens sind Spielstrategien, die Steuerung von Robotern und die automatische Entscheidungsfindung [250, S.809-810].

Die vorliegende Arbeit beschränkt sich ausschließlich auf das Verfahren des überwachten Lernens mit mathematischen Modellen. Diese Modelle lernen anhand einer vorgegebenen annotierten Datensammlung die Approximation einer Abbildungsfunktion für eine gegebene Problemstellung. Trotz der großen Vielfalt an Aufgaben und Konzepten in diesem Bereich weisen die überwachten maschinellen Lernverfahren eine allgemeine Vorgehensweise bei der Modellerstellung auf [250, S.8]. Dieses Vorgehen ist in Abbildung 2.1 visualisiert und wird im Folgenden entlang der einzelnen Schritte von der Datenerhebung bis zur Bereitstellung des Modells beschrieben.

Datenaufbereitung

Die Datenaufbereitung ist ein grundlegender Bestandteil des maschinellen Lernverfahrens und umfasst neben der Erstellung einer annotierten Stichprobe auch die Datenvorverarbeitung und -partitionierung. Dabei sei $\mathbf{X}$ die Menge aller möglichen Instanzen, die der maschinelle Lernalgorithmus beobachten oder verarbeiten kann. Dies können je nach Problemstellung beispielsweise Bilder, Textdokumente, numerische Datenpunkte, Audioaufzeichnungen oder Sensormesswerte sein. Da aus praktischen Gründen üblicherweise nicht alle Instanzen aus $\mathbf{X}$ erfasst und beim Training berücksichtigt werden können, wird eine Stichprobe für die Erstellung und Bewertung des Modells benötigt. Dazu werden zuerst Daten aufgenommen bzw. gesammelt, die für das spezifische Problem relevant sind und für das Training und die Evaluierung des Modells verwendet werden. Die Daten müssen für das Problem repräsentativ sein, um eine Annäherung an die reale Abbildungsfunktion zu ermöglichen und eine aussagekräftige Bewertungsgrundlage zu bieten [250, S.6]. Der Prozess des überwachten Lernens erfordert neben der Erfassung der Daten auch die Vorgabe der gewünschten Ausgabe für jedes Element der Stichprobe. Dies ist ein ressourcenintensiver Prozess, der in der Regel einen hohen Zeit- und Kostenaufwand erfordert [250, S.119]. In diesem Zusammenhang sei $\mathbf{D} = \{(\boldsymbol{x}^{(0)}, \boldsymbol{y}^{(0)}), \ldots, (\boldsymbol{x}^{(t-1)}, \boldsymbol{y}^{(t-1)})\}$, die annotierte Stichprobe, wobei $\boldsymbol{x}^{(i)} \in \mathbf{X}$ die Eingabe und $\boldsymbol{y}^{(i)}$ dessen gewünschte Ausgabe ist.

Sobald die Daten erfasst sind, müssen diese in der Regel vorverarbeitet werden, um sicherzustellen, dass sie in einem für das Modell geeigneten Format vorliegen. Dazu gehören Aufgaben wie die Datenbereinigung, die Behandlung fehlender Werte und die Eliminierung

von Ausreißern [250, S.40]. Darüber hinaus besteht das Ziel der Vorverarbeitung darin, die Eingabedaten für die weitere Verarbeitung im Modell zu verbessern [17, S.2-3]. Dabei wird davon ausgegangen, dass das Problem nach der Vorverarbeitung für das Modell leichter zu lösen ist [63, S.453-454]. Dies kann z.B. die Umwandlung eines Farbbildes in ein Graustufenbild oder die Entfernung von Stoppwörtern aus der Eingabe in einer Textverarbeitung sein.

Am Ende der Datenaufbereitung wird die annotierte Stichprobe in drei disjunkte Mengen aufgeteilt: Trainingsdaten, Validierungsdaten und Testdaten. Die Trainingsdaten werden zum Trainieren des Modells verwendet. Mit den Validierungsdaten wird die Leistung des Modells während des Trainings überwacht und wichtige Hyperparameter optimiert. Der Testdatensatz wird verwendet, um die endgültige Leistung des Modells nach dem Training zu bewerten.

Merkmalsextraktion

Zur Verarbeitung der Eingabedaten mit einem maschinellen Lernverfahren werden die Daten in eine numerische Merkmalsrepräsentation transformiert [17, S.2]. Dieser Prozess ermöglicht zudem eine vereinfachte Lösung der Aufgabenstellung für das Modell, indem problemrelevante Informationen bzw. Merkmale aus den Rohdaten extrahiert und irrelevante Informationen verworfen werden. Bei der Merkmalsextraktion werden generell zwischen dem klassischen und *Deep Learning* (DL)-basierten Ansätzen unterschieden [250, S.28]. Die klassische Merkmalsextraktion erzeugt Merkmale manuell oder halbautomatisch, während diese beim DL automatisch aus den Trainingsdaten gelernt werden. Der klassische Ansatz ist ein zeitaufwändiger und kreativer Prozess, der Fachwissen über die Daten und das Problem erfordert, um die bestmöglichen Merkmale zu extrahieren [63, S.4]. Dabei sind heuristische und analytische Methoden etabliert. Während die heuristischen Methoden auf vordefinierten Regeln, Annahmen und Fachwissen basieren, verwenden die analytischen Methoden mathematische bzw. statistische Verfahren zur Extraktion relevanter Merkmale aus den Trainingsdaten [63, S.147]. Die analytischen Methoden basieren auf der Optimierung eines vorgegebenen Kriteriums und führen dabei häufig eine Dimensionsreduktion der Eingabedaten durch [63, S.147]. Ein Beispiel für ein analytisches Verfahren ist die Hauptkomponentenanalyse [63, S.147-150]. Im Gegensatz zur klassischen Merkmalsextraktion, können DL-Modelle komplexe Merkmale und Muster direkt aus den Rohdaten lernen und extrahieren [63, S.5]. Dieses Vorgehen ist besonders wirksam bei der Verarbeitung von unstrukturierten Daten wie Bildern, Texten und Audiosignalen, da diese oft komplexe Merkmale enthalten, die schwierig manuell zu modellieren sind [63, S.3]. Der DL-Ansatz hat den grundsätzlichen Vorteil, dass die Merkmalsextraktion und das problemspezifische Modell nicht einzeln, sondern durchgängig optimiert werden [250, S.28].

Modellauswahl

Die Auswahl eines geeigneten Modells bzw. einer Modellarchitektur ist von der Problemstellung, den Daten und den verfügbaren Ressourcen abhängig. In der Literatur existiert eine Vielzahl von Ansätzen mit unterschiedlichen Vor- und Nachteilen hinsichtlich der Leistungsfähigkeit, Interpretierbarkeit, Flexibilität und Anwendbarkeit auf verschiedene Datentypen und Problemstellungen [142, S.20]. Die am weitesten verbreiteten Verfahren sind lineare Modelle, baumbasierte Ansätze, *Support Vector Machines* und künstliche neurona-

le Netzwerke [142, S.20]. Lineare Modelle, wie z.B. die lineare Regression, stellen eines der elementarsten Modelle für das überwachte Training dar, bieten jedoch oft nicht die erforderliche Flexibilität, insbesondere wenn komplexe nichtlineare Zusammenhänge modelliert werden sollen [63, S.15]. Baumbasierte Modelle wie z.B. Entscheidungsbäume verwenden eine hierarchische Struktur von Entscheidungsregeln, um Vorhersagen zu treffen. Sie bieten eine hohe Interpretierbarkeit und eine höhere Flexibilität als lineare Modelle [254]. Die *Support Vector Machine* (SVM) [34] ist ein leistungsfähiger Algorithmus, der eine optimale Trennebene zwischen zwei Klassen in hochdimensionalen Merkmalsräumen findet und komplexe Entscheidungsregeln modelliert [17, S.326]. Neuronale Netzwerke sind eine der flexibelsten Modelle, die aus miteinander verbundenen künstlichen Neuronen bestehen. Die neuronalen Modelle können komplexe nichtlineare Beziehungen in den Daten modellieren, benötigen dabei jedoch große Datenmengen und Rechenressourcen für das Training und sind oft weniger interpretierbar als andere Modelle [250, S.279]. Das *No-Free-Lunch* Theorem [47, S.456] besagt, dass es keinen universellen Ansatz für das maschinelle Lernen gibt, der für alle Probleme am besten geeignet ist. Dieses Theorem verdeutlicht somit die Relevanz einer umfassenden Evaluierung verschiedener Ansätze und Modelle für eine gegebene Situation.

Ein weiterer Bestandteil der Modellauswahl ist die Festlegung der Hyperparameter eines verwendeten Modells. In der Regel enthalten maschinelle Lernverfahren eine Vielzahl von Hyperparametern, die nicht gelernt, sondern vorab manuell festgelegt werden müssen [250, S.859]. Diese Parameter wirken sich auf die Struktur und das Verhalten von Modellen aus und können deren Leistungsfähigkeit erheblich beeinflussen [250, S.859].

Training

Nach dem das Modell ausgewählt ist und die Daten in einem geeigneten Format vorliegen, wird das Modell an die Trainingsdaten angepasst. Dabei wird mit den annotierten Daten und einem initialen Modell eine Abbildung der Eingabedaten auf die zugehörigen Ausgaben erlernt. Die maschinellen Lernverfahren basieren nicht auf einer einheitlichen Vorgehensweise beim Training der Modelle, sondern verwenden stark unterschiedliche Optimierungskriterien und -verfahren [93]. Die klassische SVM ist beispielsweise für die binäre Klassifikation konzipiert. Das Training besteht aus der Ermittlung einer linearen Trennebene im Merkmalsraum, welche die Trainingselemente der beiden Klassen trennt und dabei eine möglichst große Distanz zu den am nächsten gelegenen Elementen beider Klassen aufweist. Neuronale Netzwerke sind hingegen universelle Funktionsapproximatoren [73] und trainieren das Modell durch Anpassung der Modellparameter mit einem speziellen iterativen Gradientenabstiegsverfahren [169]. Das Ziel des Trainings ist die Anpassung der Modellparameter, sodass die Abweichung zwischen der Modellausgabe und der vorgegebenen Ausgabe für alle Trainingsdaten möglichst minimal ist [250, S.193].

Trotz der großen Unterschiede zwischen den Lernverfahren gibt es beim Training der Modelle einheitliche Herausforderungen, die sich auf die Leistung der Verfahren auswirken. Ein wesentliches Problem ist die Über- und Unteranpassung des Modells an die Trainingsdaten [250, S.116]. Bei der Überanpassung lernt das Modell die Trainingsdaten zu stark auswendig und verallgemeinert schlecht auf neue Daten. Bei einer Unteranpassung ist das Modell hingegen zu begrenzt, um die zugrunde liegende Datenstruktur zu erfassen. Zudem sind die Qualität und Quantität der Daten sowie die verfügbaren Rechenressourcen in vielen Anwendungsszenarien ein limitierender Faktor für die Erstellung eines leistungsfähigen Modells. Eine weitere Herausforderung ist die Hyperparameteroptimierung, bei

der die beste Kombination der Parameter für die vorliegende Situation gesucht wird [250, S.859]. Eine manuelle Anpassung der Hyperparameter ist während des Trainings anhand der Validierungsdaten und einer geeigneten Metrik möglich.

Bereitstellung

Der abschließende Entwicklungsabschnitt ist die Evaluierung des optimierten Modells bezüglich dessen Leistung zur Verarbeitung unbekannter Daten. Dazu wird die Güte des Modells mit dem Testdatensatz und einer geeigneten Metrik ermittelt. Erzielt das Modell zufriedenstellende Ergebnisse, wird es für den Einsatz in der Produktionsumgebung bereitgestellt. Hierbei werden unbekannte Anwendungsdaten ohne zugehörige Annotationen verarbeitet. Dazu durchlaufen die Eingabedaten die gleiche Vorverarbeitung und Merkmalsextraktion wie im Training und werden anschließend als Eingabe für das trainierte Modell verwendet, das die finale Ausgabe erzeugt.

2.2 KÜNSTLICHE NEURONALE NETZWERKE

Das menschliche Gehirn besteht aus durchschnittlich 86 Milliarden Neuronen, die in einer netzwerkartigen Struktur organisiert sind [60, 70]. Neuronen bilden neben den Gliazellen die wohl wichtigste Einheit des Gehirns und sind zur Wahrnehmung, Verarbeitung und Weiterleitung von Signalen im menschlichen Körper verantwortlich [60, S.11]. Der Aufbau eines Neurons besteht unter anderem aus Dendriten, einem Zellkern, einem Axon sowie Nervenenden [60, S.12]. Die Dendriten nehmen elektrische Signale auf und leiten diese an den Zellkern weiter. Der Zellkern sammelt die Signale der Dendriten und löst genau dann ein Aktionspotenzial aus, wenn die Summe aller Eingangssignale einen bestimmten Schwellenwert überschreitet [60, S.12]. Aktionspotentiale sind elektrische Ereignisse, welche die grundlegende Einheit der Kommunikation zwischen Neuronen bilden. Dabei wandert ein elektrisches Signal entlang des Axons und bewirkt die Freisetzung von Neurotransmittern, die dann von den verbundenen Neuronen empfangen und interpretiert werden [60, S.13].

Auf der Basis dieser biologischen Forschungsergebnisse wurden vereinfachte mathematische Modelle des biologischen Neurons zur automatischen Mustererkennung mittels Maschinen entwickelt [136, 139, 162]. Ein erster Ansatz wurde im Jahre 1943 von McCulloch und Pitts in [136] vorgestellt. Das Modell empfängt n binäre Signale als Eingabe und erzeugt einen binären Wert als Ausgabe. Dabei wird die Summe der Eingabewerte $\boldsymbol{x} \in \{0,1\}^n$ berechnet und bei der Überschreitung eines festgelegten Schwellenwerts $b \in \mathbb{N}$ eine 1 und andernfalls eine 0 ausgegeben. Das McCulloch-Pitts-Neuron ist mathematisch durch die Funktion $\text{mcp} : \{0,1\}^n \to \{0,1\}$ beschrieben:

$$\text{mcp}(\boldsymbol{x}) = \begin{cases} 1 & \text{wenn } \left(\sum_{i=0}^{n-1} \boldsymbol{x}_i\right) \geq b \\ 0 & \text{sonst} \end{cases} \tag{2.1}$$

Das Modell hat den grundsätzlichen Nachteil, dass der Schwellenwert manuell eingestellt werden muss und nicht lernbar ist [63, S.15]. Zudem ignoriert das Modell die Bedeutung der Verbindungsstärke zwischen den Neuronen, die sich aus der Hebbschen Lernregel [69] ableitet. Als Erweiterung des McCulloch-Pitts-Neurons entwickelte Frank Rosenblatt 1958 das einschichtige Perzeptron (engl.: *Single-Layer Perceptron*), das die Erkenntnisse von Hebb durch gewichtete Eingänge berücksichtigt und reellwertige Eingaben ermöglicht [162]. Das Perzeptron nach Rosenblatt ist in Abbildung 2.2 visualisiert und mathematisch durch

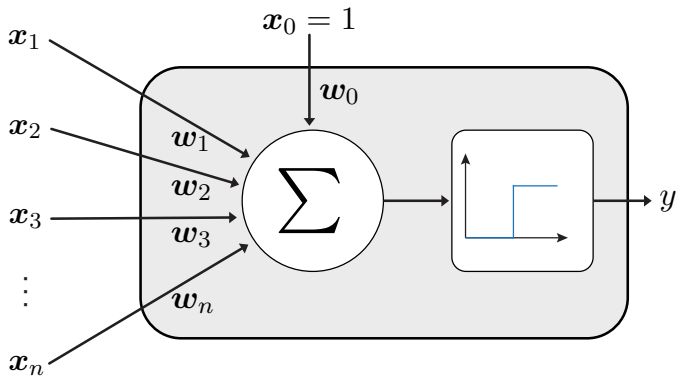


Abbildung 2.2: Eine Visualisierung des Perzeptrons nach Rosenblatt [162]. Das Modell erhält einen reellwertigen Vektor $\boldsymbol{x}$ als Eingabe und berechnet eine gewichtete Summe, die schließlich mit einer Stufenfunktion die Ausgabe y bestimmt. Der Bias-Wert ist durch $\boldsymbol{w}_0$ repräsentiert.

die Funktion $\mathrm{f} : \mathbb{R}^n \to \mathbb{R}$ mit

$$\begin{aligned} \mathrm{g}(\boldsymbol{x}) &= \left(\sum_{i=0}^{n-1} \boldsymbol{w}_i \cdot \boldsymbol{x}_i \right) + b \\ \mathrm{f}(\boldsymbol{x}) &= \mathrm{h}(\mathrm{g}(\boldsymbol{x})) \end{aligned} \tag{2.2}$$

beschrieben. Dabei repräsentiert $\boldsymbol{x} \in \mathbb{R}^n$ die Eingabe, $\boldsymbol{w} \in \mathbb{R}^n$ die Gewichte des Modells und $\mathrm{h} : \mathbb{R} \to \mathbb{R}$ eine nichtlineare Aktivierungsfunktion. Das Modell erzeugt zunächst die sogenannte Aktivierung des Perzeptrons mit der Funktion $\mathrm{g} : \mathbb{R}^n \to \mathbb{R}$ durch Addition des *Bias*-Wertes $\boldsymbol{b} \in \mathbb{R}$ mit der gewichteten Summe der Eingaben. Die Ausgabe des Modells wird durch Anwendung der Funktion $\mathrm{h}(\cdot)$ auf die Aktivierung berechnet, wobei die Aktivierungsfunktion im klassischen Perzeptron-Modell durch die Schwellenwertfunktion (engl.: *Heaviside Step Function*) mit

$$\mathrm{h}(\boldsymbol{x}) = \begin{cases} 1 & \text{wenn } \mathrm{g}(\boldsymbol{x}) > 0 \\ 0 & \text{sonst} \end{cases} \tag{2.3}$$

realisiert wird. Die Berechnungen des Perzeptrons können durch die Betrachtung des Bias-Wertes als zusätzliches Gewicht und der Überführung der gewichteten Summe als Vektormultiplikation vereinfacht werden. Dafür wird die Eingabe durch $\boldsymbol{x} = (1, \boldsymbol{x}_0, \boldsymbol{x}_1, ..., \boldsymbol{x}_{n-1})$ und der Gewichtsvektor durch $\boldsymbol{w} = (b, \boldsymbol{w}_0, \boldsymbol{w}_1, ..., \boldsymbol{w}_{n-1})$ repräsentiert. Die Berechnung eines Perzeptrons erfolgt gemäß:

$$\begin{aligned} \mathrm{f}(\boldsymbol{x}) &= \mathrm{h}\left(\left(\sum_{i=0}^{n-1} \boldsymbol{w}_i \cdot \boldsymbol{x}_i \right) + b \right) \\ &= \mathrm{h}\left(\sum_{i=0}^{n} \boldsymbol{w}_i \cdot x_i \right), \text{ mit } \boldsymbol{w}_0 = b \text{ und } \boldsymbol{x}_0 = 1 \\ &= \mathrm{h}\left(\boldsymbol{w}^\intercal \cdot \boldsymbol{x} \right) \end{aligned} \tag{2.4}$$

Neben dem mathematischen Modell entwickelte Rosenblatt ein Lernalgorithmus zur automatischen Bestimmung der Gewichte für linear separierbare, binäre Klassifikationsprobleme [139, 162].

2.2.1 *Mehrschichtiges Perzeptron*

Ein einzelnes Perzeptron kann lediglich linear trennbare Funktionen realisieren und daher insbesondere die logische *XOR*-Funktion nicht modellieren [139]. Dies ist als XOR-Problem bekannt und führte unter anderem dazu, dass die Forschung im Bereich der künstlichen neuronalen Netzwerke für etwa 20 Jahre weitgehend eingestellt wurde [172, S.24]. Ein entscheidender Beitrag zur Lösung dieses Problems ist das mehrschichtige Perzeptron (engl.: *Multi-Layer Perceptron* (MLP)). Ein MLP ist ein künstliches neuronales Netzwerk, welches aus einer Eingabe-, mindestens einer verborgenen und einer Ausgabeschicht besteht. Die Schichten enthalten Neuronen, die zwischen zwei aufeinanderfolgenden Schichten vollständig vernetzt sind. Dabei ist jedes Neuron einer Schicht unidirektional und nicht zyklisch mit allen Neuronen der nächsten Schicht verbunden. Zudem gibt es keine Verbindungen zur vorherigen Ebene und keine Verbindungen, die eine Ebene überspringen. Das MLP gehört zur Klasse der *Feedforward*-Netzwerke, wobei die Eingabedaten unidirektional von der Eingabeschicht, über die verborgenen Schichten zur Ausgabeschicht fließen. Formal besteht ein MLP aus n Schichten, welche auch als Tiefe des Netzwerks bezeichnet werden. Die Schichten können eine unterschiedliche Anzahl an Neuronen besitzen und werden für die Schicht $l \in \{0, \dots, n-1\}$ des MLPs mit $m_l \in \mathbb{N}_{>0}$ angegeben.

Mathematisch ist das MLP als eine Zusammensetzung von n Funktionen definiert, wobei die Ausgabe der Schicht l als Eingabe der Schicht $l+1$ fungiert. Für die Eingabe $\boldsymbol{x} \in \mathbb{R}^{m_0}$ ergibt sich die Ausgabe $\hat{\boldsymbol{y}} \in \mathbb{R}^{m_{n-1}}$ des Netzwerks wie folgt:

$$\hat{\boldsymbol{y}} = \mathrm{f}^{(n-1)}(\mathrm{f}^{(n-2)}(\dots(\mathrm{f}^{(0)}(\boldsymbol{x})))) \tag{2.5}$$

Dabei sei $\boldsymbol{f}^{(l)} \in \mathbb{R}^{m_l}$ die Ausgabe der Funktion $\mathrm{f}^{(l)} : \mathbb{R}^{m_{l-1}} \to \mathbb{R}^{m_l}$. Die Eingabeschicht ist ausschließlich für die Aufnahme der Eingabe in das Netzwerk zuständig und führt keine weitere Verarbeitung durch. Daraus ergibt sich für die Ausgabe der Eingabeschicht:

$$\boldsymbol{f}^{(0)} = \boldsymbol{x} \tag{2.6}$$

Die Berechnung der Ausgabe $\boldsymbol{f}^{(l)}$ für die Schicht $l \in \{1, \dots, n-1\}$ in einem MLP ist analog zu den Berechnungen eines Perzeptrons mit

$$\boldsymbol{f}^{(l)} = \mathrm{h}(\underbrace{\boldsymbol{W}^{(l)} \cdot \boldsymbol{f}^{(l-1)}}_{\boldsymbol{g}^{(l)}}) \tag{2.7}$$

definiert. Dabei ist $\boldsymbol{g}^{(l)} \in \mathbb{R}^{m_l}$ die Aktivierung der Neuronen aus der Schicht l und wird durch die gewichtete Summe bezüglich der Ausgabe aus der vorhergehenden Schicht berechnet. In diesem Zusammenhang ist $\boldsymbol{W}^{(l)} \in \mathbb{R}^{(m_l \times m_{l-1})}$ die Gewichtsmatrix der Schicht l, wobei $\boldsymbol{W}^{(l)}_{i,j}$ das Kantengewicht von Neuron j in Schicht $l-1$ zu Neuron i in Schicht l repräsentiert. Auf die Aktivierungen der Neuronen wird eine elementweise nichtlineare Funktion $\mathrm{h}(\cdot)$ angewendet, die sich für jede Schicht unterscheiden kann. Die Nichtlinearität der Aktivierungsfunktion ist entscheidend, da das MLP ansonsten nicht aussagekräftiger wäre als ein linearer Klassifikator [47, S.307]. Eine weitere wesentliche Voraussetzung für das Training eines MLPs ist, dass die verwendeten Aktivierungsfunktionen differenzierbar sind. Daher wird die Stufenfunktion des Perzeptrons klassischerweise durch eine Sigmoidfunktion ersetzt [139]:

$$\mathrm{sigmoid}(\boldsymbol{g}_i^{(l)}) = \frac{1}{1 + e^{-\boldsymbol{g}_i^{(l)}}} \tag{2.8}$$

Die Verwendung der Aktivierungsfunktion führt jedoch zu einem stagnierenden Lernverhalten in tiefen Netzwerken. Dies wird in der Literatur als *Vanishing Gradient Problem* [250, S.229] bezeichnet und erschwert die Anpassung der Parameter für die ersten Schichten des Netzwerks [16, 62]. Durch eine geeignete Normalisierung und die Verwendung der *Rectified Linear Unit* (ReLU)-Aktivierungsfunktion mit

$$\text{relu}(\boldsymbol{g}_i^{(l)}) = \max(0, \boldsymbol{g}_i^{(l)}) \tag{2.9}$$

kann dieses Problem jedoch weitgehend gelöst werden [197]. Der wesentliche Vorteil dieser Funktion ist die effiziente Berechnung der Ableitung und die Robustheit gegenüber verschwindenden Gradienten während des Lernprozesses.

Mit dem *Universellen Approximationstheorem* [73] kann bewiesen werden, dass theoretisch alle reellwertigen multivariaten Funktionen mit beliebiger Genauigkeit durch ein MLP approximiert werden können. Dazu genügt ein dreistufiges MLP mit geeigneten Gewichten, einer hinreichenden Anzahl von Neuronen und einer geeigneten nichtlinearen Aktivierungsfunktion [73]. Dieses Theorem ist jedoch in der Praxis weitgehend irrelevant, da die Anzahl der versteckten Neuronen für die meisten Funktionen gegen unendlich geht und geeignete Gewichte bestimmt werden müssen.

2.2.2 *Training*

Das MLP ist ein mathematisches Modell zur Approximation von Funktionen, das eine vektorielle Eingabe auf eine gewünschte Ausgabe abbildet. Das grundlegende Ziel besteht in einer möglichst genauen Approximation einer zugrundeliegenden Vorhersagefunktion für ein gegebenes Anwendungsproblem [63, S.168]. Neben den manuell festgelegten Hyperparametern, wie z.B. der Netzwerkarchitektur und den Aktivierungsfunktionen, basiert die Ausgabe maßgeblich auf den Gewichten im Netzwerk. Die Bestimmung von adäquaten Gewichten ist nicht trivial und wird standardmäßig mit einem überwachten Lernverfahren realisiert. Dazu wird eine repräsentative, annotierte Stichprobe $\mathbf{D} = \{(\boldsymbol{x}^{(0)}, \boldsymbol{y}^{(0)}), \ldots, (\boldsymbol{x}^{(t-1)}, \boldsymbol{y}^{(t-1)})\}$ des gegebenen Problems verwendet. Das Ziel des Trainings ist die Bestimmung der Gewichte im Modell, sodass für jede Eingabe $\boldsymbol{x}^{(i)}$ der Stichprobe die Ausgabe des Modells $\hat{\boldsymbol{y}}^{(i)}$ der gewünschten Ausgabe $\boldsymbol{y}^{(i)}$ möglichst ähnlich ist. Um dieses Ziel zu erreichen, wird ein Optimierungsproblem mit einer skalaren Verlustfunktion $\text{c}(\hat{\boldsymbol{y}}, \boldsymbol{y}) \in \mathbb{R}$ definiert. Diese Funktion berechnet einen numerischen Wert, der ein Maß für die Abweichung zwischen der vorhersagten und gewünschten Ausgabe ist. Der Wert ist minimal, wenn die beiden Ausgaben übereinstimmen. Durch die Minimierung der Verlustfunktion wird die Abweichung zwischen der Netzwerkausgabe $\hat{\boldsymbol{y}} \in \mathbb{R}^{n_e}$ und der gewünschten Ausgabe $\boldsymbol{y} \in \mathbb{R}^{n_e}$ zugleich minimiert. Ein Beispiel für eine Verlustfunktion ist der mittlere quadratische Fehler (engl.: *Mean Squared Error* (MSE)):

$$\text{c}(\hat{\boldsymbol{y}}, \boldsymbol{y}) = \frac{1}{2} \cdot \sum_{i=0}^{n_e-1} (\boldsymbol{y}_i - \hat{\boldsymbol{y}}_i)^2 \tag{2.10}$$

Analytische Verfahren zur Lösung des Optimierungsproblems sind oft nicht praktikabel, insbesondere bei komplexen Modellen mit vielen Parametern [250, S.486]. Eine praktikable Lösung bietet der Gradientenabstieg [172, S.719]. Dies ist ein iteratives Optimierungsverfahren zur Minimierung differenzierbarer Funktionen. Der Gradientenabstieg optimiert die Funktionsvariablen lokal, bietet jedoch keine Garantie dafür, das globale Minimum der

Exkurs 1: Kettenregel

Die mehrdimensionale bzw. verallgemeinerte Kettenregel ist für

$$\mathrm{g}\left(\mathrm{h}_0(\boldsymbol{x}), \mathrm{h}_1(\boldsymbol{x}), \ldots, \mathrm{h}_{\mathrm{n}\text{-}1}(\boldsymbol{x})\right)$$

wie folgt definiert:

$$\frac{\partial\, \mathrm{g}}{\partial \boldsymbol{x}_j} = \sum_{i=0}^{n-1} \frac{\partial\, \mathrm{g}}{\partial\, \mathrm{h_i}} \cdot \frac{\partial\, \mathrm{h_i}}{\partial \boldsymbol{x}_j} \tag{2.13}$$

Funktion zu finden [63, S.85]. Das Verfahren basiert auf der Berechnung des Gradienten der Verlustfunktion und ist für die Variable $\boldsymbol{w}_j$ des Modells mit

$$\frac{\partial\, \mathrm{c}(\hat{\boldsymbol{y}}, \boldsymbol{y})}{\partial \boldsymbol{w}_j} \tag{2.11}$$

definiert. Der Gradient ist ein Vektor, der die Richtung des steilsten Anstiegs der Funktion angibt und durch Anpassung der Gewichte entgegen dem Gradienten zu einer Verringerung des Funktionswertes führt [63, S.85]. Konkret werden die Variablen entsprechend einer Lernrate $\eta \in \mathbb{R}_{>0}$ und der Richtung des Gradienten mit

$$\boldsymbol{w}_j \leftarrow \boldsymbol{w}_j - \eta \cdot \frac{\partial c(\hat{\boldsymbol{y}}, \boldsymbol{y})}{\partial \boldsymbol{w}_j} \tag{2.12}$$

aktualisiert. Die Lernrate bestimmt die Größe der Schrittweite in Richtung des Gradienten und ist ein Hyperparameter, der in der Regel manuell eingestellt werden muss. Eine geeignete Wahl der Lernrate ist wichtig, um eine schnelle Konvergenz und Stabilität des Algorithmus zu gewährleisten. Bei einer zu kleinen Lernrate erfolgt die Konvergenz nur sehr langsam, während bei einer zu großen Lernrate eine Divergenz des Optimierungsprozesses auftreten kann [47, S.225]. Die Gewichte werden schrittweise aktualisiert, bis ein Abbruchkriterium erfüllt ist. Dieses Kriterium kann beispielsweise eine maximale Anzahl von Iterationen oder das Erzielen eines Verlustes unter einem bestimmten Schwellenwert sein.

Für das Training neuronaler Netzwerke wird der *Error-Backpropagation-* oder kurz *Backpropagation*-Algorithmus [169] verwendet, der weitgehend auf dem Gradientenabstiegsverfahren basiert. Der Algorithmus beginnt mit einem Netzwerk aus zufällig initialisierten oder vordefinierten Gewichten und passt diese iterativ mit den folgenden vier Schritten an. Im ersten Schritt wird für ein Beispiel $(\boldsymbol{x}, \boldsymbol{y})$ aus der annotierten Stichprobe eine Vorhersage mit dem Netzwerk durch den Vorwärtsdurchlauf (engl.: *Forward Pass*) berechnet. Dabei stellt $\boldsymbol{f}^{(n-1)}$ die Ausgabe der letzten Schicht aus dem MLP dar und ist analog zur Formel 2.5 aus dem vorherigen Abschnitt definiert. Im zweiten Schritt wird die Abweichung zwischen der Vorhersage und der vorgegebenen Annotation des Beispiels unter Verwendung der Verlustfunktion mit $\mathrm{c}\left(\boldsymbol{f}^{(n-1)}, \boldsymbol{y}\right)$ bestimmt. Anschließend werden die anteiligen Fehler für jedes Gewicht berechnet, indem der Fehler schichtweise, beginnend mit der Ausgabe, zurückgeführt wird. Im letzten Schritt werden die Gewichte des Netzwerks analog zum Gradientenabstiegsverfahren mit

$$\boldsymbol{W}_{i,j}^{(l)} \leftarrow \boldsymbol{W}_{i,j}^{(l)} - \eta \cdot \frac{\partial\, \mathrm{c}\left(\boldsymbol{f}^{(n-1)}, \boldsymbol{y}\right)}{\partial \boldsymbol{W}_{i,j}^{(l)}} \tag{2.14}$$

angepasst. Dabei beschreibt $\boldsymbol{W}_{i,j}^{(l)}$ das Kantengewicht von Neuron j in Schicht $l-1$ zu Neuron i in Schicht l. Die Verlustfunktion ist nicht direkt nach den Gewichten ableitbar, sondern basiert auf den Berechnungen der verschachtelten Funktionen im MLP. Zur Berechnung der Ableitung wird die erweiterte Kettenregel (siehe Exkurs 1) auf die Definition des Perzeptrons (siehe Formel 2.2) angewendet:

$$\frac{\partial \, \mathrm{c}\left(\boldsymbol{f}^{(n-1)}, \boldsymbol{y}\right)}{\partial \boldsymbol{W}_{i,j}^{(l)}} = \frac{\partial \, \mathrm{c}\left(\boldsymbol{f}^{(n-1)}, \boldsymbol{y}\right)}{\partial \boldsymbol{f}_{j}^{(l)}} \cdot \frac{\partial \boldsymbol{f}_{j}^{(l)}}{\partial \boldsymbol{g}_{j}^{(l)}} \cdot \frac{\partial \boldsymbol{g}_{j}^{(l)}}{\partial \boldsymbol{W}_{i,j}^{(l)}} \tag{2.15}$$

Dabei repräsentiert $\boldsymbol{f}_{j}^{(l)}$ die Ausgabe des j-ten Neurons in Schicht l und $\boldsymbol{g}_{j}^{(l)}$ dessen Aktivierung vor Anwendung der Aktivierungsfunktion $\mathrm{h}(\cdot)$. Die Ableitung der Neuronenaktivierung nach dem Gewicht ist trivial und ergibt die Ausgabe des Neurons i aus der vorherigen Schicht:

$$\frac{\partial \boldsymbol{g}_{j}^{(l)}}{\partial \boldsymbol{W}_{i,j}^{(l)}} = \frac{\partial \left\{ \sum_{k=0}^{m_{l-1}} \boldsymbol{W}_{k,j}^{(l)} \cdot \boldsymbol{f}_{k}^{(l-1)} \right\}}{\partial \boldsymbol{W}_{i,j}^{(l)}} = \boldsymbol{f}_{i}^{(l-1)} \tag{2.16}$$

Die Ableitung der Ausgabefunktion eines Neurons nach seiner Aktivierung ist in der Regel ebenfalls unkompliziert, da dies der Ableitung der Aktivierungsfunktion entspricht:

$$\frac{\partial \boldsymbol{f}_{j}^{(l)}}{\partial \boldsymbol{g}_{j}^{(l)}} = \frac{\partial \, \mathrm{h}\left(\boldsymbol{g}_{j}^{(l)}\right)}{\partial \boldsymbol{g}_{j}^{(l)}} = \mathrm{h}'\left(\boldsymbol{g}_{j}^{(l)}\right) \tag{2.17}$$

Die Ableitung der Verlustfunktion nach der Ausgabe des Neurons j in Schicht l erfordert eine Fallunterscheidung zwischen der Ausgabeschicht und den restlichen Schichten des Netzwerks. Für die Neuronen der Ausgabeschicht ist die Berechnung weitgehend trivial und entspricht für die MSE-Verlustfunktion der Differenz zwischen der gewünschten und der vorhergesagten Ausgabe am Ausgang j:

$$\frac{\partial \, \mathrm{c}\left(\boldsymbol{f}^{(n-1)}, \boldsymbol{y}\right)}{\partial \boldsymbol{f}_{j}^{(n-1)}} = \boldsymbol{y}_{j} - \boldsymbol{f}_{j}^{(n-1)} \tag{2.18}$$

Für die Neuronen der Schichten $0 \leq l \leq n-2$ ist aufgrund der schichtweisen Berechnungen im MLP die Anwendung der erweiterten Kettenregel notwendig:

$$\frac{\partial \, \mathrm{c}\left(\boldsymbol{f}^{(n-1)}, \boldsymbol{y}\right)}{\partial \boldsymbol{f}_{j}^{(l)}} = \sum_{k=0}^{m_{l+1}} \frac{\partial \, \mathrm{c}\left(\boldsymbol{f}^{(n-1)}, \boldsymbol{y}\right)}{\partial \boldsymbol{f}_{k}^{(l+1)}} \cdot \underbrace{\frac{\partial \boldsymbol{f}_{k}^{(l+1)}}{\partial \boldsymbol{g}_{k}^{(l+1)}}}_{= \mathrm{h}'\left(\boldsymbol{g}_{k}^{(l+1)}\right)} \cdot \underbrace{\frac{\partial \boldsymbol{g}_{k}^{(l+1)}}{\partial \boldsymbol{f}_{j}^{(l)}}}_{= \boldsymbol{W}_{j,k}^{(l+1)}} \tag{2.19}$$

Die letzten beiden Terme können problemlos berechnet werden, während für den ersten Term eine rekursive Berechnung erforderlich ist. Der Fehler der Schicht l hängt somit von den Fehlern der Schicht $l+1$ ab. Dies verdeutlicht den Namen des Algorithmus, bei dem zunächst der Fehler der letzten Schicht berechnet wird und dieser dann sukzessive für die Berechnung der Fehler in den versteckten Schichten zurück propagiert wird.

Bei der Beschreibung des Lernprozesses wurde bisher nur ein Element der Trainingsmenge berücksichtigt. Der Backpropagation-Algorithmus aktualisiert die Gewichte im Netzwerk jedoch auf Grundlage aller Element der Trainingsmenge [172, S.720]. Dazu werden die Gradienten für jedes Element der Trainingsmenge getrennt berechnet und der Mittelwert der Gradienten zur Aktualisierung der Gewichte verwendet. Dieses Verfahren führt zwar generell zu einer stabilen Konvergenz, jedoch auch zu einem aufwendigen und langsamen Lernverhalten. Ein schnelleres Lernverhalten bietet der stochastische Gradientenabstieg [172, S.720], bei dem die Gewichte nach jedem Element der Trainingsmenge angepasst werden [63, S.276]. Dieses Verfahren führt jedoch häufig zu einem instabilen Lernverhalten und erfordert eine hohe Rechenleistung [63, S.276]. Den besten Kompromiss zwischen stabiler Konvergenz und schnellem Lernen bietet der *Mini-Batch* Gradientenabstieg [63, S.276]. Bei diesem Verfahren wird die Trainingsmenge zufällig in sogenannte Mini-Batches mit vorgegebener Größe aufgeteilt. Die Anpassung der Gewichte erfolgt durch die Berechnung des Mittelwertes der Gradienten für alle Elemente aus der Menge. Mit einer geeigneten Größe und zufälliger Generierung der Batches wird der Gradient der gesamten Trainingsmenge approximiert [63, S.277]. Trotz der Vorteile führt dieses Verfahren einen weiteren, manuell einstellbaren Hyperparameter ein.

2.3 FALTUNGSNETZWERKE

Die Verarbeitung von Bilddaten mit einem MLP birgt fundamentale Probleme und ignoriert zudem die spezifischen Eigenschaften von Bilddaten [250, S.233]. Eine der größten Einschränkungen dieses Verfahrens ist die hohe Anzahl an Parametern. Diese ergibt sich aus der Kombination von vollvernetzten Schichten und der hohen Dimensionalität der Eingabedaten und führt zum sogenannten Fluch der Dimensionalität (engl.: *Curse of Dimensionality*) [15]. Zudem erfordert das MLP-Verfahren eine Umwandlung des Eingabebildes in einen eindimensionalen Vektor, wodurch die räumlichen Eigenschaften des Bildes verloren gehen [250, S.233]. Zur Behebung dieser Probleme wurde mit dem neuronalen Faltungsnetzwerk (engl.: *Convolutional Neural Network* (CNN)) [57, 106] ein effizientes und leistungsfähiges Verfahren speziell für Bilddaten entwickelt, das sich jedoch auch für sequentielle Eingabedaten bewährt hat [107].

Das CNN ist ein Feedforward-Netzwerk, das auf dem Prinzip der mathematischen Faltung basiert und dem menschlichen Sehvorgang nachempfunden ist [57, 76, 248]. Das Modell realisiert eine Ende-zu-Ende-Architektur und besteht aus einer Merkmalsextraktion und einem Klassifikator. Hierbei werden zunächst Merkmale durch Faltungsoperationen aus der Eingabe gelernt und extrahiert. Anschließend werden diese als Eingabe für ein MLP verwendet, das die Klassifikation durchführt. Die Extraktionsschichten sind hierarchisch aufgebaut, wobei nicht alle Neuronen zwischen zwei aufeinanderfolgenden Schichten verbunden sind. Die Inspiration für dieses Prinzip stammt aus der Biologie, in der Neuronen im primären visuellen Kortex auf einfache Merkmale wie Kanten und Farben reagieren und daraus in tieferen Schichten immer komplexere Bildmerkmale generieren, die schließlich als Objekte wahrgenommen werden [76]. Das Modell basiert auf dem Prinzip der lokalen Konnektivität, bei dem Filter mit trainierbaren Gewichten und vorgegebener Größe über die Eingabe bewegt werden und das Auftreten des durch den Filter repräsentierten Merkmals in einer sogenannten Merkmalskarte erfasst wird. Aufgrund der Filtergröße stehen einem Neuron nur begrenzte Informationen der Eingabe zur Verfügung [76, 126]. Die Menge der berücksichtigten Eingabedaten wird als rezeptives Feld bezeichnet und erweitert sich

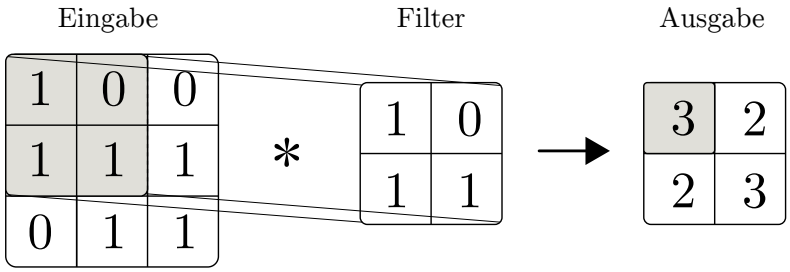


Abbildung 2.3: Eine beispielhafte Visualisierung der Faltungsoperation zwischen einem 2×2 Filter und einer 3×3 Eingabe. Der Filter wird mit der Schrittweite 1 über die Eingabe bewegt.

durch die hierarchische Anordnung in jeder Faltungsschicht [115]. Da die Filter an jeder Position des Eingabebildes mit denselben Gewichten arbeiten, reduziert sich die Anzahl der Parameter im Vergleich zu einem MLP drastisch. Zudem führt dieses Verfahren zu einer Translationsinvarianz, wobei Muster aus der Eingabe unabhängig von ihrer Position, Orientierung und Skalierung identifiziert werden können [250, S.236].

Die Faltungsschicht ist die grundlegende Komponente der CNN-Architektur. Eine Faltungsschicht besteht aus n Filtern mit einer Breite w_f und einer Höhe h_f. Formal wird der Filter $t \in \{0, \ldots, n-1\}$ durch eine Matrix $\boldsymbol{K}^{(t)} \in \mathbb{R}^{(h_f \times w_f \times k)}$ aus lernbaren Gewichten repräsentiert. Dabei ist k die Anzahl der Kanäle, z.B. 3 für ein Farbbild und 1 für ein Graustufenbild. Jeder dieser Filter wird schrittweise über die Eingabe $\boldsymbol{B} \in \mathbb{R}^{(h_b \times w_b \times k)}$ bewegt und erzeugt eine Merkmalskarte $\boldsymbol{M}^{(t)} \in \mathbb{R}^{(h_b - h_f + 1 \times w_b - w_f + 1 \times n)}$. Dabei ist w_b die Breite und h_b die Höhe der Eingabe. Die Merkmalskarte weist hohe Werte in den Bereichen der Eingabe auf, die das kodierte Muster des Filters enthalten. Konkret wird eine elementweise Multiplikation zwischen dem Filter und dessen aktueller Position auf der Eingabe berechnet und die resultierenden Werte aufsummiert. Abschließend wird eine Aktivierungsfunktion $\mathrm{h}(\cdot)$ auf die Merkmalskarten angewendet. Diese Vorgehensweise ist in Abbildung 2.3 beispielhaft dargestellt. Die Berechnung der Aktivierung an der Position i, j mit dem t-ten Filter ist wie folgt spezifiziert:

$$\boldsymbol{M}_{i,j}^{(t)} = \mathrm{h}\left(\sum_{c=0}^{k-1} \sum_{u=0}^{m-1} \sum_{v=0}^{n-1} \boldsymbol{B}_{i+u,j+v,c} \cdot \boldsymbol{K}_{u,v,c}^{(t)}\right) \tag{2.20}$$

Die Faltungsoperation hat zwei grundlegende Parameter, *Stride* und *Padding*, welche die Größe der Ausgabe beeinflussen. Das Problem der klassischen Faltung besteht darin, dass jede Faltungsoperation die Ausgabegröße in Abhängigkeit von der Filtergröße verkleinert (siehe Definition von $\boldsymbol{M}^{(t)}$). Eine intuitive Lösung für dieses Problem bietet das Padding $\boldsymbol{p} \in \mathbb{N}^2$, bei dem zusätzliche Pixel um den Rand des Eingabebildes hinzugefügt werden. In der Regel wird das Padding entsprechend der Filtergröße gewählt, sodass die Eingabe- und Ausgabegröße nach der Faltung identisch sind [250, S.249]. Der Stride $\boldsymbol{s} \in \mathbb{N}^2$ steuert die Anzahl der Einheiten in horizontaler und vertikaler Richtung, um die der Filter verschoben wird. Dabei entspricht ein Wert von 1 der bisher beschriebenen Vorgehensweise. Die Auswirkungen der beiden Parameter auf die Höhe h_m und Breite w_m der Ausgabe sind wie folgt gegeben:

$$\begin{aligned} h_m &= \left\lfloor \frac{h_b - h_f + 2 \cdot \boldsymbol{p}_0}{\boldsymbol{s}_0} \right\rfloor + 1 \\ w_m &= \left\lfloor \frac{w_b - w_f + 2 \cdot \boldsymbol{p}_1}{\boldsymbol{s}_1} \right\rfloor + 1 \end{aligned} \tag{2.21}$$

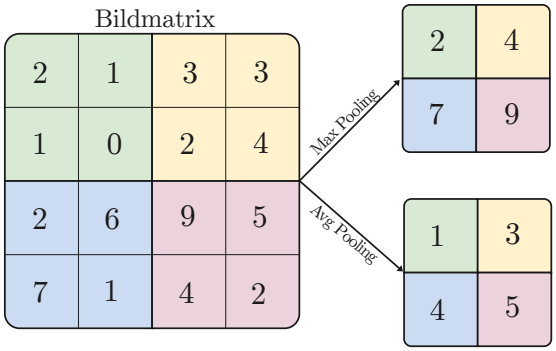


Abbildung 2.4: Eine beispielhafte Visualisierung der Pooling-Operation auf der Grundlage des Maximalwerts (Max) oder des Mittelwerts (Avg). Das Pooling wird auf die Bildmatrix mit der Größe 4×4 und einer Fenstergröße von 2×2 mit einer Schrittweite von 2 angewendet.

Neben der Faltung ist das *Pooling* die zentrale Operation in einem CNN. Hierbei werden die Pooling-Schichten zwischen zwei Faltungsschichten oder am Ende der Merkmalsextraktion angewendet. Analog zur Faltung besteht das Pooling aus einem Fenster mit fester Größe, das entsprechend der Schrittweite über die Eingabe geschoben wird. Die Pooling-Schicht hat jedoch keine Parameter, sondern berechnet in deterministischer Weise den Maximalwert oder den Mittelwert der Elemente im vorgegebenen Fenster. Dieses Verfahren ist in Abbildung 2.4 dargestellt. Bei Eingabedaten mit mehreren Merkmalskarten wird jede Karte separat verarbeitet, wodurch die Anzahl der Merkmalskarten unverändert bleibt. Das Pooling reduziert die Größe der Merkmalskarten unter Beibehaltung der wichtigsten Merkmale und ist durch den biologischen Prozess der lateralen Inhibition im visuellen Kortex motiviert [56, S.235]. Die Reduktion der Karten bewirkt generell eine Verringerung des Speicherbedarfs und des Rechenaufwandes für das CNN. Wesentlich relevanter ist jedoch die Einführung einer Translationsinvarianz durch das Pooling [250, S.257]. Dies wird durch die Zusammenfassung der Daten innerhalb eines Fensters zu einem Wert erreicht, sodass die genaue Position des Merkmals weniger relevant ist. Darüber hinaus führt die Verwendung von Pooling zu einer Vergrößerung des rezeptiven Feldes für nachfolgende Faltungsschichten.

Die Faltungs- und Pooling-Schichten sind verantwortlich für das Lernen und die Extraktion von Merkmalen aus der Eingabe, welche als Grundlage für die Entscheidungslogik im Klassifikator dienen. Für die Verarbeitung der extrahierten Merkmalskarten mit einem MLP müssen diese zunächst in einen Vektor mit fester Dimensionalität transformiert werden. Eine triviale Lösung bietet das sogenannte *Flattening* [250, S.143], bei dem die mehrdimensionalen Merkmalskarten aus der letzten Extraktionsschicht in einer vorgegebenen Reihenfolge elementweise durchlaufen und so in einen Vektor überführt werden. Dieser Ansatz basiert auf der Annahme, dass die Merkmalskarten in der letzten Schicht für jede Eingabe die gleiche Dimensionalität aufweisen. Die Ausgabegröße variiert jedoch in Abhängigkeit von der Eingabegröße, sodass die Eingaben zunächst auf eine feste Dimensionalität transformiert werden müssen. Dies führt zu einer Verzerrung der Eingabedaten, wodurch wichtige Informationen verloren gehen können [67]. Um dies zu vermeiden, wurden spezielle Pooling-Strategien entwickelt, die Eingaben variabler Größe in eine Merkmalsrepräsentation mit fester Dimensionalität transformieren [67, 68, 192]. Am gebräuchlichsten ist das *Spatial Pyramid Pooling* (SPP) [67]. Bei diesem Verfahren werden die extrahierten Merkmale des Eingabebildes auf Grundlage mehrerer hierarchischer Ebenen in eine feste Dimen-

sionalität überführt. Auf jeder Ebene wird eine festgelegte Anzahl von Pooling-Regionen verwendet, deren Filtergrößen dynamisch an die Eingabegröße angepasst werden. So entsteht eine feste Ausgabedimensionalität, die sowohl lokale als auch globale Informationen enthält.

Die klassische CNN-Architektur realisiert eine Klassifikation, wobei die Anzahl der Neuronen in der Ausgabeschicht mit der Anzahl der zu unterscheidenden Klassen k korrespondiert. Auf die Ausgabe der letzten Schicht des MLPs ($\boldsymbol{z} \in \mathbb{R}^k$) wird in der Regel eine *Softmax*-Funktion mit

$$\text{softmax}(\boldsymbol{z})_i = \frac{e^{\boldsymbol{z}_i}}{\sum_{j=0}^{k-1} e^{\boldsymbol{z}_j}}, \quad \text{für } i \in \{0, ..., k-1\} \tag{2.22}$$

angewendet [108, 183]. Diese Funktion erzeugt für die Ausgabe eine Pseudowahrscheinlichkeitsverteilung und ermöglicht damit ein interpretierbares Ergebnis.

2.4 RESIDUALE NETZWERKE

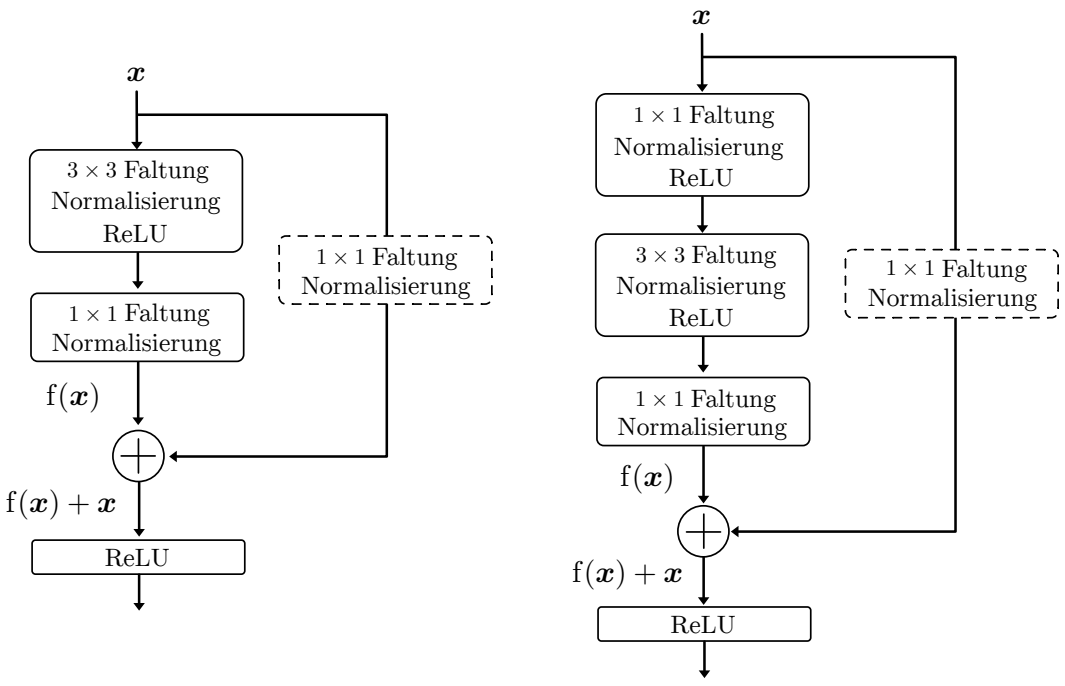
Mit der Entwicklung von tiefen Netzwerkarchitekturen hat sich die Leistung von CNNs auf den meisten Benchmarks signifikant verbessert [183, 195]. Diese Leistungssteigerung ist auf die höhere Anzahl der Parameter im Netzwerk und der Möglichkeit zur Extraktion von komplexeren hierarchischen Merkmalen zurückzuführen. Eine simple Erhöhung der Schichten im klassischen CNN-Modell führt jedoch zu dem sogenannten *Degradation Problem* [68]. Hierbei ergibt sich durch das Hinzufügen von weiteren Schichten in einem Modell ein Leistungsverlust sowohl hinsichtlich der Test- als auch der Trainingsdaten. Zur Lösung dieses Problems wird das *Residual Network* (ResNet) [68] vorgestellt. Dies ist eine spezielle CNN-Architektur, die sogenannte Skip-Verbindungen (engl.: *Skip Connections*) in die klassische Netzwerkarchitektur einbaut und damit eine effiziente Überführung der Eingabe in tiefere Schichten ermöglicht.

Das ResNet basiert auf der theoretischen Überlegung, dass das Hinzufügen von Schichten zu einem neuronalen Modell die Leistung entweder verbessern oder zumindest nicht verschlechtern sollte. Dies kann durch ein Gedankenexperiment veranschaulicht werden, bei dem das tiefere Netzwerk die Gewichte des kleineren Modells übernimmt und für die neuen Schichten die Identitätsfunktion $\mathrm{g}(\boldsymbol{x}) = \boldsymbol{x}$ verwendet. In der Praxis hat sich jedoch gezeigt, dass das Lernen der Identitätsfunktion mit den klassischen Faltungsschichten nicht trivial ist [68]. Vor diesem Hintergrund wird in [68] die Hypothese aufgestellt, dass es bei tiefen Netzwerken einfacher ist, die Faltungen mit der Formel

$$\mathrm{h}(\boldsymbol{x}) = \mathrm{f}(\boldsymbol{x}) + \boldsymbol{x} \tag{2.23}$$

zu realisieren. Hierbei ist $\mathrm{h}(\boldsymbol{x})$ die gewünschte zu lernende Ausgabefunktion, $\boldsymbol{x}$ die Eingabe und $\mathrm{f}(\boldsymbol{x})$ die sogenannte Residualfunktion (engl.: *Residual Function*). Für die Identitätsabbildung muss $\mathrm{f}(\boldsymbol{x})$ nur die Nullfunktion erzeugen, die für ein neuronales Modell wesentlich einfacher zu bilden ist als die Approximation einer Identitätsabbildung mit einem Block nichtlinearer Schichten.

Das ResNet realisiert dieses Konzept durch die Verwendung residualer Blöcke. Die Architektur dieser Blöcke (siehe Abbildung 2.5a) basiert auf einer Kombination von klassischen Faltungsschichten und einer Skip-Verbindung. Dabei ist die Ausgabe der Faltungsschichten eine Umsetzung der Residualfunktion $\mathrm{f}(\boldsymbol{x})$ aus Formel 2.23. Konkret wird $\mathrm{f}(\boldsymbol{x})$ mit zwei 3×3 Faltungsschichten realisiert. Die Ausgabe jeder Schicht wird mit der Batch-Normalisierung [80] auf einen Mittelwert von Null und eine Einheitsvarianz transformiert.



(a) Standardblock (b) *Bottleneck*-Block

Abbildung 2.5: Eine Visualisierung der residualen Blöcke im ResNet-Modell für die Eingabe $\boldsymbol{x}$. Der gestrichelte Block in der Skip-Verbindung wird nur bei unterschiedlicher Dimensionalität von $\boldsymbol{x}$ und $\mathrm{f}(\boldsymbol{x})$ verwendet. Der Bottelneck-Block wird normalerweise nur in Modellen mit einer Tiefe von 50 Faltungsschichten oder mehr eingesetzt.

Diese Normalisierung führt generell zu einem stabileren und effizienteren Training [80]. Zudem wird die ReLU-Funktion auf die Ausgabe der ersten Schicht und auf das Ergebnis der Skip-Verbindung angewendet.

Die ResNet-Architektur basiert analog zu einem CNN auf einer Merkmalsextraktion und einem MLP. Die Merkmalsextraktion besteht aus einer initialen 7×7 Faltungsschicht und einer 3×3 *Max-Pooling*-Schicht mit jeweils einer Schrittweite von zwei. Das Ergebnis sind 64 Merkmalskarten, die als Eingabe für vier logische Einheiten mit einer variablen Anzahl von gestapelten residualen Blöcken dienen. An den Übergängen zwischen den Einheiten wird die Größe der Merkmalskarten mit der Max-Pooling-Operation und einer Schrittweite von zwei halbiert. Außerdem wird die Anzahl der Merkmalskarten verdoppelt, sodass $\boldsymbol{x}$ und $\mathrm{f}(\boldsymbol{x})$ unterschiedliche Dimensionalitäten aufweisen. Für die elementweise Addition bei der Skip-Verbindung müssen die beiden Darstellungen jedoch die gleiche Dimensionalität haben. Daher wird $\boldsymbol{x}$ an die geringere Dimensionalität von $\mathrm{f}(\boldsymbol{x})$ durch eine 1×1 Faltung mit einer Schrittweite von zwei angepasst und die Anzahl der Merkmalskarten entsprechend erhöht. Für die Verarbeitung der Merkmalskarten mit einem MLP müssen diese in eine Vektorrepräsentation mit fester Dimensionalität überführt werden. Beim ResNet-Modell wird dazu das *Global Average Pooling* (GAP)-Verfahren [117] verwendet, welches den Mittelwert jeder Merkmalskarte bildet und damit eine Ausgabe entsprechend der Anzahl der Merkmalskarten in der letzten Schicht erzeugt. Der Vektor dient als Eingabe für ein MLP, welches aus einer Ein- und Ausgabeschicht besteht. Die Ausgabedimensionalität $d \in \mathbb{N}$ des Modells kann beliebig angepasst werden.

Die Anzahl der Blöcke kann pro Einheit variabel festgelegt werden. In der Literatur haben sich jedoch bestimmte Kombinationen etabliert (z.B. *ResNet34* und *ResNet50*) [68].

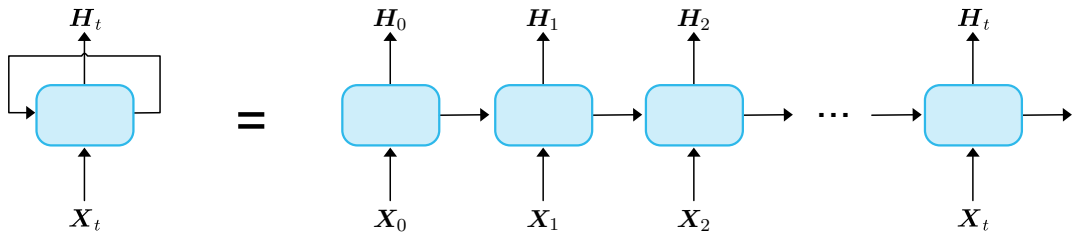


Abbildung 2.6: Eine Visualisierung der klassischen RNN-Architektur mit einem zyklischen Eingang und die Überführung des Netzwerks in eine azyklische Architektur für eine gegebene Sequenz $\boldsymbol{X}$ bis zum Zeitpunkt t.

Aufgrund der erhöhten Trainingszeit und des hohen Speicherbedarfs von Modellen mit mehr als 50 Blöcken wird für diese Modelle die Verwendung eines *Bottlenecks* (siehe Abbildung 2.5b) anstelle des Standardblocks verwendet. Die Grundidee besteht darin, die residualen Blöcke möglichst flach zu halten und die Anzahl der Parameter und Matrixmultiplikationen zu verringern. Dabei hat der Bottleneck- im Gegensatz zum Standardblock nicht mehr zwei 3×3 Faltungen, sondern besteht aus einer Kombination aus 1×1 und 3×3 Faltungen. Die 1×1 Schichten sind für die Reduzierung und anschließende Wiederherstellung der Dimensionen verantwortlich, wodurch die 3×3 Schicht mit einer reduzierten Anzahl an Merkmalskarten arbeitet.

2.5 REKURRENTE NETZWERKE

Ein rekurrentes neuronales Netzwerk (engl.: *Recurrent Neural Network* (RNN)) [168] ist eine spezielle neuronale Architektur zur Verarbeitung sequentieller Daten. Das Modell basiert auf einem rekursiven Ansatz, bei dem die Eingabedaten sequentiell verarbeitet werden und die Ausgabe zum Zeitpunkt t als zusätzliche Eingabe für den nächsten Zeitschritt $t+1$ dient. Auf diese Weise fließen Informationen aus früheren Zeitschritten in die aktuelle Berechnung ein. Trotz der Einführung dieser zyklischen Verknüpfung lässt sich ein RNN als tiefes Feedforward-Netzwerk auffassen [47, p. 329]. Dies ist in Abbildung 2.6 illustriert. Die zyklische Architektur wird in ein Modell ohne Zyklen überführt, indem das RNN-Modul für jedes Element der Eingabesequenz repliziert wird. Die replizierten Modelle teilen sich die Gewichte und sind daher identisch. Formal berechnet ein RNN für eine Eingabe $\mathbf{E} = \{\mathbf{e}^{(0)}, \ldots, \mathbf{e}^{(n-1)}\}$ mit variabler Länge n eine Ausgabe $\boldsymbol{H} \in \mathbb{R}^{(n \times d)}$. Dabei ist d ein frei wählbarer Parameter, der die Anzahl der verborgenen Schichten im Modell und damit die Ausgabedimensionalität festlegt. Zur Verarbeitung der Eingabesequenz $\mathbf{E}$ wird diese zunächst in eine Matrix $\boldsymbol{X} \in \mathbb{R}^{(n \times d)}$ umgewandelt. Es existieren verschiedene Varianten von RNNs, wobei das klassische Modell die Ausgabe $\boldsymbol{H}_t$ zum Zeitpunkt $t \in \{0, \ldots, n-1\}$ mit

$$\boldsymbol{H}_t = \tanh(\underbrace{(\boldsymbol{H}_{t-1} \parallel \boldsymbol{X}_t)^\intercal}_{=\boldsymbol{K}_t} \cdot \boldsymbol{W}) \tag{2.24}$$

berechnet. Hierbei wird die Eingaberepräsentation $\boldsymbol{X}_t$ und die vorherige Ausgabe des Modells $\boldsymbol{H}_{t-1}$ konkateniert und transponiert, wodurch die Zwischenrepräsentation $\boldsymbol{K}_t \in \mathbb{R}^{(1 \times 2 \cdot d)}$ entsteht. Anschließend wird auf diese die lernbare Gewichtsmatrix $\boldsymbol{W} \in \mathbb{R}^{(2 \cdot d \times d)}$ und die nichtlineare *Tangens Hyperbolicus* (tanh)-Aktivierungsfunktion angewendet.

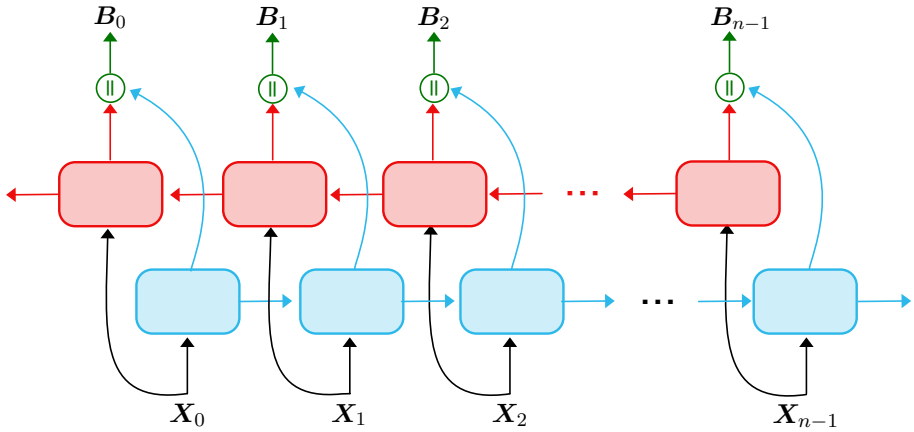


Abbildung 2.7: Der Aufbau eines bidirektionalen RNNs für die Eingabe $\boldsymbol{X}$ mit einer Länge von n. Das Modell besteht aus einem unidirektionalen Vorwärtsmodell (blau) und einem Rückwärtsmodell (rot). Die Ausgaben der Modelle werden in jedem Zeitschritt miteinander konkateniert und ergeben die Ausgabe $\boldsymbol{B}$.

Die Verarbeitung der Eingabesequenz $\mathbf{E}$ erfolgt bei klassischen RNNs unidirektional. Damit basiert die Vorhersage zum Zeitpunkt t auf den Eingabedaten $\{\mathbf{e}^{(0)}, \ldots, \mathbf{e}^{(t-1)}\}$. Es gibt jedoch Anwendungen, wie z.B. die Sprachverarbeitung, bei denen für ein Zeitpunkt nicht nur vergangene Eingaben sondern auch zukünftige vorliegen [127]. Modelle, die auf der gesamten Eingabesequenz basieren, werden als bidirektionale RNNs [176] bezeichnet und bestehen aus zwei unidirektionalen RNNs. Der Aufbau und die Vorgehensweise dieser bidirektionalen Modelle ist in Abbildung 2.7 illustriert. Im Folgenden werden die beiden unidirektionalen Modelle als Vorwärts- und Rückwärtsmodell bezeichnet. Beim Vorwärtsmodell erfolgt die Vorhersage zum Zeitpunkt t auf der Eingabe $\{\mathbf{e}^{(0)}, \ldots, \mathbf{e}^{(t-1)}\}$ und beim Rückwärtsmodell auf der Eingabe $\{\mathbf{e}^{(t+1)}, \ldots, \mathbf{e}^{(n-1)}\}$. Das Vorwärtsmodell erzeugt die Ausgabe $\boldsymbol{V} \in \mathbb{R}^{(n \times d)}$ und das Rückwärtsmodell die Ausgabe $\boldsymbol{R} \in \mathbb{R}^{(n \times d)}$. Die beiden Modelle interagieren nicht direkt miteinander, sondern erzeugen durch eine konkatenierte Darstellung der Ausgaben zu jedem Zeitschritt eine gemeinsame Ausgabe $\boldsymbol{B} \in \mathbb{R}^{(n \times 2 \cdot d)}$. Dies ist für den Zeitpunkt t wie folgt spezifiziert:

$$\boldsymbol{B}_t = \boldsymbol{V}_t \parallel \boldsymbol{R}_t \tag{2.25}$$

Das Training der RNNs erfolgt mit dem *Backpropagation-Through-Time*-Algorithmus [230]. Aufgrund der Kombination von rekurrenten Verbindungen, der meist langen Sequenzlängen der Eingabedaten und der tanh-Aktivierungsfunktion kann es beim Training zum *Vanishing* bzw. *Exploding Gradient*-Problem kommen. Dies erschwert die Anpassung der Parameter für die ersten Schichten des RNNs bzw. führt zu einem instabilen Lernverhalten. Insbesondere das erste Problem macht es dem RNN schwer, langfristige Abhängigkeiten in den Daten zu erfassen. Um diesem Problem entgegenzuwirken, hat sich die Verwendung von speziellen rekurrenten neuronalen Netzwerken wie dem LSTM [71] oder den GRUs [32] etabliert. Diese beiden Architekturen basieren auf einem ähnlichen Prinzip wie das klassische RNN, ermöglichen jedoch eine robuste Modellierung langfristiger Abhängigkeiten durch sogenannte *Gates*. Diese Gates steuern den Informationsfluss im Netzwerk, wobei in jedem Zeitschritt der interne Zustand des Modells aktualisiert wird.

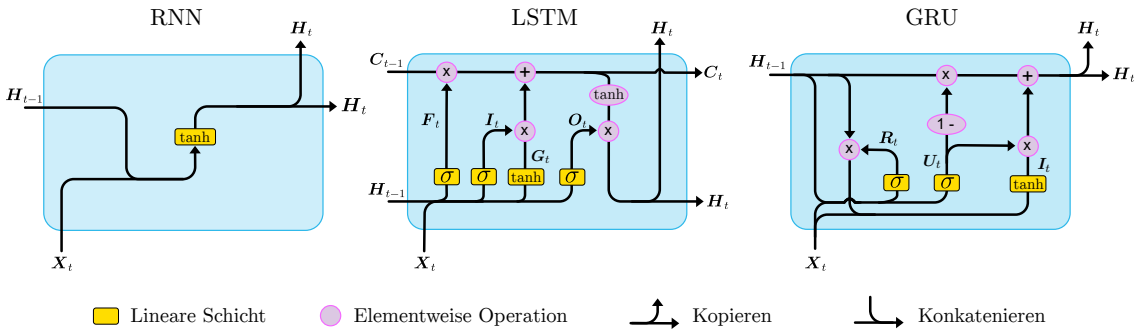


Abbildung 2.8: Ein Vergleich der Netzwerkstrukturen und Berechnungen in RNN-, LSTM- und GRU-Modellen. Die Sigmoid-Aktivierungsfunktion ist durch das Symbol σ repräsentiert.

2.5.1 *Long Short-Term Memory*

Das *Long Short-Term Memory* (LSTM) [71] ist eine spezielle Variante des RNNs. Im Gegensatz zum klassischen RNN wird das Modell um einen internen Zustand erweitert, der speziell zur Kodierung von langfristigen Abhängigkeiten konzipiert wurde. Ein Überblick über die Architektur des Modells ist in Abbildung 2.8 gegeben. Die Eingabe des Modells zum Zeitpunkt t besteht aus dem aktuellen Element der Eingabesequenz $\boldsymbol{X}_t \in \mathbb{R}^d$, der Ausgabe des vorhergehenden Zeitschrittes $\boldsymbol{H}_t \in \mathbb{R}^d$ und dem vorherigen Zustand $\boldsymbol{C}_{t-1} \in \mathbb{R}^d$. Der allgemeine Ablauf eines LSTM-Blocks kann in drei Schritte unterteilt werden. Zuerst werden Informationen aus dem internen Zustand entfernt, anschließend werden dem Zustand neue Informationen hinzugefügt und am Ende wird die Ausgabe des Modells für den aktuellen Zeitschritt erzeugt. Zur Realisierung dieser Vorgänge basiert das LSTM auf den sogenannten *Forget-*, *Input-* und *Output-Gates*. Diese Gates aktualisieren selektiv die Informationen des internen Zustandes und steuern die Ausgabe des Modells. Hierbei erzeugen die Gates auf der Basis der vorherigen Ausgabe $\boldsymbol{H}_{t-1}$ und der aktuellen Eingabe $\boldsymbol{X}_t$ eine individuelle Repräsentation $\boldsymbol{F}_t$, $\boldsymbol{I}_t$ bzw. $\boldsymbol{O}_t \in [0, \ldots, 1]^d$. Für das Forget-Gate ist dies mit

$$\boldsymbol{F}_t = \text{sigmoid}\left(\boldsymbol{K}_t \cdot \boldsymbol{W}^{(f)}\right) \tag{2.26}$$

spezifiziert. Dabei wird $\boldsymbol{K}_t$, welche die konkatenierte Darstellung von $\boldsymbol{H}_{t-1}$ und $\boldsymbol{X}_t$ repräsentiert, durch eine lineare Schicht $\boldsymbol{W}^{(f)} \in \mathbb{R}^{(2 \cdot d \times d)}$ mit anschließender Sigmoid-Aktivierungsfunktion verarbeitet. In diesem Zusammenhang stehen $\boldsymbol{F}_t$ und $\boldsymbol{C}_t$ in Beziehung zueinander. Konkret bewirkt der Wert von $\boldsymbol{F}_{t,i}$ nahe Null, dass die Information an der Position i des Zustands gelöscht wird, während sie bei einem Wert nahe Eins erhalten bleibt. Intuitiv bestimmt das Forget-Gate auf diese Weise, welche Informationen im Zustand gelöscht werden sollen.

Im zweiten Schritt werden mit dem Input-Gate und einer Kandidatenrepräsentation $\boldsymbol{G} \in \mathbb{R}^d$ neue Informationen für den Zustand ermittelt. Dabei wird zunächst die Ausgabe des Input-Gates $\boldsymbol{I}$ analog zum Forget-Gate mit

$$\boldsymbol{I}_t = \text{sigmoid}\left(\boldsymbol{K}_t \cdot \boldsymbol{W}^{(i)}\right) \tag{2.27}$$

berechnet. Für die Erzeugung der Kandidatenrepräsentation wird eine lineare Schicht $\boldsymbol{W}^{(g)} \in \mathbb{R}^{(2\cdot d\times d)}$ mit einer tanh-Aktivierungsfunktion auf die konkatenierte Darstellung der vorherigen Ausgabe des Modells und der aktuellen Eingabe angewendet:

$$\boldsymbol{G}_t = \tanh\left(\boldsymbol{K}_t \cdot \boldsymbol{W}^{(g)}\right) \tag{2.28}$$

Der neue Zustand $\boldsymbol{C}_t$ ergibt sich schließlich aus der Multiplikation des alten Zustands mit der Ausgabe des Forget-Gates und der Addition der Kandidatenrepräsentation mit dem Input-Gate:

$$\boldsymbol{C}_t = \boldsymbol{F}_t \odot \boldsymbol{C}_{t-1} + \boldsymbol{I}_t \odot \boldsymbol{G}_t \tag{2.29}$$

Die Ausgabe des LSTMs basiert auf dem aktuellen Zustand $\boldsymbol{C}_t$ und der Ausgabe des Output-Gates $\boldsymbol{O}_t$. Dabei ist das Output-Gate analog zum Forget- und Input-Gate durch

$$\boldsymbol{O}_t = \text{sigmoid}\left(\boldsymbol{K}_t \cdot \boldsymbol{W}^{(o)}\right) \tag{2.30}$$

definiert. Intuitiv modelliert das Gate, welche Informationen aus dem aktuellen Zustand relevant für die Ausgabe sind. Die Ausgabe des Modells zum Zeitpunkt t wird letztlich mit

$$\boldsymbol{H}_t = \boldsymbol{O}_t \odot \tanh(\boldsymbol{C}_t) \tag{2.31}$$

berechnet. Dazu wird der Zustand $\boldsymbol{C}_t$ elementweise mit einer tanh-Funktion verarbeitet und mit der Ausgabe des Output-Gates multipliziert.

2.5.2 *Gated Recurrent Units*

Das *Gated Recurrent Unit* (GRU)-Modell [32] bietet eine vereinfachte Version der LSTM-Architektur. Die Unterschiede zwischen den beiden Modellen sind in Abbildung 2.8 visuell ersichtlich. Analog zum klassischen RNN besitzen GRUs einen einzigen Ausgabevektor $\boldsymbol{H}_t \in \mathbb{R}^d$, der als eine Kombination aus der Ausgabe $\boldsymbol{H}_t$ und dem Zustand $\boldsymbol{C}_t$ des LSTMs aufgefasst werden kann. Anstelle der drei in LSTMs verwendeten Gates wird der Informationsfluss in GRUs durch das *Reset-* und das *Update-Gate* gesteuert. Aufgrund der Kodierung des Zustands und der Ausgabe in einer Repräsentation ist das Output-Gate nicht nötig. Zudem ist das Update-Gate eine Art Kombination der Forget- und Input-Gates aus dem LSTM. Dies ermöglicht ein einfacheres Training und eine schnellere Ausführung im Vergleich zu LSTMs [33]. Formal werden die Ausgaben der Gates analog zum LSTM über die linearen Schichten $\boldsymbol{W}^{(u)}$ und $\boldsymbol{W}^{(r)} \in \mathbb{R}^{(2\cdot d\times d)}$ mit anschließender Sigmoid-Aktivierungsfunktion berechnet. Dies ist für das Update-Gate $\boldsymbol{U}_t \in [0,\ldots,1]^d$ und das Reset-Gate $\boldsymbol{R}_t \in [0,\ldots,1]^d$ zum Zeitpunkt t mit

$$\begin{aligned} \boldsymbol{U}_t &= \text{sigmoid}\left(\boldsymbol{K}_t \cdot \boldsymbol{W}^{(u)}\right) \\ \boldsymbol{R}_t &= \text{sigmoid}\left(\boldsymbol{K}_t \cdot \boldsymbol{W}^{(r)}\right) \end{aligned} \tag{2.32}$$

definiert. In Analogie zum LSTM wird eine Kandidatenrepräsentation $\boldsymbol{A}_t \in \mathbb{R}^d$ für den neuen Zustand mit

$$\boldsymbol{A}_t = \tanh\left(((\boldsymbol{R}_t \odot \boldsymbol{H}_{t-1}) \parallel \boldsymbol{X}_t) \cdot \boldsymbol{W}^{(h)}\right) \tag{2.33}$$

erzeugt. Dazu wird zunächst eine elementweise Multiplikation des Reset-Gates $\boldsymbol{R}_t$ mit der vorherigen Ausgabe $\boldsymbol{H}_{t-1}$ durchgeführt, wodurch ausgewählte Informationen aus dem

Zustand entfernt werden. Die resultierende Repräsentation wird mit der aktuellen Eingabe $\boldsymbol{X}_t$ konkateniert und mit einer linearen Schicht $\boldsymbol{W}^{(h)} \in \mathbb{R}^{(2 \cdot d \times d)}$ sowie der tanh-Aktivierungsfunktion in die Kandidatenrepräsentation überführt. Die Ausgabe für den Zeitschritt t wird mit

$$\boldsymbol{H}_t = (1 - \boldsymbol{U}_t) \odot \boldsymbol{H}_{t-1} + \boldsymbol{U}_t \odot \boldsymbol{A}_t \tag{2.34}$$

berechnet. Dazu wird die vorherige Ausgabe $\boldsymbol{H}_{t-1}$ und die Kandidatenrepräsentation $\boldsymbol{A}_t$ anhand der Ausgabe des Update-Gates $\boldsymbol{U}_t$ gewichtet kombiniert.

2.6 TRANSFORMER

Ein Transformer [218] ist eine neuronale Netzwerkarchitektur zur Verarbeitung sequentieller Daten. Die Architektur basiert nicht auf Rekursionen oder Faltungen sondern verwendet stattdessen den *Attention*-Mechanismus [11]. Dieser ermöglicht im Gegensatz zu sequentiellen Ansätzen eine parallele Verarbeitung der gesamten Eingabesequenz und die Kodierung langfristiger Abhängigkeiten zwischen den Eingabedaten. Generell basiert die Transformer-Architektur auf dem *Encoder-Decoder*-Paradigma. Bei diesem Verfahren bildet der Encoder eine gegebene Eingabesequenz auf eine Folge kontextsensitiver Vektordarstellungen ab und der Decoder generiert auf dieser Basis autoregressiv eine Ausgabesequenz. Die in dieser Arbeit vorgestellten Transformer-Modelle fokussieren sich auf das Textverständnis und nicht auf die Textgenerierung und nutzen daher nur den Encoder aus der Architektur. Aus diesem Grund beschränkt sich die Beschreibung des Transformers im Folgenden ausschließlich auf den Encoder, der in Abbildung 2.9 visualisiert ist.

Im ersten Schritt wird die Eingabesequenz $\mathbf{E} = \{\mathbf{e}^{(0)}, \ldots, \mathbf{e}^{(l-1)}\}$ in eine Repräsentation $\boldsymbol{X} \in \mathbb{R}^{(l \times d)}$ umgewandelt. Der Encoder basiert im Wesentlichen auf dem Attention-Mechanismus, der eine positionsunabhängige Verarbeitung der Sequenzdaten durchführt [218]. Da die Positionsangaben in der Regel relevante Informationen enthalten, hat sich die Kodierung der relativen oder absoluten Positionsdaten in $\boldsymbol{X}$ bewährt [218]. Dazu werden die Positionsinformationen aus der Sequenz in eine Repräsentation $\boldsymbol{P} \in \mathbb{R}^{(l \times d)}$ überführt, die auf Sinus- und Kosinusfunktionen mit verschiedenen Frequenzen basiert [218]. Formal wird die Repräsentation $\boldsymbol{P}_{i,j}$ für die i-te Position und der Dimension $j \in \{0, \ldots, d-1\}$ mit

$$\boldsymbol{P}_{i,j} = \begin{cases} \sin\left(\frac{i}{10000^{j/d}}\right) & \text{wenn } j \equiv 0 \pmod 2 \\ \cos\left(\frac{i}{10000^{(j-1)/d}}\right) & \text{sonst} \end{cases} \tag{2.35}$$

bestimmt. Anschließend werden die Positionskodierungen $\boldsymbol{P}$ mit den Eingabevektoren $\boldsymbol{X}$ elementweise addiert und dienen als Eingabe für das Encoder-Modell. Dieses besteht aus einer stapelweise aufgebauten Architektur von n identischen Modulen. Jedes dieser Module setzt sich aus einer sequentiellen Anwendung der sogenannten *Multi-Head-Attention* [218] und einem MLP zusammen. Das MLP besteht aus zwei linearen vollvernetzten Schichten mit einer dazwischen liegenden ReLU-Aktivierungsfunktion und wird elementweise auf die Ausgabe der Attention-Schicht angewendet. Für ein stabileres und schnelleres Training werden analog zum ResNet residuale Verbindungen mit anschließender Normalisierung der Repräsentationen für die Komponenten verwendet [118, S.3]. Die Ausgabe des Encoders ist eine kontextsensitive Darstellung der Eingabedaten mit der gleichen Sequenzlänge.

Der Attention-Mechanismus stellt die zentrale Komponente des Transformers dar und wurde ursprünglich entwickelt, um ein grundlegendes Problem von sequentiellen Encoder-

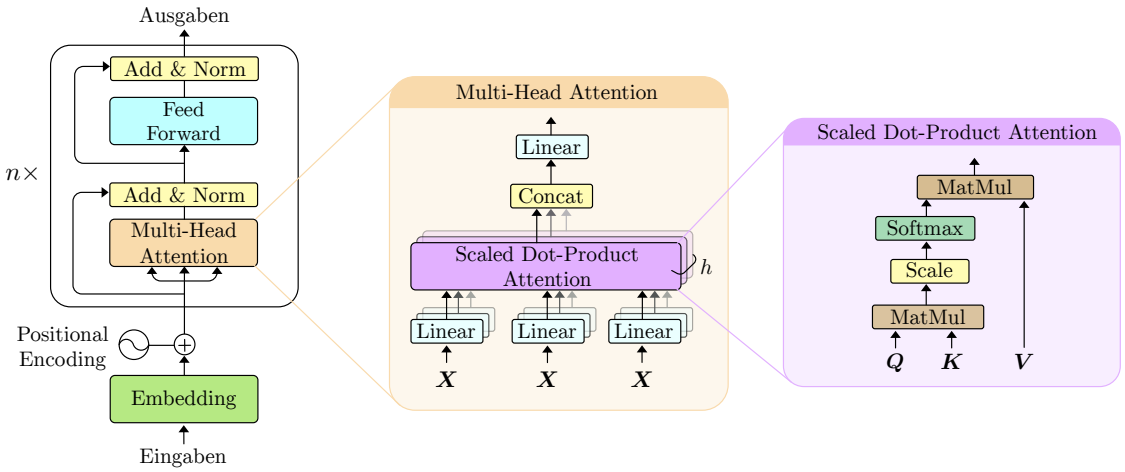


Abbildung 2.9: Eine Übersicht über das Encoder-Modell aus der Transformer-Architektur mit einer detaillierten Darstellung der wichtigsten Module. Die Grafik ist angelehnt an [218].

Decoder-Modellen zu lösen [11]. Hierbei ist die sequentielle Umwandlung der Encoder-Ausgaben in einen Kontextvektor insbesondere bei langen und komplexen Sequenzen problematisch, da dabei wichtige Informationen verloren gehen können [11]. Die Grundidee des Attention-Mechanismus besteht darin, den Kontextvektor nur aus relevanten Eingabedaten zu bilden, sodass auch bei langen Sequenzen die wesentlichen Informationen zur Dekodierung verfügbar sind [11]. Dazu wird in jedem Dekodierschritt die Relevanz jeder Encoder-Ausgabe bestimmt und der Kontextvektor mit einer nach Relevanz gewichteten Summe der Encoder-Ausgaben gebildet. Das Encoder-Modell basiert auf einem Spezialfall des Attention-Mechanismus, der sogenannten *Self-Attention* [28]. Hierbei besteht die Eingabe im Vergleich zum Encoder-Decoder-Modell nur aus einer Sequenz. Das Ziel ist die Modellierung der Zusammenhänge zwischen den Elementen innerhalb der Sequenz und die Ausgabe einer kontextsensitiven Repräsentation der Eingabedaten. Die Berechnungen der Self-Attention werden formal aus der verallgemeinerten Attention-Formel [218] abgeleitet, die durch

$$\text{attn}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{softmax}(\frac{\boldsymbol{Q} \cdot \boldsymbol{K}^\intercal}{\sqrt{d_k}}) \cdot \boldsymbol{V} \tag{2.36}$$

definiert ist. Die Formel basiert auf den sogenannten *Query-*, *Key-* und *Valuerepräsentationen*, welche bei der Self-Attention durch die Matrizen $\boldsymbol{Q} \in \mathbb{R}^{(l \times d_k)}$, $\boldsymbol{K} \in \mathbb{R}^{(l \times d_k)}$ und $\boldsymbol{V} \in \mathbb{R}^{(l \times d_v)}$ repräsentiert sind. In einem ersten Schritt werden die d-dimensionalen Elemente der Sequenz $\boldsymbol{X}$ mit den lernbaren Transformationsmatrizen $\boldsymbol{W}^{(q)} \in \mathbb{R}^{(d \times d_k)}$, $\boldsymbol{W}^{(k)} \in \mathbb{R}^{(d \times d_k)}$ und $\boldsymbol{W}^{(v)} \in \mathbb{R}^{(d \times d_v)}$ zu $\boldsymbol{Q}$, $\boldsymbol{K}$ und $\boldsymbol{V}$ transformiert. Anschließend werden auf Grundlage der Repräsentationen die sogenannten *Alignment*-Werte mit dem Skalarprodukt zwischen $\boldsymbol{Q}$ und $\boldsymbol{K}$ berechnet. Zur Stabilisierung des Trainings werden diese mit dem Skalierungsfaktor $\sqrt{d_k}$ normiert [218]. Die zeilenweise Anwendung der Softmax-Funktion auf die skalierten Alignment-Werte bestimmt die Attention-Gewichte, welche für jedes Element der Sequenz die Relevanz zu den anderen Elementen der Sequenz durch eine Pseudowahrscheinlichkeitsverteilung repräsentiert. Schließlich wird für jedes Element der Eingabe eine kontextabhängige Repräsentation berechnet, die sich aus der gewichteten Summe von $\boldsymbol{V}$ bezüglich der Attention-Gewichte ergibt.

Bei der Multi-Head-Attention werden verschiedene Arten von Beziehungen zwischen den Eingabedaten modelliert. Hierbei werden h verschiedene Query-, Key- und Valuerepräsentationen auf Basis der Eingabesequenz $\boldsymbol{X}$ mit lernbaren linearen Projektionen

$\boldsymbol{W}^{(q_i)} \in \mathbb{R}^{(d \times d_k)}, \boldsymbol{W}^{(k_i)} \in \mathbb{R}^{(d \times d_k)}$ und $\boldsymbol{W}^{(v_i)} \in \mathbb{R}^{(d \times d_v)}$ erzeugt, wobei $i \in \{0, \ldots, h-1\}$ ist. Für jede dieser projizierten Versionen wird die Attention-Formel simultan ausgeführt, wodurch sich für die i-te Projektion die Darstellung $\boldsymbol{H}^{(i)} \in \mathbb{R}^{(l \times d_v)}$ mit

$$\boldsymbol{H}^{(i)} = \text{attn}(\boldsymbol{X} \cdot \boldsymbol{W}^{(q_i)}, \boldsymbol{X} \cdot \boldsymbol{W}^{(k_i)}, \boldsymbol{X} \cdot \boldsymbol{W}^{(v_i)}) \tag{2.37}$$

ergibt. Diese Repräsentationen werden konkateniert und mit einer lernbaren linearen Projektion $\boldsymbol{O} \in \mathbb{R}^{(h \cdot d_v \times d_v)}$ zu

$$\boldsymbol{M} = (\boldsymbol{H}^{(0)} || \ldots || \boldsymbol{H}^{(h-1)}) \cdot \boldsymbol{O} \tag{2.38}$$

transformiert. Die daraus resultierende Matrix $\boldsymbol{M} \in \mathbb{R}^{(l \times d_v)}$ repräsentiert eine kontextsensitive Darstellung für jedes Element der Eingabedaten unter Berücksichtigung vielfältiger Abhängigkeiten.

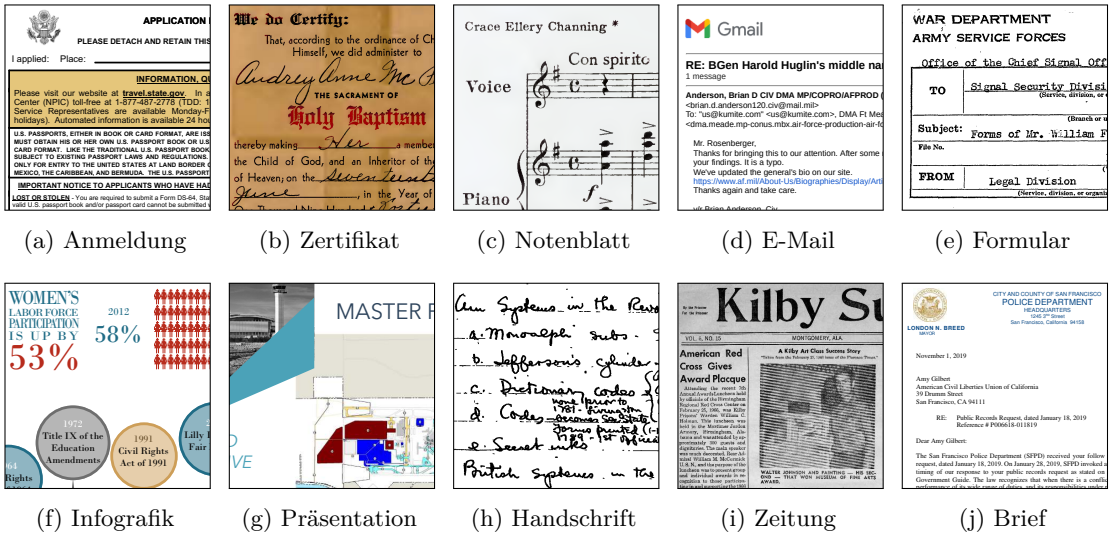
3 SEMANTISCHE DOKUMENTENBILDANALYSE

Nachdem im Kapitel 2 die relevanten methodischen Grundlagen erarbeitet wurden, werden in diesem Kapitel die anwendungsspezifischen Grundlagen dieser Arbeit vorgestellt. Die vorliegende Arbeit beschäftigt sich mit der automatischen Verarbeitung und dem inhaltlichen Verständnis von natürlichsprachlichen Texten in Dokumentenbildern. Dabei vereint die vorgestellte Methodik Komponenten der Dokumentenbildanalyse (siehe Abschnitt 3.1) und des NLP (siehe Abschnitt 3.2). Die Arbeit ist dem interdisziplinären Forschungsgebiet der semantischen Dokumentenbildanalyse zuzuordnen. In diesem Forschungsbereich existieren zahlreiche Anwendungsfälle, wobei in dieser Arbeit der Fokus auf relevante Anwendungen aus digitalen Bibliotheken gelegt wird. Von besonderem Interesse sind in diesem Zusammenhang die semantische Schlüsselwortsuche, die Extraktion von NEs und die Beantwortung von Fragen auf der Basis von Dokumentenbildern. Die genannten Anwendungen werden im Abschnitt 3.3 zusammen mit einem detaillierten Literaturüberblick vorgestellt. Im Abschnitt 3.4 wird der aktuelle Stand der Forschung auf dem Gebiet der semantischen Dokumentenbildanalyse zusammengefasst und der methodische Ansatz dieser Arbeit motiviert.

3.1 DOKUMENTENBILDANALYSE

Die Dokumentenbildanalyse (engl.: *Document Image Analysis*) ist ein Forschungsgebiet, das sich mit der elektronischen Verarbeitung von Dokumentenbildern beschäftigt. In dieser Arbeit wird ein Dokument als ein physischer Informationsträger definiert, der Schrift enthält. Dokumente dienen der Erfassung, Speicherung, Übermittlung und Archivierung von Informationen. Ein Dokumentenbild ist eine digitale Repräsentation eines physischen Dokuments, wie z.B. Papier. Die Digitalisierung kann unter anderem durch einen Scanner oder eine Kamera erfolgen. Das Dokumentenbild basiert auf einer dreidimensionalen Matrix aus Höhe, Breite und Farbkanälen, deren Matrixelemente als Pixel bezeichnet werden. Ein Pixel ist die elementare Komponente eines Bildes, die eine Farbe bzw. einen Intensitätswert repräsentiert [250, S.5]. In einem Farbbild wird jedes Pixel durch einen Farbraum kodiert. Eine der bekanntesten Verfahren ist der additive RGB-Farbraum, in dem drei ganzzahlige Intensitätswerte im Intervall von 0 bis 255 für die Grundfarben Rot, Grün und Blau kodiert werden [250, S.5].

Das Ziel der Dokumentenbildanalyse ist die Extraktion von Informationen jeglicher Art aus Bildern von Dokumenten, um diese weiter zu verarbeiten oder zu verstehen. Die Analyse von Dokumentenbildern bietet zahlreiche Anwendungsfelder [100, 133, 185, 206, 233] und erfordert die Verarbeitung diverser Arten von Dokumenten mit individuellen Herausforderungen [103]. Aufgrund der vielfältigen Anwendungsgebiete und Problemstellungen existiert kein einheitlicher Ansatz zur Analyse von Dokumentenbildern. Dennoch umfassen alle Anwendungsbereiche Ansätze aus der Bildverarbeitung und der Mustererkennung. Moderne Analyseverfahren basieren auf dem maschinellen Lernen, insbesondere dem DL [119]. Hierbei hat sich die Verwendung von Faltungsnetzwerken oder Transformer-Modellen zur automatischen Extraktion von Merkmalen aus den Bildern etabliert [87, 114, 193]. Im Folgenden werden relevante Dokumentenarten und Anwendungsbereiche vorgestellt. Die



(a) Anmeldung (b) Zertifikat (c) Notenblatt (d) E-Mail (e) Formular

(f) Infografik (g) Präsentation (h) Handschrift (i) Zeitung (j) Brief

Abbildung 3.1: Eine Auswahl verschiedener Dokumententypen und -formate im Bereich der Dokumentenbildanalyse. Die Struktur und Dokumentenbilder stammen aus [103].

Übersicht erhebt keinen Anspruch auf Vollständigkeit, sondern beschränkt sich auf die Aspekte dieser Arbeit. Für einen detaillierten Überblick über das Forschungsfeld der Dokumentenbildanalyse siehe [45].

Dokumentarten und Herausforderungen

Ein Dokumentenbild kann eine Vielzahl von Formaten aufweisen, wie beispielsweise Textdokumente, Formulare, Verträge oder Zeitungen. Die Dokumente besitzen häufig eine hohe visuelle Komplexität, die auf anspruchsvollen *Layouts* und einer Kombination verschiedener Inhalte wie z.B. Texte, Grafiken und Tabellen zurückzuführen ist [103]. Zur Veranschaulichung der Variabilität von unterschiedlichen Dokumentarten ist in Abbildung 3.1 eine Auswahl von Beispielen dargestellt.

Bei der Analyse von Dokumentenbildern wird üblicherweise zwischen modernen und historischen Daten unterschieden [6]. Moderne Dokumentenbilder sind in der Regel leichter zu analysieren als historische [66, 189]. Dies ist unter anderem auf die oft standardisierten Formate, Schriftarten und Formatierungen in modernen Dokumenten zurückzuführen. Außerdem werden moderne Dokumente in der Regel mit hochwertigen Kameras oder Scannern digitalisiert und weisen nur geringe Bildartefakte auf. Ein weiterer Vorteil moderner Dokumente ist die hohe öffentliche Verfügbarkeit von Datensätzen und die Generierung neuer repräsentativer Trainingsdaten durch kostengünstige Syntheseverfahren. Historische Dokumente weisen dagegen eine Reihe von Herausforderungen auf. Ein kritischer Faktor ist der physische Zustand der Dokumente vor der Digitalisierung, da diese häufig aufgrund von Alterungsprozessen, unsachgemäßer Lagerung oder Handhabung beeinträchtigt sind [194]. Beispiele hierfür sind Verfärbungen, Wasserschäden, fehlende Bildbereiche, ein Pilzbefall oder das Durchscheinen der Schrift von der Rückseite [144, 194]. Derartige Schäden können die Lesbarkeit des Textes erheblich beeinträchtigen und damit die maschinelle Verarbeitung der Dokumente erschweren [66]. Eine weitere Herausforderung ist die oft begrenzte Anzahl von annotierten Trainingsmaterialien [144]. Dies ist sowohl auf die aufwendige Digitalisierung der physisch fragilen Dokumente als auch auf die aufwändige manuelle

Annotation der Daten zurückzuführen [48, 144]. Neben den visuellen Herausforderungen enthalten historische Texte oft einen kulturellen und historischen Kontext, der für eine Textanalyse berücksichtigt werden muss [48]. Dazu gehören insbesondere veraltete Sprachformen, Wörter und Ausdrücke, die in modernen Sprachkorpora nicht vorkommen. Zudem sind historische Dokumente oft sehr individuell und in speziellen Schreibschriften verfasst [144].

Neben dem Alterungsgrad wird bei der Dokumentenbildanalyse üblicherweise zwischen maschinell gedruckten und handgeschriebenen Texten unterschieden [6, 144]. Die Fehlerrate moderner Texterkennungsmodelle ist bei maschinell gedruckten Texten in der Regel geringer als bei handgeschriebenen Texten [151]. Dies ist unter anderem auf die vergleichsweise hohe Variabilität der Handschrift und die geringe Verfügbarkeit von repräsentativen und annotierten Trainingsdaten zurückzuführen. Dabei hat jeder Mensch eine einzigartige Handschrift, die sich unter anderem durch den Schreibstil, die Schriftgröße oder die Verbindung der Buchstaben verändern lässt [74]. So kann ein und dasselbe geschriebene Wort von derselben Person visuell grundlegend anders aussehen. Darüber hinaus verändert sich die Handschrift eines Menschen im Laufe der Zeit durch Alterung oder Krankheit [40].

Anwendungsbereiche der Dokumentenbildanalyse

Im Bereich der Dokumentenbildanalyse existiert eine enorme Vielfalt von Anwendungsgebieten. Zu den klassischen Anwendungen zählen die Texterkennung, die Extraktion sowie Suche von Informationen in Dokumentenbildern und die Analyse von Dokumentenlayouts [45]. Weitere typische Anwendungsgebiete sind die Klassifikation von Dokumententypen, die Identifizierung von Schreibern, forensische Analysen sowie Syntheseverfahren zur Dokumentengenerierung. In den letzten Jahren ist zudem eine starke Fokussierung auf das inhaltliche Verständnis von Dokumenten zu beobachten, wobei insbesondere Verfahren aus dem NLP-Bereich zunehmend bedeutsamer werden [36]. Im Folgenden werden die für diese Arbeit relevanten Anwendungsbereiche der Dokumentenbildanalyse vorgestellt und zueinander in Beziehung gesetzt.

LAYOUTANALYSE Die Layoutanalyse bestimmt den strukturellen Aufbau und die Position vordefinierter Elemente wie z.B. Textblöcke, Bilder, Überschriften oder Tabellen in einem Dokumentenbild. Das Verfahren dient als Grundlage für die automatisierte Verarbeitung von Dokumenteninhalten und wird in vielen spezifischen Anwendungsfällen, wie z.B. der Texterkennung, explizit oder implizit eingesetzt [6]. Das primäre Ziel der Analyse ist die Gliederung des Dokuments in logische Segmente, um eine spezifische Verarbeitung und Interpretation der Bereiche zu ermöglichen. Ein zentrales Teilgebiet der Layoutanalyse ist die Textsegmentierung, mit der die Textinhalte des Dokumentenbildes in logische Einheiten wie Wörter, Zeilen oder Absätze zerlegt werden [6].

TEXTERKENNUNG Eine klassische Anwendung der Dokumentenanalyse ist die Texterkennung, welche den Text aus digitalen Bildern extrahiert und in ein maschinenlesbares Format umwandelt. Generell wird bei der Texterkennung zwischen maschinell gedrucktem und handgeschriebenem Text unterschieden. Das Ergebnis des Verfahrens ist ein maschinenlesbarer Text, der durchsucht, bearbeitet und weiterverarbeitet werden kann. Häufig dient die Texterkennung als Zwischenschritt für nachfolgende Analyseverfahren.

INFORMATIONSEXTRAKTION Eine zentrale Aufgabenstellung der Dokumentenanalyse ist die Suche von Informationen in Dokumentenbildern. Ein Beispiel hierfür ist die Indexierung von Dokumenteninhalten, welche die Identifizierung und Lokalisierung von benutzerspezifischen Eingabewörtern innerhalb der Dokumente umfasst. Neben der Suche befasst sich dieser Forschungsbereich auch mit der automatischen Extraktion von vordefinierten Informationen aus Dokumenten, wie beispielsweise Namen, Daten, Beträgen oder Produkten. Durch die Extraktion dieser Informationen aus z.B. Formularen, Rechnungen oder Bestellungen können manuelle Extraktionsprozesse automatisiert und somit die Effizienz von Geschäftsprozessen gesteigert werden [36].

SEMANTISCHE ANALYSE Bei der semantischen Analyse von Dokumentenbildern werden die textuellen Inhalte der Bilder verarbeitet und interpretiert. Typische Anwendungen in diesem Bereich sind die semantische Informationsextraktion, die Beantwortung von Fragen auf der Basis von Dokumentenbildern und die Zusammenfassung von Textinhalten. Diese Ansätze kombinieren semantische und visuelle Informationen, um den Inhalt und den Kontext des Dokuments zu erfassen. Dies erfordert eine umfassende Analyse des gesamten Dokuments, einschließlich des Textinhalts, der Struktur und der visuellen Elemente. Hierbei werden häufig Ansätze aus dem NLP-Bereich eingesetzt.

3.2 NATURAL LANGUAGE PROCESSING

Natural Language Processing (NLP) ist ein Teilgebiet der künstlichen Intelligenz und beschäftigt sich mit dem Verständnis und der Generierung von sogenannter natürlicher Sprache[1] mit Maschinen. Ein zentrales Ziel ist die Entwicklung von Systemen, die Sprache erkennen, verstehen und erzeugen können. Das Forschungsgebiet unterteilt sich formal in die Bereiche Sprachverstehen (engl.: *Natural Language Understanding*) und Spracherzeugung (engl.: *Natural Language Generation*) [84, 251]. Beide Bereiche sind jedoch komplementär und werden häufig gemeinsam in einem System eingesetzt, insbesondere in komplexen Anwendungen wie z.B. Sprachassistenten [251].

Diese Arbeit beschränkt sich auf NLP-Anwendungen, die auf maschinenlesbarem Text basieren und keine Textgenerierung erfordern. Als maschinenlesbarer Text wird eine Folge von Symbolen definiert, die in einem standardisierten Format wie ASCII oder Unicode vorliegt. In diesen Formaten wird jedes Symbol aus einem vordefinierten Vokabular durch einen eindeutigen numerischen Wert repräsentiert. Aufgrund der Allgegenwärtigkeit und praktischen Relevanz von Text verfügt der Forschungsbereich der natürlichsprachlichen Textverarbeitung über eine langjährige Historie mit vielfältigen Anwendungen in verschiedensten Gebieten [84]. Gleichzeitig handelt es sich um eine anspruchsvolle Disziplin, die aufgrund der Komplexität und Vielfalt menschlicher Kommunikation mit zahlreichen Herausforderungen konfrontiert ist [146]. Insbesondere in den letzten Jahren konnten durch DL-basierte Ansätze erhebliche Fortschritte beim Verstehen und Generieren von maschinenlesbaren Texten erzielt werden [251]. Im Folgenden wird ein allgemeiner Überblick über die Geschichte, die Anwendungsgebiete und die Herausforderungen im NLP-Bereich gegeben. Für einen detaillierten Einblick in das Forschungsgebiet siehe [84].

1 Unter natürlicher Sprache wird im Rahmen dieser Arbeit jede von Menschen gesprochene Sprache verstanden, wie z.B. Deutsch, Englisch oder Chinesisch, die sich innerhalb einer Sprachgemeinschaft ungesteuert entwickelt hat und von deren Mitgliedern im täglichen Leben verwendet wird. Im Gegensatz dazu stehen formale Sprachen, wie z.B. Programmiersprachen, die durch bewusste Planung geschaffen wurden.

Vorgehensweise und Historie

Traditionelle NLP-Ansätze bestehen aus einer Abfolge von separat durchgeführten Verarbeitungsschritten wie der Tokenisierung, Satzsegmentierung, Wortartenanalyse, Syntaxanalyse und Semantikanalyse. Erste Ansätze zur Realisierung der einzelnen Schritte nutzen regelbasierte Systeme mit komplexen Regelwerken und Wörterbüchern, die von Linguisten und Computerwissenschaftlern entwickelt wurden [30, 84, S.61]. Eine flexiblere Alternative zu regelbasierten Systemen bieten statistische Modelle, wie z.B. das *Hidden Markov Model* (HMM), welche Wahrscheinlichkeiten für das Auftreten vorgegebener Wortfolgen und Strukturen berechnen [84, S.169-176]. Diese Modelle haben insbesondere im Bereich der Sprachmodellierung und der Sequenzvorhersage bedeutende Fortschritte erzielt [84, S.184-185]. Ein grundlegender Paradigmenwechsel im Bereich des Textverstehens wurde durch DL-basierte Ansätze herbeigeführt, bei denen die traditionelle NLP-*Pipeline* durch ein Ende-zu-Ende-Lernverfahren ersetzt wird [204]. Mit diesem Ansatz lernt ein Modell automatisch sprachrelevante Merkmale aus den Textdaten, ohne dass explizite Vorverarbeitungs- und Analyseschritte erforderlich sind. Durch den Einsatz von tiefen neuronalen Netzwerken in Kombination mit Textkorpora im zwei- bis dreistelligen Gigabyte-Bereich und leistungsfähigen Rechenressourcen können komplexe semantische und strukturelle Informationen gelernt werden [18, 42, 122]. Hierbei werden RNNs [149], Faltungsnetzwerke [18, 137] oder insbesondere Transformer-Modelle [42, 122] mit einer Parameteranzahl im zwei- bis dreistelligen Millionenbereich verwendet. Das Training dieser Modelle basiert auf unüberwachten Lernverfahren und kodiert universelle Sprachmerkmale sowie Weltwissen in kontinuierlichen vektoriellen Wortrepräsentationen [182]. Bekannte Beispiele für Worteinbettungsverfahren sind FastText [18], ELMo [149], BERT [42] und RoBERTa [122]. Die Anpassung dieser universellen Sprachmodelle an spezifische Anwendungen, wie z.B. der Textklassifikation, erfolgt durch das *Transfer-Learning* auf der Basis eines überwachten Trainingsprozesses. Der aktuelle Trend im NLP-Bereich tendiert zu großen Sprachmodellen (engl.: *Large Language Models* (LLMs)) mit einer Parameteranzahl im zwei- bis dreistelligen Milliardenbereich, welche natürlichsprachliche Texte generieren können [138, 251]. Diese neuronalen Modelle werden mit unüberwachten Lernverfahren und -techniken auf Textkorpora von mehreren Terabyte an Textdaten trainiert [138]. Die Entwicklung von LLMs ist sowohl kosten- als auch ressourcenintensiv und kann daher derzeit nur von wenigen großen Akteuren durchgeführt werden [251].

Aufgaben und Anwendungsbereiche

Das Anwendungsspektrum im NLP-Bereich ist vielfältig und umfasst Aufgaben von der Textübersetzung bis hin zur komplexen Textanalyse und Generierung natürlichsprachlicher Inhalte [84]. Dieser Abschnitt enthält eine Auswahl von grundlegenden und für diese Arbeit relevanten NLP-Anwendungen. Zur besseren Übersicht sind die Anwendungen in die Bereiche Textverstehen und Textgenerierung eingeordnet. Eine eindeutige Zuordnung ist jedoch nicht in allen Fällen möglich, da einige Anwendungen aus einer Kombination beider Bereiche bestehen. Zudem lassen sich manche Anwendungen sowohl dem Textverständnis als auch der Textgenerierung zuordnen. Ein Beispiel hierfür ist das QA, bei dem die Antwort entweder aus einem Text extrahiert oder als natürlichsprachlicher Text erzeugt werden kann.

TEXTVERSTÄNDNIS Das Textverständnis umfasst Aufgaben, die darauf abzielen, die Bedeutung und Struktur von Texten zu verstehen ohne dabei neuen Text zu generieren. Eine der meistverwendeten Anwendungen in diesem Bereich ist die Klassifikation von Texten in vorgegebene Kategorien, welche zur automatischen Organisation und Filterung von Inhalten verwendet wird. Bekannte Beispiele hierfür sind die Identifizierung von Spam-Nachrichten in E-Mails und die Stimmungsanalyse (engl.: *Sentiment Analysis*) in Kundenbewertungen. Eine weitere praxisrelevante Anwendung im NLP-Bereich ist die Informationsextraktion, bei der strukturierte Informationen aus unstrukturierten Texten extrahiert werden. Dies kann z.B. die Wissensgraphextraktion [83], das Erkennen von Entitäten in Texten [242], sowie das Extrahieren von Beziehungen zwischen diesen Entitäten sein [181]. Zudem ist für diese Arbeit die *Machine Reading Comprehension* (MRC)-Aufgabe [249] von besonderem Interesse. Dies ist ein Teilgebiet des QAs bei dem Fragen zu einem gegebenen Text beantwortet werden, indem das System die Position der Antwort innerhalb des Textes extrahiert, ohne dabei Text zu generieren. Im Bereich des Textverstehens gibt es zudem eine Vielzahl von Anwendungen zur logischen Textanalyse [84, S.248], wie z.B. die Paraphrasenerkennung, die Textkohärenzbewertung oder die textuelle Inferenz, welche in praktischen Anwendungsbereichen eine vergleichsweise geringe Verbreitung haben.

TEXTGENERIERUNG Die Textgenerierung umfasst Aufgaben, die darauf abzielen, Texte zu erzeugen. Klassische Beispiele hierfür sind die Erstellung natürlichsprachlicher Produktbeschreibungen, Blogbeiträge und Zeitungsartikel aus strukturierten Daten [159]. Diese Funktionen helfen dabei, den Arbeitsaufwand für das Verfassen von Texten zu reduzieren und die Produktivität zu steigern. Viele der Anwendungen zur Textgenerierung erfordern ein vorheriges Verständnis einer gegebenen Eingabe, um daraufhin eine angemessene Ausgabe erzeugen zu können. Klassische Beispiele hierfür sind die Textzusammenfassung, die maschinelle Übersetzung und Dialogsysteme. Bei der Zusammenfassung von Texten wird der Inhalt einer längeren textuellen Eingabe in eine kompaktere Form überführt. Dies erfordert das Verständnis des Inhalts und eine Generierung von natürlichsprachlichem Text auf Basis des kodierten Wissens [84, S.196]. Ähnlich verhält es sich bei der maschinellen Übersetzung, bei der Texte von einer Sprache in eine andere übersetzt werden. Die kontextbezogene Natur von Übersetzungen erfordert ein tiefes Verständnis der Syntax und Semantik beider Sprachen, um präzise Übersetzungen zu erzeugen [84, S.267-272]. Ein Beispiel bei der die Kombination aus Kontextverständnis und Textgenerierung besonders relevant ist, sind Dialogsysteme bzw. Chatbots [245]. Die Aufgabe der Systeme besteht darin, auf natürlichsprachliche Eingaben von Benutzern zu reagieren und relevante Antworten zu erzeugen, die dem Kontext der Unterhaltung entsprechen und die beabsichtigten Informationen vermitteln.

Herausforderungen

Die automatisierte Verarbeitung von Sprache stellt aufgrund der Komplexität und Vielfalt menschlicher Kommunikation zahlreiche Herausforderungen dar. Hierbei sind insbesondere die Ambiguität und Variabilität der Sprache sowie die Rechtschreib- und Grammatikfehler eine große Herausforderung.

AMBIGUITÄT Sprache ist von Natur aus mehrdeutig, da einzelne Wörter und Sätze je nach Kontext verschiedene Bedeutungen haben können [146, S.38]. Die Auflösung dieser

Mehrdeutigkeit ist essentiell für das Verstehen und die Generierung von Sprache. Es werden verschiedene Formen der Ambiguität unterschieden [146, S.38], wobei die lexikalische, die syntaktische und die referentielle Ambiguität in dieser Arbeit am relevantesten sind. Die lexikalische Ambiguität [146, S.39-43] bezieht sich auf Wörter oder Ausdrücke, die mehrere Bedeutungen haben. Hierbei bilden die Polysemie und Homonymie zwei spezifische Arten der lexikalischen Ambiguität [146, S.39-42]. Ein Beispiel für die Homonymie ist das Wort „Rock“, das sowohl ein Kleidungsstück als auch eine Musikrichtung bezeichnet. Eine weitere Form der Ambiguität liegt vor, wenn ein Satz mehrere syntaktische Strukturen zulässt und somit auf verschiedene Weise interpretiert werden kann [146, S.44]. Ein Beispiel ist der Satz „Erik sah den Täter mit dem Fernglas“. Hier ist nicht klar, ob Erik den Täter sah, der ein Fernglas hatte, oder ob Erik den Täter sah, während er selbst ein Fernglas benutzte. Besonders problematisch ist die referentielle Ambiguität [146, S.44]. Dabei handelt es sich um die Unklarheit, welche Entität mit einem Ausdruck gemeint ist. Ein Beispiel ist der Satz „Anna erzählte Maria, dass sie gewonnen hat“, wobei unklar ist, ob das Wort „sie“ sich auf Anna oder Maria bezieht.

VARIABILITÄT Die Verwendung von Sprachen ist je nach Kontext, Person und Region sehr unterschiedlich [84, S.15-17]. Dies gilt insbesondere für Dialekte und regionale Variationen, bei denen es sich um Unterschiede im Wortschatz, in der Grammatik und in der Aussprache handelt. Ein Beispiel dafür sind die Bezeichnungen „Brötchen“, „Semmel“ oder „Wecken“ für dasselbe Kleingebäck. Eine weitere Form der Variabilität sind Sprachvarianten, die von bestimmten gesellschaftlichen Gruppen verwendet werden. Diese können beispielsweise Berufsgruppen, Altersgruppen oder kulturelle Gemeinschaften sein. Ein typisches Beispiel ist die Jugendsprache, in der *Slang* und neue, schnell wechselnde Begriffe verwendet werden. Weitere Variabilität bringen Redewendungen (z.B. „Ins Gras beißen“), Sprichwörter (z.B. „Aller Anfang ist schwer“) oder Phrasen (z.B. „Im Handumdrehen“). Hierbei handelt es sich um sprachliche Ausdrücke, deren Bedeutung sich nicht aus den einzelnen Wörtern ableiten lässt. Eine weitere Form der Variabilität sind Synonyme [146, S.27], wobei verschiedene Wörter oder Ausdrücke dieselbe oder eine sehr ähnliche Bedeutung haben.

TEXTFEHLER Bei der Verarbeitung von Texten stellen Textfehler, wie Rechtschreib- und Grammatikfehler, eine große Herausforderung dar [85]. Sie beeinträchtigen häufig die Genauigkeit und Effizienz vieler Anwendungen [124, 189]. Textfehler verändern die Bedeutung eines Wortes oder eines ganzen Textes, wodurch die Analyse und das Verständnis des Inhalts stark beeinträchtigt werden [85]. Ein Beispiel hierfür ist das Adjektiv „rot“, das durch die Änderung des ersten Buchstabens mit einem „t“ zu „tot“ wird und damit eine gänzlich andere semantische Bedeutung bekommt. Dies ist beispielsweise bei Suchanfragen problematisch, da die genaue Absicht des Nutzers nur schwer zu ermitteln ist. Darüber hinaus können Textfehler beim Training von maschinellen Lernverfahren problematisch sein, da das Modell falsche Muster lernt [143].

3.3 ANWENDUNGSBEREICHE DIESER ARBEIT

Aufgrund der unterschiedlichen Vorgehensweisen und Anforderungen von Anwendungen im Bereich der Dokumentenbildanalyse beschränkt sich diese Arbeit auf die inhaltliche Analyse von handschriftlichen Dokumentenbildern ohne Textgenerierung. Von besonderer

Relevanz sind Aufgaben aus dem Bereich der digitalen Bibliotheken, da diese oft große Mengen an handschriftlichen Dokumentenbildern enthalten, deren Textinhalte nicht in einem maschinenlesbaren Format vorliegen. Ungeachtet der komplexen Verarbeitung von Bilddaten erwarten die Anwender von digitalen Bibliotheken eine benutzerfreundliche und effiziente Exploration der Inhalte mit natürlichsprachlicher Interaktion zur Informationsbeschaffung. Um diesen hohen Anforderungen gerecht zu werden, ist eine effiziente Suche, Indexierung und Exploration der großen Dokumentenmengen notwendig. In diesem Kontext sind insbesondere die semantische Schlüsselwortsuche, die NER und das QA von zentraler Relevanz. Die drei semantischen Aufgaben werden im Folgenden ausführlich motiviert und vorgestellt. Dazu wird neben einer formalen Definition der Aufgaben ein Literaturüberblick auf dem Gebiet der semantischen Analyse von handschriftlichen Dokumentenbildern gegeben. Für die NER- und QA-Aufgaben wird zusätzlich ein allgemeiner Überblick über aktuelle Ansätze im NLP-Bereich und verwandte Arbeiten auf dem Gebiet der semantischen Analyse von maschinell gedruckten Dokumentenbildern vorgestellt.

3.3.1 *Semantische Schlüsselwortsuche*

Die Suchfunktion ist eine der grundlegendsten und wichtigsten Funktionalitäten von digitalen Bibliotheken. Das Ziel dieser Funktionalität ist das effiziente Auffinden relevanter Textstellen in den oft umfangreichen Datenmengen bezüglich einer benutzerdefinierten Suchanfrage. Die Qualität der Suchergebnisse ist ein zentraler Faktor für die Nutzerzufriedenheit und die Effizienz der Informationsbeschaffung. Die Realisierung einer leistungsfähigen Suchfunktion für handschriftliche Dokumentenbilder ist jedoch nicht trivial und stellt ein umfangreich untersuchtes Forschungsgebiet dar [61]. Ein intuitiver Ansatz zur Realisierung einer Suchfunktion für handschriftliche Dokumentenbilder ist die Umwandlung der textuellen Inhalte aus den Dokumentenbildern in ein maschinenlesbares Format und die anschließende Nutzung etablierter Ansätze aus dem *Information Retrieval* (IR)-Bereich [140]. Jedoch kann es trotz großer Fortschritte bei der Texterkennung speziell für historische Dokumente zu hohen Fehlerraten kommen, die sich erheblich auf die Qualität der IR-Verfahren auswirken [140]. Daher hat sich für diese Eingabedaten mit der sogenannten Schlüsselwortsuche (engl.: *Keyword Spotting* oder kurz *Word Spotting* (WS)) ein robusteres Verfahren etabliert, das analog zu Websuchmaschinen eine *Retrieval*-Liste als Suchergebnis bereitstellt. Dabei wird eine explizite Texterkennung vermieden und die Ähnlichkeit zwischen einem Wortbild und einer Anfrage auf der Basis von Merkmalsrepräsentationen bestimmt. Das Ziel der Schlüsselwortsuche ist die Identifizierung von relevanten Wortbildern aus einer gegebenen Dokumentensammlung bezüglich einer Suchanfrage und die Anordnung dieser Wortbilder in einer Retrieval-Liste nach ihrer Ähnlichkeit zur Anfrage. Hierbei werden Instanzen mit derselben Transkription wie die Suchanfrage als relevant angesehen. In dieser Arbeit wird die Schlüsselwortsuche, bei der die Ähnlichkeit zwischen einem Wortbild und einer Anfrage ausschließlich auf der Basis von visuellen Merkmalen bestimmt wird, als syntaktische Suche bezeichnet.

Der Nachteil der syntaktischen Schlüsselwortsuche besteht darin, dass die semantische Bedeutung der Suchanfrage nicht berücksichtigt wird und daher z.B. bei einer Suche nach dem Wort „Hund“ das Wortbild mit der Transkription „Dalmatiner“ als unähnlich eingestuft wird, obwohl es für einen Benutzer potentiell relevant ist. Ein Lösungsansatz, der bereits zu einer deutlichen Verbesserung der Qualität von Websuchmaschinen geführt hat, ist die Berücksichtigung semantischer Informationen beim Retrieval [217]. Dies ermöglicht

es den Nutzern, nicht nur nach Wortbildern mit einer bestimmten Transkription zu suchen, sondern auch nach Konzepten, wie z.B. ähnliche Bedeutungen oder kategorische Beziehungen. Eine semantische Schlüsselwortsuche ermöglicht somit eine genauere Erfassung der Bedeutung von Suchanfragen und hilft Nutzern bei der Exploration von neuen und relevanten Inhalten, indem Wortbilder aus semantisch verwandten Themenbereichen in der Retrieval-Liste angezeigt werden. Die semantische Schlüsselwortsuche ist im Allgemeinen eine Erweiterung der syntaktischen Suche, bei der nicht nur visuelle, sondern auch semantische Informationen zur Bestimmung der Ähnlichkeit zwischen einem Wortbild und einer gegebenen Anfrage berücksichtigt werden. Das Ziel dieser Aufgabe ist die Erstellung einer Retrieval-Liste, in der alle Wortbilder mit der gleichen Transkription wie die Suchanfrage an den obersten Stellen erscheinen, gefolgt von semantisch ähnlichen Wortbildern.

Terminologie

Neben der Unterscheidung von semantischen und syntaktischen Ansätzen existiert eine weitgehend anerkannte Terminologie zur Schlüsselwortsuche, die im Bereich der Dokumentenanalyse verwendet wird. Anhand der im Folgenden eingeführten Fachwörter können WS-Ansätze beschrieben und in unterschiedliche Kategorien unterteilt werden. Hierbei unterscheiden sich die WS-Ansätze zunächst in *Online*- und *Offline*-Verfahren, wobei die Eingabedaten für ein Online-Modell [79, 81, 82] mit einem elektronischen Eingabegerät aufgezeichnet wurden und über zeitliche Informationen bezüglich der Handschrifterstellung verfügen. Die Eingabedaten für Offline-Modelle [55, 129, 170] sind digitalisierte Aufnahmen von Dokumentenbildern und weisen folglich Artefakte und keine Informationen über den Entstehungsprozess auf. Eine weitere grundlegende Unterscheidung ist die Art der erwarteten Segmentierung für WS-Modelle, wobei sich die Ansätze hinsichtlich einer Segmentierung auf Wort- [8, 97, 129, 191], Zeilen- [55, 92, 163] oder Seitenebene [7, 164, 170, 234] unterscheiden. Die wort- und zeilensegmentierten Verfahren haben die Einschränkung, dass sie eine externe Segmentierung der Dokumente auf Wort- bzw. Zeilenebene benötigen, während die segmentierungsfreien Ansätze eine implizite Segmentierung durchführen und somit auf dem gesamten Dokumentenbild arbeiten. Eine weitere Unterscheidung kann zwischen trainingsbasierten und trainingsfreien Ansätzen getroffen werden, je nachdem, ob ein explizites Training des WS-Modells erforderlich ist [61]. Für die praktische Anwendung ist die Trennung auf Basis des Trainings jedoch nicht von grundlegender Bedeutung, sondern vielmehr die Frage, ob manuell annotierte Trainingsdaten für die Erstellung des Ansatzes benötigt werden. Aus diesem Grund hat sich der Fokus in den letzten Jahren auf die Unterscheidung zwischen annotationsbasierten und annotationsfreien Ansätzen verschoben [236]. Ein grundlegendes Charakteristikum von WS-Ansätzen ist zudem das Format, in dem die Suchanfragen gestellt werden können. Es existiert eine Vielzahl unterschiedlicher Anfragetypen [61, 171, 231], von denen *Query-by-Example* (QbE) und *Query-by-String* (QbS) die bekanntesten und am häufigsten verwendeten Formate darstellen [61]. Ein QbE-basiertes WS-Modell erfordert ein beispielhaftes Wortbild des gesuchten Wortes als Eingabe. Dieses Format ist für einen Benutzer sehr umständlich, da das Anfragebild des gesuchten Wortes erst im Dokument gefunden werden muss. Ein QbS-basiertes WS-System erwartet als Eingabe ein maschinenlesbares Textformat und ist damit für einen Benutzer wesentlich komfortabler, da beliebige Anfragen z.B. über die Tastatur eingegeben werden können.

Syntaktische Schlüsselwortsuche

Die syntaktische Schlüsselwortsuche ist kein zentraler Anwendungsfall dieser Arbeit, jedoch basieren die in der Literatur vorgestellten semantischen WS-Ansätze weitgehend auf den Entwicklungen in diesem Bereich. Darüber hinaus werden in dieser Arbeit syntaktische und semantischen WS-Ansätze verglichen. Daher wird im Folgenden ein allgemeiner Literaturüberblick über syntaktische WS-Verfahren gegeben. Aufgrund des generellen Fokus dieser Arbeit auf wortsegmentierten Verfahren beschränkt sich der Literaturüberblick auf WS-Ansätze, die eine externe Segmentierung der Eingabedaten auf Wortebene erfordern. An dieser Stelle sei darauf hingewiesen, dass der Literaturüberblick insbesondere für die klassischen WS-Ansätze stark verallgemeinert ist und keinen Anspruch auf Vollständigkeit erhebt. Für einen detaillierten Überblick über das Forschungsgebiet der Schlüsselwortsuche auf Dokumentenbildern ist die Arbeit von Giotis et al. in [61] zu empfehlen.

Bei der wortsegmentierten Schlüsselwortsuche wird die gegebene Dokumentensammlung als eine Menge von segmentierten Wortbildern aufgefasst. Die in der Literatur beschriebenen Verfahren folgen in der Regel einem einheitlichen Ansatz zur Ermittlung der Ergebnislisten. Hierbei werden die Wortbilder der Dokumentensammlung zunächst in eine Merkmalsrepräsentation überführt und in einer Datenbank indexiert. Anschließend werden für eine gegebene Anfrage die paarweisen Distanzen zwischen der Anfrage und den Elementen aus der Datenbank auf der Basis einer gemeinsamen Repräsentation bestimmt. Abschließend werden alle Wortbilder aus der Datenbank anhand der Distanzen in aufsteigender Reihenfolge in einer Ergebnisliste aufgelistet. Die Ansätze aus der Literatur unterscheiden sich vor allem hinsichtlich der Wortbildrepräsentation.

Der erste WS-Ansatz für handschriftliche Dokumentenbilder wurde von Manmatha et al. in [129] vorgestellt. Das Verfahren basiert auf dem Prinzip des *Template Matchings*, bei dem die Ähnlichkeit zwischen zwei Wortbildern auf Basis der Pixelintensitätswerte durchgeführt wird [89, 128, 129]. Hierbei werden alle Wortbilder aus der Datenbank zunächst binarisiert und anhand des Anfragebildes ausgerichtet. Anschließend werden die Distanzen zwischen dem binarisierten Anfragebild und jedem Wortbild aus der Datenbank ermittelt, indem eine XOR-Operation zwischen den Wortbildern durchgeführt und die Anzahl der übereinstimmenden Pixel berechnet wird. Ein pixelweiser Vergleich ist aufgrund der hohen Variabilität von Handschriften problematisch [129]. Daher haben sich merkmalsbasierte Ansätze etabliert, bei denen sowohl die Wortbilder aus der Datenbank als auch das Anfragebild in vektorielle bzw. sequentielle Merkmalsrepräsentationen überführt werden [61]. Besonders geometrische Merkmale, wie z.B. Konturen oder Projektionsprofile haben sich als Wortbildrepräsentation bewährt [3, 61, 128, 158]. Dabei handelt es sich im Allgemeinen um heuristische Merkmale, die für jede Spalte eines Graustufenbildes bestimmt werden und somit eine Sequenz in Abhängigkeit von der Länge des Wortbildes ergeben. Aufgrund der häufig unterschiedlichen Längen der Sequenzen und den Variationen von Handschriften erweist sich das *Dynamic Time Warping* [174] als geeignetes Distanzmaß für diese Repräsentationsformen [157]. Sequenzielle Modelle wie z.B. HMMs [51, 164] und RNNs [55] werden ebenfalls erfolgreich für die Schlüsselwortsuche eingesetzt.

Motiviert von den Fortschritten aus dem Forschungsbereich der *Computer Vision* (CV) haben sich gradientenbasierte Merkmale im Bereich der Schlüsselwortsuche als robustere Alternative zu den geometrischen Merkmalen herausgestellt [61]. In diesem Zusammenhang haben sich lokale Bilddeskriptoren auf der Basis von Gradientenstatistiken, wie z.B. *Scale Invariant Feature Transform* [125], *Local Binary Pattern* [145] oder *Histogram-of-Gradients* [37] als besonders effektiv erwiesen. Im Gegensatz zu den sequentiellen Wortrepräsenta-

tionen aus den geometrischen Ansätzen können mit dem *Bag-of-Features* (BoF)-Ansatz [35] ganzheitliche vektorielle Wortbilddarstellung für gradientenbasierte Verfahren erzeugt werden. Dies ermöglicht die Bestimmung der Ähnlichkeit zweier Wortbilder durch einen effizienten Vektorvergleich. Bei dem BoF-Verfahren werden aus einer gegebenen Stichprobe von Wortbildern lokale Bilddeskriptoren berechnet und mit einem *Clustering*-Ansatz eine vorgegebene Anzahl von möglichst repräsentativen Merkmalen ermittelt. Für jedes Wortbild werden anschließend die enthaltenen Bilddeskriptoren anhand der berechneten Clusterzentroide nach dem Nächster-Nachbar-Prinzip quantisiert und in ein Histogramm überführt. Ein wesentliches Problem des BoF-Ansatzes besteht in der fehlenden Kodierung von Merkmalspositionen, die unter anderem im Hinblick auf Anagramme von grundlegender Bedeutung sind. Eine Lösung für dieses Problem bietet das SPP [105], welches das gegebene Wortbild zunächst in k möglichst gleich große Regionen unterteilt und die Histogramme des BoF-Ansatzes für jede Region unabhängig voneinander erzeugt. Das Ergebnis sind k Histogramme, die durch eine Konkatenation zur finalen Wortbildrepräsentation führen.

Einen weiteren vielversprechenden Forschungsansatz im WS-Bereich bilden die sogenannten strukturellen Ansätze. Diese Verfahren basieren auf der Annahme, dass ein handgeschriebenes Wort aufgrund seines sequentiellen Entstehungsprozesses auf natürliche Weise als Graph dargestellt werden kann. Dabei wird angenommen, dass Graphdarstellungen im Vergleich zu Vektordarstellungen im Allgemeinen eine leistungsfähigere und natürlichere Darstellung für Wortbilder bieten [52, 160, 186, 187, 223]. Um das WS-Problem in der Graphdomäne anzugehen, muss das Eingabebild zunächst in eine Graphdarstellung überführt werden. In diesem Bereich haben sich sowohl *Keypoint*-basierte Ansätze [52] als auch Verfahren auf der Basis von Projektionsprofilen [186] bewährt. Ausgehend von der Graphdarstellung wird schließlich die Ähnlichkeit eines Anfragebildes mit allen Wortbildern aus der Datenbank mittels einer Grapheditierdistanz berechnet. Aufgrund des hohen Rechenaufwands für die Bestimmung der Ähnlichkeit zwischen zwei Graphen ist eine Approximation der Graphdistanz notwendig [53].

Ein elegantes und effizientes WS-Verfahren bietet das von Almazán et al. in [8] vorgestellte Konzept der eingebetteten Attributrepräsentationen. Die Grundidee besteht darin, maschinenlesbaren Text und Wortbilder in einen gemeinsamen Vektorraum abzubilden, sodass die Merkmalsrepräsentationen eines Wortbildes und seiner Transkription im Vektorraum nahe beieinander liegen. Die Ähnlichkeit zwischen zwei Elementen wird durch den Abstand ihrer Repräsentationen im Vektorraum bestimmt. Dieser Ansatz ermöglicht damit sowohl QbE- als auch QbS-Anfragen. Der gemeinsame Vektorraum für Wortbilder und maschinenlesbare Wörter wird durch die sogenannte *Pyramidal Histogram of Characters* (PHOC)-Repräsentation [8] realisiert. Dies ist eine vektorielle Wortrepräsentation, die das Vorkommen von Zeichen und ihre räumlichen Positionen in einem gegebenen Wort kodiert. Für weitere Informationen über diese attributbasierte Repräsentation siehe Exkurs 2. Ein erster Ansatz zur Realisierung des Verfahrens stammt von Almazán et al. in [8]. Zur Vorhersage von PHOC-Vektoren für ein Wortbild überführt der vorgestellte Ansatz zunächst das Bild in eine vektorielle Merkmalsrepräsentation. Hierfür werden Fisher-Vektoren [148] verwendet, die als eine Erweiterung des BoF-Verfahrens angesehen werden können. Mit den Wortbilddarstellungen wird anschließend ein SVM-basiertes Modell zur Vorhersage von PHOC-Vektoren trainiert. Dieses Modell besteht aus einem *Ensemble* von SVMs, von denen jede SVM ein Attribut des PHOC-Vektors vorhersagt.

Basierend auf den Fortschritten neuronaler Netzwerke im Bereich der CV haben Sudholt et al. mit dem *PHOCNet* [191] die Kombination aus Fisher-Vektoren und SVMs

Exkurs 2: Pyramidal Histogram of Characters

Die *Pyramidal Histogram of Characters* (PHOC)-Repräsentation [8] ist derzeit einer der am weitesten verbreiteten Ansätze zur vektoriellen Darstellung von handschriftlichen Wortbildern. Dabei wird ein Wort in einen binären Vektor umgewandelt, indem es in eine pyramidale Struktur unterteilt wird und für jede dieser Regionen das Vorhandensein bzw. Fehlen von Zeichen mit Histogrammen kodiert wird. Die finale Repräsentation setzt sich aus der Konkatenation aller Histogramme zusammen. Eine pyramidale Struktur ist notwendig, da bei der Verwendung eines einzelnen Histogramms die Reihenfolge der Zeichen verloren geht und somit Anagramme (z.B. Ampel und Palme) aufgrund der gleichen Darstellung nicht unterschieden werden können. Neben einem Vokabular müssen zudem die Ebenen der pyramidalen Struktur angegeben werden, wobei die Ebene $i \in \mathbb{N}_{\geq 1}$ das Wort in i gleichgroße Bereiche unterteilt und für jedes Zeichen des Wortes eine Breite von 1 angenommen wird. Für jedes Zeichen im Vokabular wird das Vorkommen in jeder der Wortregionen ermittelt. Dafür werden die entsprechenden Positionen im Histogramm für mindestens einmal vorkommende Zeichen in der Wortregion auf 1 und für fehlende Zeichen auf 0 gesetzt. Für die Bestimmung des Vorkommens eines Zeichens in einer Wortregion muss das Zeichen mindestens eine 50-prozentige Überschneidung mit der Region aufweisen.

zur Bestimmung der PHOC-Vektoren aus [8] durch ein CNN ersetzt. Hierbei handelt es sich um einen Ende-zu-Ende-Ansatz, der die PHOC-Repräsentation direkt auf Basis eines Wortbildes lernt und vorhersagt. Dazu wird eine zum *VGG-19* [183] ähnliche Netzwerkarchitektur zur Extraktion von Merkmalen aus den Bildern und ein dreischichtiges MLP zur Vorhersage der PHOC-Repräsentation verwendet. Die Anzahl der Neuronen in der letzten Schicht des MLPs entspricht der Dimension des zu lernenden PHOC-Vektors. Aufgrund der binären Zielrepräsentation wird eine sigmoidale Aktivierungsfunktion auf die Ausgabe des Netzwerks angewendet. Sudholt et al. haben das PHOCNet in [192] um eine *Temporal Pyramid Pooling* (TPP)-Schicht erweitert. Diese Schicht transformiert die Merkmalskarten aus dem CNN in eine vordefinierte Dimensionalität und integriert zudem die zeitlichen Aspekte des Schreibprozesses.

Eine zum PHOCNet ähnliche Vorgehensweise wird im zweistufigen WS-Ansatz von Wilkinson et al. [233] realisiert. Hierbei wird die Vorhersage der Wortrepräsentation jedoch nicht mit einer Ende-zu-Ende-Architektur gelernt, sondern das Wortbild zunächst in eine Bildrepräsentation mit einem sogenannten *Triplet*-CNN und der *SoftPN*-Verlustfunktion [12] überführt. Ein Triplet besteht aus einem Ankerwortbild, einem Positivbeispiel mit der gleichen Annotation und einem Negativbeispiel mit einer vom Anker abweichenden Annotation. Während des Trainings wird dasselbe CNN für alle drei Wortbilder des Triplets verwendet und darauf trainiert, dass die Repräsentationen der Wortbilder mit gleicher Annotation im Vektorraum möglichst nahe beieinander liegen und der Abstand zum negativen Beispiel möglichst groß ist. Nach dem Training des CNNs wird eine Abbildung von der Bildrepräsentation mit einem zweischichtigen MLP-Modell auf die gewünschte Wortrepräsentation (z.B. PHOC) gelernt.

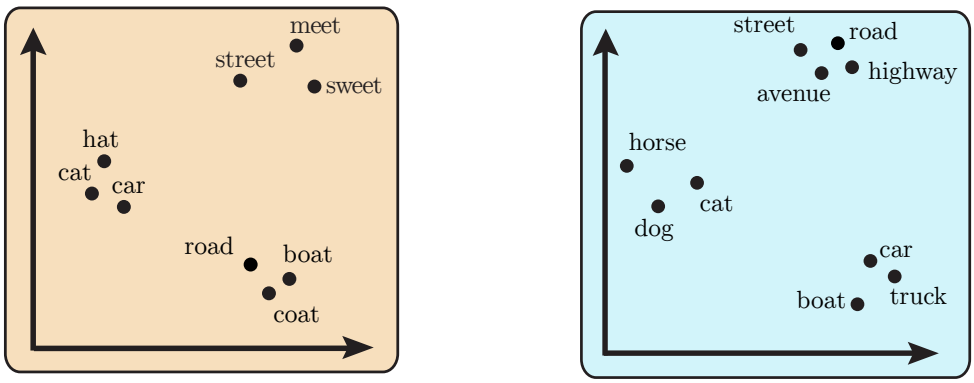
Die bisher vorgestellten Ansätze haben beim Training des WS-Modells lediglich die Abbildung von Wortbildern in einen textuell vordefinierten Vektorraum gelernt und die

Repräsentationen von maschinenlesbaren Texten während des Trainings nicht angepasst. Inspiriert von dieser Idee haben sich Ansätze etabliert, die Wortbilder und Texte in einem gemeinsamen Vektorraum abbilden, ohne die Zielrepräsentation auf textueller Ebene vorzugeben. Eine der ersten und bekanntesten Ansätze in diesem Bereich ist das von Krishnan et al. vorgestellte *HWNetv2* [97]. In diesem Ende-zu-Ende-Modell werden während des Trainings die Wortbilder und ihre textuellen Annotationen mit neuronalen Modellen in einen gemeinsamen Vektorraum projiziert und auf Basis dieser Repräsentationen eine lexikonbasierte Worterkennung mit einem MLP durchgeführt. Die Anzahl der Neuronen in der letzten Schicht des MLPs entspricht dabei der Anzahl der eindeutigen Wortannotationen in der Trainingsmenge. Für die Projektion von Wortbildern und Texten werden unterschiedliche Ansätze verwendet, wobei im Falle der Wortbilder ein auf dem ResNet34 basierendes Modell eingesetzt wird. Die Projektion eines maschinenlesbaren Wortes ist komplexer. Hier wird das Wort zunächst mit computerbasierten Schriftarten in ein synthetisches Wortbild umgewandelt. Dieses wird anschließend mit einem CNN in eine Merkmalsrepräsentation mit fester Dimensionalität transformiert und mit der PHOC-Repräsentation des Wortes konkateniert. Die Textrepräsentation wird schließlich mit einer linearen Schicht auf die Ausgabedimension des Wortbildmodells abgebildet. Nach dem Training werden die Merkmale der vorletzten Schicht des MLPs als Repräsentation für Wortbilder und maschinenlesbare Wörter verwendet. Ein wesentlicher Vorteil dieses Ansatzes ist, dass die Dimensionalität der Repräsentation flexibel festgelegt werden kann. Eine Optimierung der Architektur durch Verwendung eines gemeinsamen CNN-Modells für synthetische und reale Wortbilder wurde kürzlich in [94] veröffentlicht.

Semantische Schlüsselwortsuche

Die semantische Schlüsselwortsuche ist ein vergleichsweise neues Forschungsgebiet, zu dem bisher nur wenige Arbeiten veröffentlicht wurden. Ein erster Ansatz von Krishnan et al. [95] verwendet manuell annotierte semantische Informationen zur Erweiterung der syntaktischen Schlüsselwortsuche auf synonyme Wortbilder. Dazu werden verwandte Wörter im Bezug auf das Anfragebild mit einem semantischen Index ermittelt, ohne dass eine explizite Texterkennung durchgeführt wird. Der semantische Index basiert auf einer Teilmenge der Trainingsdaten und verwendet eine Kombination aus BoF-Repräsentationen und semantischen Informationen aus sogenannten Ontologien zur Bestimmung von synonymen Wortbildern für die Anfrage. Bei Ontologien handelt es sich um manuell annotierte Wissensdatenbanken, welche unter anderem semantische Beziehungen zwischen Wörtern in einer graphartigen Datenstruktur speichern. Für das Anfragebild und einer gefilterten Menge der synonymen Wortbilder aus dem semantischen Index werden jeweils Retrieval-Listen auf der Grundlage von BoF-Repräsentationen berechnet und in einem finalen Schritt zusammengeführt.

Ein fundamentaler Nachteil bei ontologiebasierten Ansätzen ist die geringe Anzahl an semantischen Beziehungen aufgrund des hohen manuellen Annotationsaufwandes. Daher geht der aktuelle Trend im NLP-Bereich zu vektoriellen Wortrepräsentationen, die semantische Eigenschaften von und zwischen Wörtern durch ein selbstüberwachtes Training auf umfangreichen Textkorpora erlernen [21, 42, 149, 182]. Die Ansätze der semantischen Schlüsselwortsuche bauen weitestgehend auf dem Ansatz des gemeinsamen Vektorraums zwischen Wortbildern und maschinenlesbaren Wörtern aus dem syntaktischen WS-Bereich auf. Dabei werden die syntaktischen Wortrepräsentationen durch semantische Worteinbet-



(a) Syntaktisch

(b) Semantisch

Abbildung 3.2: Eine beispielhafte Visualisierung eines semantischen und syntaktischen Worteinbettungsraums. Die Abstände zwischen Wörtern im syntaktischen Einbettungsraum korrelieren in etwa mit ihrer String-Editierdistanz. Dagegen entsprechen die Distanzen im semantischen Vektorraum den semantischen Ähnlichkeiten zwischen Wörtern.

tungen aus dem NLP-Bereich ersetzt (siehe Abbildung 3.2). Das Ziel dieses Ansatzes besteht daher nicht darin, semantische Beziehungen zwischen Wortbildern aus großen Text- oder Bildkorpora zu lernen, sondern das bereits kodierte semantische Wissen aus den textuellen Repräsentationen zu nutzen. Wilkinson et. al entwickelten die Grundidee zur Abbildung von Wortbildern in einen textuell vortrainierten semantischen Worteinbettungsraum [233]. Hierbei handelt es sich um denselben zweistufigen Ansatz aus dem syntaktischen WS-Bereich, welcher zunächst eine Bildrepräsentation mit einem Triplet-CNN erlernt und aufbauen darauf eine Abbildung in den semantischen Raum mit einem MLP realisiert [233]. Zudem können Ende-zu-Ende-Ansätze [98, 193] bestehend aus einem CNN und einem MLP zur Abbildung der Wortbilder in einen textuell vortrainierten semantischen Vektorraum verwendet werden. Bei den Ansätzen handelt es sich analog zu dem von Wilkinson et al. nicht um eine explizite semantische Architektur, sondern um einen generellen Ansatz zur Abbildung von handschriftlichen Wortbildern auf beliebige vektorielle Wortrepräsentationen. Aufgrund des wenig bis nicht vorhandenen Zusammenhanges zwischen visuellen und semantischen Eigenschaften bei Wörtern ist die Vorhersage von semantischen im Vergleich zu syntaktischen Wortrepräsentationen erheblich schwieriger [214]. Die Kombination von semantischen und syntaktischen Einbettungen verbessert die Vorhersagefähigkeit auf Basis von Wortbildern [98].

3.3.2 *Named Entity Recognition*

Digitale Bibliotheken enthalten in der Regel eine große Anzahl von Dokumenten, deren Textinhalte in einem unstrukturierten Format vorliegen. Aufgrund dieses Formats ist die Echtzeitberechnung von Suchergebnissen über eine Schlüsselwortsuche oft ineffizient und praktisch nicht realisierbar. In Analogie zu Websuchmaschinen hat sich daher der Aufbau eines Suchindexes in digitalen Bibliotheken als unverzichtbar erwiesen. Damit können Suchanfragen in Sekundenschnelle durchgeführt werden, was den Zugriff auf relevante Informationen erheblich beschleunigt und die Nutzerzufriedenheit erhöht. Es hat sich gezeigt, dass benannte Entitäten eine der am häufigsten verwendeten Suchanfragen in digitalen Bi-

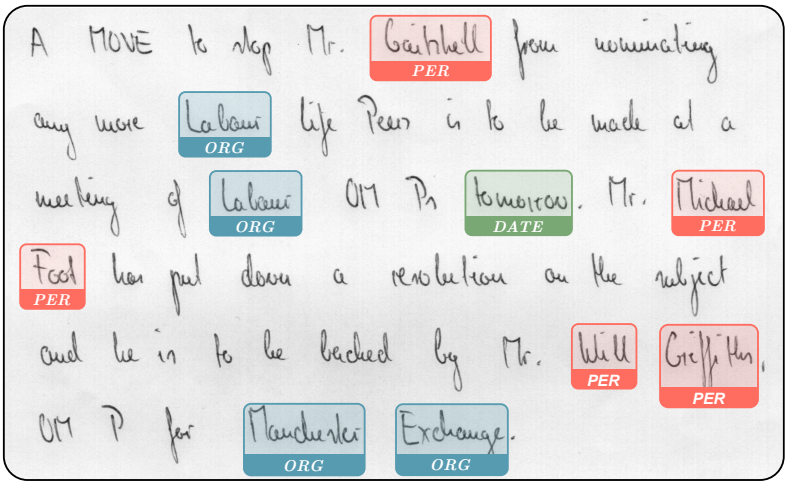


Abbildung 3.3: Ein Beispiel für die Extraktion von benannten Entitäten auf einem handschriftlichen Dokumentenbild aus der IAM-Datenbank.

bliotheken sind [232]. Bei NEs handelt es sich um Objekte aus der realen Welt, wie z.B. Personen, Orte, Organisationen und Produkte, die mit einem Eigennamen bezeichnet werden können. Die Identifizierung und Klassifizierung von NEs in Dokumenten ist daher von entscheidender Bedeutung für die effektive Organisation und den Zugang zu Informationen in digitalen Bibliotheken. Neben der Indexierung können NEs für die Entwicklung effizienterer Suchalgorithmen und den Aufbau von Wissensdatenbanken genutzt werden. Insgesamt ermöglicht die Identifikation von Entitäten in Dokumenten eine effizientere Verwaltung von Bibliotheksbeständen und eine schnelle, intuitive sowie interaktive Navigation durch deren Inhalte.

Die Extraktion von NEs aus maschinenlesbaren Texten ist ein traditionelles und anspruchsvolles Forschungsgebiet des NLP und wird in der Literatur als *Named Entity Recognition* (NER) bezeichnet. Hierbei handelt es sich um ein sequentielles *Labeling*-Problem, bei dem die Detektion und Klassifikation von NEs in einer gegebenen Folge von maschinenlesbaren Wörtern angestrebt wird. Für die Klassifikation steht eine vordefinierte Menge von Entitätsklassen $\mathbf{E}$ sowie eine Art Rückweisungsklasse O zur Verfügung. Die Rückweisungsklasse wird für alle Wörter verwendet, die nicht zu einer der vordefinierten Klassen aus $\mathbf{E}$ zugeordnet werden können. Somit wird jedem Wort aus der Eingabesequenz genau eine der Klassen aus $\mathbf{E} \cup \{O\}$ zugewiesen. Insbesondere ist in den letzten Jahren das Interesse an der Extraktion von NEs aus Dokumentenbildern erheblich gestiegen [36, 48]. Im Gegensatz zu maschinenlesbaren Texten weisen Dokumentenbilder im Allgemeinen eine deutlich höhere Variabilität auf und enthalten neben den textuellen Daten auch visuelle und strukturelle Eigenschaften. Abbildung 3.3 zeigt beispielhaft die Komplexität dieser Aufgabe für ein handschriftliches Dokumentenbild. Die veröffentlichten NER-Ansätze aus der Literatur können entsprechend ihrer Ausrichtung auf maschinell gedruckten oder handschriftlichen Dokumentenbildern unterteilt werden. Im Folgenden wird ein allgemeiner Überblick über die wichtigsten Fortschritte im Forschungsbereich der NER auf maschinenlesbaren Texten und maschinell gedruckten Dokumentenbildern präsentiert. Anschließend wird ein detaillierter Überblick der veröffentlichten NER-Ansätze auf handschriftlichen Dokumentenbildern vorgestellt.

Maschinenlesbare Texte

Traditionelle NER-Methoden basieren hauptsächlich auf manuell erstellten Regeln, Wörterbüchern, orthografischen Merkmalen oder Ontologien [242]. Das Hauptproblem dieser Ansätze ist ihre mangelnde Generalisierbarkeit, da sie auf bestimmte Domänen beschränkt sind und für ihre Entwicklung Expertenwissen erfordern. Statistische Modelle wie das HMM und *Conditional Random Field* (CRF) beheben diese Einschränkungen und erzielen signifikante Fortschritte in der NER [228]. In den letzten Jahren ist eine Vielzahl von Ansätzen mit tiefen neuronalen Netzwerken publiziert worden, welche die Erkennungsgenauigkeiten auf den meisten Benchmark-Datensätzen erheblich verbessern [48]. Insbesondere Kombinationen aus RNNs und CRFs haben sich bewährt [102]. Eine fundamentale Leistungssteigerung wird durch die Integration von vortrainierten semantischen Worteinbettungen in NER-Modellen erreicht [149]. Die Ersetzung von LSTMs- durch Transformerbasierte Architekturen führt auf den meisten NER-Benchmarks zu marginalen Verbesserungen, erfordert jedoch einen deutlich höheren Ressourceneinsatz [177, 244]. Die Transformer zeichnen sich insbesondere bei langen Eingabesequenzen durch eine bessere Modellierung der Kontextinformationen aus [244]. Die Leistung der NER-Modelle kann auf den meisten Benchmarks durch spezielle Anpassungen des Attention-Mechanismus [243] und einer geschickten Kombination von Worteinbettungen [225] weiter optimiert werden. Für einen detaillierten Überblick über das Forschungsgebiet der NER im maschinenlesbaren Bereich siehe [242].

Maschinell gedruckte Dokumentenbilder

Die ersten Ansätze zur semantischen Analyse von maschinell gedruckten Dokumentenbildern verfolgen einen zweistufigen Ansatz. Hierbei wird zunächst das Dokumentenbild mit einem *Optical Character Recognition* (OCR)-Verfahren in maschinenlesbaren Text überführt und anschließend auf der Grundlage des Erkennungsergebnisses die NER-Aufgabe gelöst [48]. Die Anzahl der Erkennungsfehler für moderne und rein textuelle Dokumentenbilder ist in der Regel so niedrig, dass die OCR-Fehler nur geringe negative Auswirkungen auf die Leistung des maschinenlesbaren NER-Verfahrens haben [78]. Dies gilt jedoch nicht für historische Dokumente, die aufgrund von Alterungsprozessen häufig Bildbeeinträchtigungen und generell eine hohe Variabilität aufweisen. Diese Eigenschaften führen häufig zu einer hohen Fehlerrate bei der Texterkennung, welche einen negativen Einfluss auf die Ergebnisse von maschinenlesbaren NER-Modellen haben [65, 66].

Der zweistufige Ansatz ignoriert visuelle und strukturelle Informationen aus Dokumentenbildern und extrahiert die Entitäten ausschließlich auf der Grundlage der textuellen Eingaben des OCR-Verfahrens. Dies ist speziell für komplexe Dokumentenbilder mit Strukturinformationen, wie z.B. Formulare, problematisch. Bei den Entitäten dieser Dokumente handelt es sich hauptsächlich um sogenannte *Key-Value*-Paare, die häufig nur über ihre strukturellen Eigenschaften identifiziert werden können. Aus diesen Gründen haben sich robustere Verfahren für die Extraktion von Entitäten in Dokumentenbildern etabliert, welche zusätzlich zu den textuellen auch strukturelle Informationen berücksichtigen [23, 58, 72]. Diese multimodalen Ansätze basieren auf Transformer- [10, 58, 90, 240] oder Graph-Architekturen [23, 38, 59] und erhalten die strukturellen und textuellen Daten von einem externen OCR-Verfahren.

In den graphbasierten Modellen bilden die extrahierten Wörter die Knoten und die Entitätsbeziehungen zwischen den Wörtern die Kanten. Eine Kante zwischen zwei Wörtern bedeutet in diesem Kontext, dass diese zu derselben Entitätsklasse gehören. Das Lernen der Kanten wird mit einem *Graph Neural Network* (GNN) erreicht [38, 59]. Graphbasierte Ansätze erzielen im Kontext der Informationsextraktion mit wenigen Parametern annähernd *state-of-the-art* Ergebnisse [23], werden aber von Transformer-basierten Ansätzen auf den meisten Benchmarks übertroffen [147]. Diese Transformer-Modelle, kodieren die textuellen und strukturellen Informationen in kontinuierliche Vektorrepräsentationen. Dabei werden die textuellen Daten in semantische Worteinbettungen aus dem NLP-Bereich überführt [75]. Diese Informationen dienen als Eingabe für einen mehrschichtigen Transformer-Encoder, der die Beziehungen zwischen den Modalitäten modelliert und mit selbstüberwachten Aufgaben vortrainiert wird [9, 75, 147]. Zur Informationsextraktion wird ein MLP auf die Ausgabe des Transformers angewendet, wobei die Anzahl der Neuronen in der letzten Schicht der Anzahl der potentiellen Entitätsklassen entspricht.

Eine weitere relevante Eingabemodalität sind die visuellen Eigenschaften von Dokumenten. Dafür werden Merkmale des Dokumentenbildes mit einem CNN extrahiert und als Eingabe für den Transformer-Encoder verwendet. Durch die Berücksichtigung von visuellen Eigenschaften als weitere Eingabemodalität kann die Leistung auf den meisten Benchmarks signifikant verbessert werden [9]. Die Untersuchung verschiedener Strategien zur Verarbeitung dieser Modalitäten bietet eine interessante Forschungsfrage. Ein gemeinsames Modell zur Verarbeitung textueller und visueller Eingaben erweist sich gegenüber getrennten Modellen für jede Modalität als vorteilhaft [9]. Dies ist wahrscheinlich auf die enge Verbindung zwischen den Modalitäten und auf die frühe Möglichkeit der Interaktion zwischen diesen Modalitäten im Modell zurückzuführen.

Auch wenn der zweistufige Ansatz für die meisten Benchmarks zu state-of-the-art Ergebnissen führt, weist dieser auch eine Reihe von Nachteilen auf. Ein fundamentales Problem des Ansatzes ist zum einen der hohe Ressourcenaufwand und zum anderen die Propagierung von Erkennungsfehlern des externen OCR-Verfahrens. Zur Lösung dieser Probleme haben sich in den letzten Jahren Ende-zu-Ende-Verfahren etabliert, welche auf eine explizite Texterkennung verzichten [13, 39, 44, 90, 113, 247]. Die Ende-zu-Ende-Modelle weisen im Gegensatz zu zweistufigen Ansätzen eine hohe Robustheit gegenüber OCR-Fehlern auf und verringern die benötigen Ressourcen erheblich [10, 44, 90]. Der erste Schritt dieser Ansätze besteht in der Extraktion von visuellen Informationen aus dem Eingabebild und deren Überführung in eine kompakte zweidimensionale Merkmalsrepräsentation. Dafür verwenden die Ansätze aus der Literatur sowohl Transformer- als auch CNN-basierte Modelle [39, 44, 90]. Ein Transformer-basierter Decoder erzeugt auf der Grundlage der Merkmale eine zeichenweise Ausgabe, die schließlich in ein gewünschtes strukturiertes Format umgewandelt werden kann. Dabei enthält die Ausgabe für jede gefundene Entität dessen Transkription zusammen mit der vorhergesagten Entitätsklasse. Während des Trainings benötigen die OCR-freien Modelle jedoch üblicherweise ebenfalls die textuellen Gold-Standard Annotationen der Dokumentenbilder.

Der aktuelle Trend bei der Entwicklung von multimodalen Transformer-Architekturen tendiert zu universellen Sprachmodellen für die semantische Analyse von Dokumentenbildern [39, 75, 90, 138, 199, 247]. Hierbei werden die Modelle auf einer großen Anzahl von Dokumenten mit selbstüberwachten Aufgaben vortrainiert und anschließend auf die eigentliche Aufgabe angepasst. Für einen detaillierten Überblick über das Forschungsgebiet der NER im maschinell gedruckten Bereich siehe [36, 48].

Handschriftliche Dokumentenbilder

Analog zu Ansätzen für maschinell gedruckte Dokumentenbilder werden für handschriftliche Dokumente zweistufige Methoden verwendet, welche das Dokumentenbild zunächst in maschinenlesbaren Text umwandeln und anschließend ein NER-Verfahren auf diesen Text anwenden [54, 141]. Erste Ansätze aus der Literatur verwenden zur Klassifikation von Entitäten aus dem maschinenlesbaren Text *Bidirectional Long Short-Term Memory* (BLSTM)-Modelle, CRF-Modelle oder Reguläre Ausdrücke [54, 153]. Aufgrund der hohen Variabilität von Handschriften ist die Verarbeitung von handgeschriebenen Dokumenten jedoch im Allgemeinen schwieriger als die von maschinell gedruckten Dokumentenbildern. Dies führt in der Regel zu einer höheren Anzahl von Erkennungsfehlern und vergrößert damit das Problem der Fehlerpropagierung. Daher ist die Entwicklung von robusteren Ansätzen für die semantische Analyse von handgeschriebenen Dokumentenbildern erforderlich.

Eine Alternative zu dem zweistufigen Ansatz ist die Verwendung von integrierten Modellen, welche die Handschrifterkennung und NER-Aufgabe in einem gemeinsamen Schritt durchführen [22, 24, 153, 165, 200, 201]. Die integrierten Ansätze basieren auf unterschiedlichen Voraussetzungen hinsichtlich des erwarteten Eingabeformates. Dabei existieren sowohl Methoden, die ohne eine explizite Segmentierung auskommen [22, 165, 200], als auch solche, die eine externe Segmentierung auf Wort-, Zeilen- oder Absatzebene [24, 43, 153, 201] erfordern. In der Literatur existieren mehrere Arbeiten für den Vergleich von zweistufigen und integrierten NER-Ansätzen auf handschriftlichen Dokumentenbildern [19, 43, 201]. Die Ergebnisse dieser Arbeiten sind jedoch nicht übereinstimmend. Im Allgemeinen zeigt sich, dass integrierte Ansätze bei teilstrukturierten Dokumenten bessere Ergebnisse erzielen als zweistufige Ansätze, während bei unstrukturierten Texten das Gegenteil der Fall ist. Die Architektur der integrierten Ansätze ohne explizite Segmentierung basiert im Allgemeinen auf dem Encoder-Decoder-Prinzip. Dazu wird das Dokumentenbild zunächst mit einem CNN in eine zweidimensionale Merkmalskarte überführt. Auf der Grundlage dieser Merkmalsrepräsentation werden separate, aufgabenspezifische Decoder für die Handschrifterkennung und NER-Aufgabe angewendet und gemeinsam optimiert [22]. Die Kombination von Transkriptionen und NE-*Labeln* in einer gemeinsamen Ausgabe hat sich als leistungsfähige Alternative zur getrennten Vorhersage erwiesen [24, 165, 200]. Daher wird der Decoder in aktuellen Ansätzen auf eine verschachtelte Vorhersage der Transkriptions- und Entitätsdaten trainiert. Die gewünschte Ausgabe umfasst die Transkription des handgeschriebenen Textes im Bild zusammen mit den entsprechenden NE-Kategorien, z.B. *Mein Name ist <PERSON> Oliver Tüselmann </PERSON>*. Für die Dekodierung werden BLSTM- [43] oder Transformer-Modelle [200] verwendet und auf die Texterkennung mit speziellen Zeichen für die NEs trainiert. Die Integration eines Zustandsautomaten im Dekodierungsschritt kann zudem eine syntaktische Korrektheit der vorhergesagten NE-Label bei geringerem Mehraufwand gewährleisten [219]. Analog zu den segmentierungsfreien Ansätzen überführen die zeilensegmentierten Ansätze aus der Literatur [24, 201] die gegebene Textzeile mit einem CNN in eine zweidimensionale Merkmalsrepräsentation. Anschließend werden die Beziehungen zwischen den Merkmalen mit einem BLSTM-Modell kodiert und eine verschachtelte Vorhersage der Transkriptions- und Entitätsdaten als Ausgabe generiert. Die Verwendung des Attention-Mechanismus im LSTM-basierten Decoder erzielt auf den meisten Benchmarks signifikante Verbesserungen [201]. Dies ist wahrscheinlich auf die Fokussierung von bestimmten Bildbereichen während des Dekodierprozesses zurückzuführen.

Eine weitere vielversprechende Alternative zu den zweistufigen und integrierten Ansätzen bieten sogenannte HTR-freie Verfahren. Diese Ende-zu-Ende-Modelle vermeiden eine explizite Texterkennung und bestimmen die Entitäten ausschließlich auf Grundlage des Dokumentenbildes. Erste Lösungsansätze in diesem Bereich fokussieren sich auf die Erkennung von NEs in handgeschriebenen Dokumentenbildern mit manuell erstellten Regeln [1]. Diese wortsegmentierten Ansätze erlauben jedoch lediglich die Detektion von NEs ohne eine Klassifikation der Wortbilder in vordefinierte Kategorien durchzuführen. Ein erster Ansatz zur Detektion und Klassifikation von NEs auf Basis von Wortbildern verwendet ein zum PHOCNet ähnliches CNN, wobei die letzte Schicht des MLPs der Anzahl der NE-Klassen entspricht [207]. Eine Erweiterung des Ansatzes kann durch die Berücksichtigung von Kontextinformationen erreicht werden. Hierbei wurde zunächst eine *Bigram*-inspirierte Architektur [206] veröffentlicht, welche das aktuelle Wortbild und das vorhergesagte NE-Label des vorherigen Wortbildes als Eingabe erhält und damit das NE-Label für das aktuelle Wortbild vorhersagt. Auch wenn die Leistung durch die Berücksichtigung des vorherigen Wortbildes verbessert werden kann, ignoriert der Ansatz wichtige Kontextinformationen. Die Ersetzung der Bigram-Architektur mit einem BLSTM ermöglicht die Berücksichtigung aller vorherigen Wortbilder bei der Klassifikation des NE-Labels für das aktuelle Wortbild und zudem das Lernen von relevanten Beziehungen zwischen Wörtern [2, 166, 206]. Die von Toledo et al. in [206] vorgestellte Architektur besteht aus der Kombination eines zum PHOCNet ähnlichen CNNs und eines zweischichtigen BLSTMs. Das Modell wird Ende-zu-Ende mit der Kreuzentropie Verlustfunktion auf die Vorhersage von NE-Labeln für die Wortbilder aus einem gegebenen Dokument trainiert. Dabei werden die Wortbilder sequentiell bezüglich ihres Auftretens im Dokument verarbeitet. Eine Optimierung der Ende-zu-Ende-Architektur wird von Rowtula et al. in [166] vorgestellt. Diese Architektur besteht aus der Kombination eines ResNets und eines zweischichtigen BLSTMs. Die Autoren dieses Modells beobachteten zudem, dass NEs mit der Position und Verteilung von *Part-of-Speech* (PoS)-Labeln in einem Satz in Verbindung stehen. Daher wird das Modell zunächst auf die Vorhersage von PoS-Labeln trainiert und anschließend auf die NEs spezialisiert. Ein weiterer Optimierungsansatz des Ende-zu-Ende-Verfahrens kann durch die gemeinsame Betrachtung aller segmentierten Wortbilder während der Merkmalsextraktion erreicht werden [2]. Der Leistungsgewinn wurde bisher jedoch auf nur wenigen Benchmarks nachgewiesen und benötigt eine ressourcenaufwendige CNN-Architektur, welche ab einer bestimmten Sequenzlänge unpraktikabel wird.

Kürzlich wurde mit dem *Document end-to-end self-supervised understanding and recognition transformer* (Dessurt) [39] ein Modell zum universellen Verständnis von handschriftlichen Dokumentenbildern publiziert. Hierbei handelt es sich um einen Encoder-Decoder-basierten Ende-zu-Ende-Ansatz auf Dokumentenebene, welcher die Textsegmentierung und Texterkennung implizit lernt. Das Dessurt-Modell erhält ein Graustufenbild, sowie einen Aufgabenstring als Eingabe und generiert autoregressiv einen aufgabenspezifischen Text als Ausgabe. Das Dokumentenbild wird zunächst mit einem CNN-basierten Encoder in eine zweidimensionale Merkmalsrepräsentation überführt und anschließend mit gelernten zweidimensionalen räumlichen Einbettungen kombiniert. Der Eingabetext wird in *Tokens* umgewandelt und diese in zufällig initialisierte, aber lernbare Einbettungen mit der gleichen Dimensionalität wie ein Merkmal des Bildes überführt. Die Modalitäten werden separat mit Transformern verarbeitet und die Interaktion zwischen den Modalitäten mit *Cross-Attention*-Schichten [123] erreicht. Die Ausgabe erfolgt autoregressiv durch die Auswahl des wahrscheinlichsten Zeichens bei jedem Dekodierungsschritt. Das Modell wird selbstüberwacht auf einer Vielzahl von Aufgaben vortrainiert. Hierbei werden unter ande-

rem die maskierte Sprachmodellierung und HTR-basierte Aufgaben wie z.B. die Bestimmung der Position eines gegebenen Textes im Bild oder die Transkription eines gegebenen Textausschnitts eingesetzt. Für das Training werden synthetisch generierte Dokumentenbilder auf der Grundlage von *Wikipedia*-Texten mit über 10000 Schriftarten und synthetisch erstellten Dokumentenbildern von Tabellen und Formularen verwendet. Zudem findet eine Destillation von Wissen aus einem vortrainierten Sprachmodell statt. Dafür wird eine Art Pseudolösung mit einem vortrainierten BART-Modell [112] für einen gegebenen Text und einer Aufgabe bestimmt und diese für das Training des Dessurt-Modells auf der Grundlage eines synthetisch generierten Dokumentenbildes des Textes verwendet.

3.3.3 *Question Answering*

Die Retrieval-basierte Suche ist ein bewährter Ansatz für das Auffinden von Informationen in großen Dokumentensammlungen. Ein wesentlicher Nachteil dieses Ansatzes besteht jedoch darin, dass der Benutzer die Antwort auf seine Anfrage manuell aus der Ergebnisliste herausfiltern muss. Darüber hinaus ist es oft notwendig, die Anfrage in Schlüsselwörter umzuwandeln, damit eine Suchmaschine die relevantesten Ergebnisse findet. Um diesen zeitaufwändigen und fehleranfälligen Prozess zu vermeiden, fordern Nutzer von modernen Suchmaschinen ein natürlichsprachliches Ein- und Ausgabeformat des Systems. Diese anspruchsvolle Aufgabe wird in der Forschung als *Question Answering* (QA) bezeichnet. Dabei können QA-Systeme den Zeitaufwand für die Suche nach relevanten Informationen erheblich reduzieren, da der Nutzer direkte Antworten auf seine Fragen erhält, ohne die relevanten Ressourcen manuell durchsuchen zu müssen. Darüber hinaus ist die Integration von QA-Systemen in digitale Bibliotheken ein Beispiel für den Einsatz fortschrittlicher Technologien zur Verbesserung der Informationsvermittlung. Dies kann dazu beitragen, die Bibliothek als wichtigen Bestandteil des digitalen Zeitalters zu positionieren und ihre Relevanz für die Nutzer zu erhöhen. Ausgehend von diesen Vorteilen zeigt das Forschungsgebiet der Dokumentenbildanalyse insbesondere in den letzten Jahren ein wachsendes Interesse an der QA-Aufgabe [103, 133, 134, 205]. Das Hauptziel in diesem Bereich ist die Beantwortung von Fragen auf der Grundlage von Wissen, das in einer vorgegebenen Sammlung von Dokumentenbildern enthalten ist. Eine besondere Herausforderung stellen dabei die multimodalen Eigenschaften der Dokumentenbilder dar. Diese enthalten neben textuellen Daten auch strukturelle und visuelle Informationen, die für die Beantwortung der Fragen relevant sind. Die derzeit verfügbaren QA-Modelle für Dokumentenbilder basieren weitgehend auf Konzepten für maschinenlesbare Eingaben. Im Folgenden wird daher zunächst ein Überblick über die wichtigsten Fortschritte auf dem Gebiet des QAs im NLP-Bereich gegeben. Aufgrund der zahlreichen Forschungsaktivitäten in diesem Bereich und der Fokussierung dieser Arbeit auf Dokumentenbilder ist der Literaturüberblick stark verallgemeinert und erhebt keinen Anspruch auf Vollständigkeit. Für einen detaillierten Überblick über das Forschungsgebiet des textuellen QAs siehe [14, 252]. Nach der Übersicht über maschinenlesbare Ansätze wird ein detaillierter Literaturüberblick zu QA-Verfahren auf der Basis von Dokumentenbildern gegeben.

Maschinenlesbare Texte

Die QA-Ansätze für maschinenlesbare Texte werden im Allgemeinen in extrahierende und generierende Modelle unterteilt [14]. Der generative Ansatz realisiert die anspruchsvollere

Anwendung, da die Ausgabe des Modells nicht auf den gegebenen Kontext beschränkt ist, sondern beliebige natürlichsprachliche Antworten erzeugen kann. Der extrahierende Ansatz wird in der Literatur als MRC-Verfahren bezeichnet und basiert auf der Annahme, dass die Antwort auf die Frage im gegebenen Kontext vorhanden ist. Es werden somit keine Antworten generiert, sondern lediglich ein Antwortbereich aus dem Kontext extrahiert, der die Antwort auf die Frage enthält [14, 249]. Da der primäre Fokus dieser Arbeit auf dem MRC-Verfahren beruht, beschränkt sich der folgende Literaturüberblick ausschließlich auf diesen Forschungsbereich.

Traditionelle MRC-Ansätze basieren auf manuell erstellten Regeln oder Merkmalen [121, 249]. Der grundsätzliche Nachteil dieser Ansätze ist die domänenspezifische Ausrichtung, wodurch für jedes neue Anwendungsgebiet ein neues Regelwerk aufwendig erstellt werden muss. Der Einsatz von tiefen neuronalen Modellen beseitigt diese Einschränkung und führt zu signifikanten Verbesserungen gegenüber klassischen Methoden auf den meisten Benchmarks [14]. Die allgemeine Vorgehensweise bei den neuronalen QA-Ansätzen besteht aus einer Einbettungs-, Analyse- und Vorhersagephase. In der Einbettungsphase werden zunächst die Kontext- und Fragewörter in vektorielle Repräsentationen überführt. Dafür wurden in der Literatur unter anderem *One-Hot*-Kodierungen und statische Wortrepräsentationen eingesetzt [14, 121]. Aktuelle Ansätze basieren auf kontextsensitiven Worteinbettungsverfahren, welche die Ambiguität von Wörtern berücksichtigen [14]. In der Analysephase wird die Relevanz der Kontextwörter für die Beantwortung der Frage ermittelt. Dazu wird die Interaktion zwischen den Frage- und Kontextrepräsentationen im Allgemeinen mit RNN- und Transformer-Architekturen modelliert [14]. Die Modellierung der Relevanz zwischen dem Kontext und der Frage kann entweder unidirektional oder bidirektional bestimmt werden [14]. Insbesondere BLSTM-basierte Modelle mit einem Attention-Mechanismus werden in diesem Bereich häufig verwendet [180]. In der Vorhersagephase wird schließlich die Antwort aus dem Kontext extrahiert. Hierbei wird die Start- und die Endposition der Antwort mit einem LSTM oder MLP in Kombination mit einer Softmax-Funktion geschätzt [42, 149, 180]. Das Ergebnis ist jeweils eine Pseudowahrscheinlichkeitsverteilung über die Kontextwörter für die Start- und die Endposition. Neben der klassischen Pipeline dominieren in der Literatur Ende-zu-Ende-Architekturen, welche die Frage- und Kontextwörter in einem gemeinsamen Transformer-Modell kodieren [42, 243]. Diese Modelle werden zunächst auf großen Textkorpora selbstüberwacht vortrainiert und anschließend auf die QA-Aufgabe angepasst. Dazu werden die gelernten Merkmale der Kontextwörter separat mit einem gemeinsamen MLP verarbeitet und die Pseudowahrscheinlichkeiten für die Start- und Endposition der Antwort berechnet. Die Transformer-Modelle führen zu state-of-the-art Ergebnissen auf den meisten MRC-Benchmarks und können durch geeignete Adaptionen der klassischen Transformer-Verfahren weiter verbessert werden [14, 243]. Ein Beispiel für eine Erweiterung ist der entitätsbasierte Attention-Mechanismus, der die Entitätsklassen der Eingabewörter bei der Berechnung der Attention-Werte berücksichtigt [243].

Der Kontext der im bisherigen Literaturüberblick vorgestellten Ansätze basiert auf einem einzelnen Dokument bzw. einer Textpassage. Die Beantwortung von Fragen auf der Grundlage einer Sammlung von Dokumenten ist jedoch realitätsnäher [27]. Klassische Ansätze in diesem Bereich bestehen aus Effizienzgründen aus einer Kombination von Dokumentenretrieval und Antwortextraktion. Beim Retrieval wird die gegebene Dokumentensammlung zunächst auf wenige relevante Dokumente reduziert [252]. Dazu bestimmt der Retriever für jedes Dokument der Sammlung dessen Relevanz für die Beantwortung der Frage. Ein verbreitetes Verfahren besteht aus der separaten Überführung der Frage

und des Dokuments in eine vektorielle Repräsentation, wodurch die Relevanz mit einer vektoriellen Distanzberechnung bestimmt wird [88, 109, 179]. Dieser Ansatz ist zwar effizient, erlaubt jedoch keine direkte Interaktion zwischen den Frage- und Kontextwörtern. Daher haben sich Modelle etabliert, welche die Frage- und Dokumentenwörter entweder direkt in einem gemeinsamen Modell verarbeiten [42] oder diese zunächst separat modellieren und die Repräsentationen anschließend in ein gemeinsames Modell zur Bestimmung der Ähnlichkeit überführen [226]. Die Extraktion der Antwort aus einer Sammlung von Dokumenten basiert in der Regel auf MRC-Verfahren mit nur einem Dokument [252]. Ein verbreiteter Ansatz berechnet für jedes relevante Dokument einen Antwortbereich mit einem zugehörigen Konfidenzwert und gibt den Bereich mit der höchsten Konfidenz als endgültige Antwort aus [227]. Die gemeinsame Betrachtung aller relevanten Dokumente erfordert zwar mehr Ressourcen, führt jedoch bei den meisten Benchmarks zu signifikant besseren Ergebnissen [226]. In der Literatur existieren zudem Ende-zu-Ende-Modelle, welche ein gemeinsames Training der Retrieval- und Antwortmodelle ermöglichen [252]. Der aktuelle Trend zur Beantwortung von Fragen auf großen Dokumentenkollektionen tendiert zu generativen Transformer-Modellen [21]. Diese Modelle sind in der Lage, eine große Menge an Wissen aus den Kollektionen in ihren Parametern zu speichern und so die Frage ohne ein separates Retrieval effizient zu beantworten [21, 155].

Maschinell gedruckte Dokumentenbilder

Die Beantwortung natürlichsprachlicher Fragen auf der Basis von Dokumentenbildern ist aufgrund ihrer multimodalen Eigenschaften und vielfältigen Layouts eine anspruchsvolle Aufgabe. Erste Benchmarks in diesem Bereich beschränken sich auf synthetisch generierte Diagrammbilder mit vordefinierten Fragestrukturen [86]. Diese Benchmarks werden der realen Komplexität von QA-Aufgaben jedoch nicht gerecht, da die meisten Fragen mit „Ja“ oder „Nein“ beantwortet werden können und die Diagramme nur einen geringen textuellen Anteil enthalten. Ein entscheidender Meilenstein in diesem Bereich wurde durch die Veröffentlichung des *DocVQA*-Datensatzes [134] und der Durchführung der zugehörigen *DocVQA-Challenge* im Jahr 2020 erreicht [135]. Der Datensatz enthält reale Dokumente mit komplexen Layouts, wie z.B. Formulare, Briefe, Rechnungen und teilweise handschriftlich verfasste Dokumente. Aufgrund der hohen Komplexität dieser Daten ist zur Beantwortung der Fragen ein grundlegendes Verständnis von Dokumentenbildern erforderlich. Zur Vereinfachung beschränkt sich der Benchmark zunächst auf die MRC-Aufgabe, bei der ein einzelnes Dokumentenbild als Kontext dient und die Lösung im Allgemeinen ein Textauszug aus diesem Dokument ist. Die sukzessive Erhöhung der Komplexität von QA-Benchmarks für Dokumentenbilder [104, 132, 133, 198] und die Durchführung von Wettbewerben [103, 205] führt zu weiteren Innovationen und eine Annäherung an reale Bedingungen in diesem Bereich. Dafür stehen spezielle QA-Datensätze mit den Schwerpunkten auf Infographiken [132] und mehrseitigen Dokumentenbildern [104, 198] zur Verfügung.

Initiale Ansätze in diesem Bereich verwenden ein sequenzielles Verfahren aus einem OCR-Modell und einem textuellen QA-Modell [135]. Hierbei wird das gegebene Dokument zunächst mit einem OCR-Verfahren in einen maschinenlesbaren Text umgewandelt und anschließend ein rein textuelles QA-Modell aus dem NLP-Bereich zur Beantwortung der Frage verwendet. Obwohl die veröffentlichten Verfahren eine Vielzahl der Fragen des Benchmarks korrekt beantworten, weisen sie Defizite bei Dokumenten mit strukturell kodierten Informationen auf [135]. Im Vergleich zu den rein sequentiellen Ansätzen erzielen multi-

modale Verfahren [10, 75, 90, 147, 152, 241], die neben textuellen auch strukturelle Daten berücksichtigen, signifikant bessere Ergebnisse auf den meisten Benchmarks. Insbesondere erzielen Verfahren auf Basis von multimodalen Transformer-Modellen state-of-the-art Resultate [10, 90]. Hierbei werden zunächst die visuellen, textuellen und räumlichen Informationen eines Dokumentenbildes extrahiert und in eine vektorielle Merkmalsrepräsentation überführt. Die im Dokumentenbild enthaltenen Texte und deren räumlichen Positionen werden in den meisten Modellen von externen OCR-Ansätzen extrahiert [9, 10, 58, 72, 75, 152, 199, 240, 241]. Die Extraktion der Bildmerkmale basiert auf bewährten Modellen aus dem CV-Bereich, wie z.B. CNNs [9], *U-Nets* [152] und *Region-Proposal*-Netzwerken [147, 241]. Diese rechenintensiven Objekterkennungsmodelle sind jedoch nicht zwingend erforderlich, sondern können häufig ohne Leistungsverluste durch eine Architektur mit wenigen Faltungsschichten ersetzt werden [10, 75]. Die extrahierten Merkmale des gegebenen Dokuments dienen als zusätzliche Eingabe für das Transformer-Modell. Dieses Modell wird bezüglich mehrerer selbstüberwachter Aufgaben vortrainiert, um die multimodalen Merkmale zu kombinieren und aufeinander abzustimmen. Zu den häufigsten Aufgaben gehört die maskierte Sprachmodellierung (engl.: *Masked Language Modelling* (MLM)), die Text-Bild-Zuordnung und die visuelle Rekonstruktion des Dokumentenbildes. Nach dem Vortraining wird das Modell vollüberwacht auf die QA-Aufgabe angepasst. Im Allgemeinen werden die veröffentlichten Ansätze hinsichtlich der Ausgabefunktionalität unterschieden, wobei die Antwort entweder generiert oder extrahiert wird. Bei generativen Modellen [39, 90] wird zusätzlich zum Encoder ein autoregressiver Decoder trainiert, der die Antwort in einem meist strukturierten, anwendungsspezifischen Format ausgibt. Bei extrahierenden Modellen [42, 149] wird auf die kodierten Kontextwörter des Encoders ein MLP mit zwei Neuronen in der Ausgabeschicht angewendet, welche die Start- und Endposition der Antwort definieren. Eine weitere Unterscheidung sind die verwendeten Modalitäten und die Art des Vortrainings von QA-Ansätze aus der Literatur [147, 240]. Äquivalent zum NER-Bereich besteht der generelle Trend zu universellen Sprachmodellen, die nicht nur exklusiv für die QA-Aufgabe entwickelt und trainiert werden [39, 75, 90, 138, 199, 247]. Hierbei werden insbesondere multimodale OCR-freie Modelle entwickelt, die state-of-the-art Resultate auf den meisten QA-Benchmarks erzielen [39, 90, 110, 239].

Handschriftliche Dokumentenbilder

In der Literatur existieren nur wenige Arbeiten und Benchmarks, die sich mit der Beantwortung von Fragen im Zusammenhang mit handgeschriebenen Dokumentenbildern befassen. Die zwei verfügbaren Benchmarks in der Literatur basieren auf einer synthetisch generierten bzw. historischen Dokumentensammlung [221]. Dabei wurden lediglich ein HTR-freies und ein HTR-basiertes QA-Modell aus [133] sowie das Dessurt-Modell [39] auf diesen Benchmarks evaluiert. Das Dessurt-Modell wurde jedoch ausschließlich auf dem synthetischen Benchmarks getestet und beschränkt sich zudem auf die Beantwortung anhand eines einzelnen Dokumentenbildes anstelle einer Sammlung von Dokumentenbildern. Das Modell bietet einen universellen Ansatz für die semantische Analyse von Dokumentenbildern, sodass nur die Ein- und Ausgabeformate für das QA definiert werden müssen. Mit dem vorgegebenen Format und einem Trainingsdatensatz wird das Modell dann an die QA-Aufgabe angepasst, ohne dass die Architektur geändert werden muss. Hierbei dienen das Dokumentenbild und die kodierte Frage als Eingabe für das Modell, das die Antwort autoregressiv in einem maschinenlesbaren Format erzeugt. Der von Mathew et

al. in [133] vorgestellte HTR-freie Ansatz unterscheidet sich fundamental vom Dessurt-Modell und ist speziell für handschriftliche Dokumentensammlungen entwickelt worden. Aus Effizienzgründen kombiniert der Ansatz ein Dokumentenretrieval mit einem extrahierenden QA-Modell. Hierbei wird die gegebene Dokumentensammlung zunächst mit einem Retrieval-Modell auf eine geringe Anzahl fragerelevanter Dokumente reduziert und diese anschließend zur Antwortextraktion an das QA-Modell übergeben. Der Retrieval-Ansatz basiert auf der Heuristik, dass die Relevanz eines Dokuments mit der Anzahl der darin enthaltenen Fragewörter zusammenhängt. Dabei wird die Übereinstimmung eines maschinenlesbaren Fragewortes und eines Wortbildes aus dem Dokument analog zur syntaktischen Schlüsselwortsuche über die Ähnlichkeit ihrer Repräsentationen in einem gemeinsamen Einbettungsraum durch das HWNetv2 realisiert. Für eine effiziente Bewertung der Relevanz in großen Dokumentensammlungen basiert das Retrieval-Verfahren auf einer Aggregationsstrategie. Bei diesem Verfahren wird jedes Dokumentenbild sowie die Frage in eine Vektorrepräsentation umgewandelt und die Ähnlichkeit zwischen einem Dokument und der Frage mit der Kosinusähnlichkeit ihrer Repräsentationen bestimmt. Dazu werden die Wortbilder aus dem Dokument bzw. die Wörter aus der Frage zunächst mit dem HWNetv2 in eine Sequenz von Vektorrepräsentationen überführt und anschließend mit dem *Fisher-Vektor-Framework* [148] in einen Vektor aggregiert. Nach der Identifizierung der k relevantesten Dokumente bestimmt das QA-Modell einen Bildbereich aus einem dieser Dokumente, der die Antwort auf die Frage enthält. Dafür werden zunächst die Dokumentenbilder in eine Menge von zweizeiligen Bildausschnitten umgewandelt, die als potentielle Antworten in Erwägung gezogen werden. Für jedes dieser Ausschnitte wird analog zum Retrieval-Ansatz eine aggregierte Vektordarstellung auf der Grundlage der enthaltenen Wortbilder in dem Ausschnitt ermittelt und die Ähnlichkeit mit dem aggregierten Anfragevektor berechnet. Schließlich wird der Ausschnitt mit dem höchsten Ähnlichkeitswert als Antwort zurückgegeben. Im Vergleich zu dem HTR-basierten QA-Modell kann der HTR-freie Ansatz eine deutliche Leistungssteigerung bei dem historischen QA-Benchmark erzielen, ist jedoch auf dem synthetischen QA-Benchmark unterlegen [133].

3.4 DISKUSSION

Die semantische Analyse von Dokumenten ist ein traditionelles Forschungsgebiet des NLP und erfährt insbesondere in den letzten Jahren eine zunehmende Beachtung im Bereich der bildbasierten Dokumentenanalyse [48, 75, 110, 133]. Im Allgemeinen unterscheiden sich die in der Literatur veröffentlichten Modelle für handgeschriebene und maschinell gedruckte Dokumentenbilder grundlegend. Hierbei werden im Handschriftbereich überwiegend Modelle verwendet, die aufgabenspezifisch sind und eine externe Segmentierung des Eingabebildes erfordern, während für maschinell gedruckte Dokumente multimodale Transformer-Modelle dominieren [133, 166, 206]. Hierbei gilt speziell im Handschriftbereich, dass die Leistung von sequentiellen Modellen, die aus einem HTR- und einem NLP-Modell bestehen, im Vergleich zu HTR-freien Modellen auf den meisten Benchmarks signifikant besser sind [133, 215].

Analog zur maschinell gedruckten Domäne werden im Handschriftbereich derzeit multimodale Transformer-Modelle entwickelt [39, 90, 138]. Diese multimodalen Sprachmodelle werden mittels selbstüberwachtem Lernen auf großen Bild- und Textkorpora vortrainiert und anschließend für spezifische Anwendungen angepasst. Aufgrund der Textausgabe sind diese Verfahren sehr flexibel und können zahlreiche Anwendungen mit demselben Modell

lösen, wobei die Aufgaben durch textuelle Anweisungen, sogenannte *Prompts*, differenziert werden [138, 246]. Diese Verallgemeinerung ermöglicht einen Wissenstransfer zwischen verschiedenen Aufgaben und führt zu state-of-the-art Ergebnissen auf diversen Benchmarks im maschinell gedruckten Bereich [138]. Aktuell gibt es nur wenige Erkenntnisse und Ergebnisse zu multimodalen Sprachmodellen auf Datensätzen mit handgeschriebenen Dokumentbildern, sodass die Leistungsfähigkeit und Anwendbarkeit der Modelle in der Handschriftdomäne noch unter Beweis gestellt werden muss.

Die multimodalen Sprachmodelle bringen eine Reihe von Nachteilen und Herausforderungen mit sich. Eine der größten Einschränkungen ist der hohe Bedarf an Rechenressourcen sowohl für das Training als auch für die Inferenz [10, 21, 75, 90]. Dies kann hohe Kosten verursachen und erfordert eine Infrastruktur, die nur wenigen Organisationen und Instituten zur Verfügung steht. Derzeit werden die state-of-the-art multimodalen Sprachmodelle fast ausschließlich von großen Unternehmen entwickelt und teilweise öffentlich zur Verfügung gestellt [246]. Die Trainingsdaten werden dabei nur selten offengelegt, wodurch nicht garantiert werden kann, dass die Testdaten aus öffentlich zugängliche Benchmarks beim Vortraining der Modelle nicht verwendet wurden. Dies ist besonders im akademischen Bereich problematisch. Das Training multimodaler Sprachmodelle erfordert zudem eine große Menge an repräsentativen und annotierten Daten, die insbesondere im handschriftlichen Dokumentenbereich nicht öffentlich verfügbar sind [144]. Auch eine Ergänzung der Trainingsdaten durch synthetisch erzeugte Dokumentenbilder kann den Mangel an verfügbaren realen Daten vermutlich nicht beheben, da die Variabilität realer handschriftlicher Dokumente mit aktuellen Syntheseverfahren nicht hinreichend reproduzierbar ist [237].

Weitere grundsätzliche Probleme multimodaler Sprachmodelle sind die geringe Interpretierbarkeit von Ende-zu-Ende-Modellen und Halluzinationen [246]. Im Zusammenhang mit Sprachmodellen sind Halluzinationen Informationen aus der erzeugten Ausgabe, die nicht auf den Eingabedaten basieren oder faktisch falsch sind [138]. Darüber hinaus ist die eingeschränkte Kontrolle über das Ausgabeformat problematisch, da viele Anwendungen eine strukturierte Ausgabe benötigen und es nicht trivial ist, diese Struktur aus der natürlichsprachlichen Ausgabe zu extrahieren. Dabei muss entweder ein Ansatz zur Konvertierung der Textausgabe in das gewünschte Format entwickelt oder ein aufwendiges fine-tuning des Sprachmodells durchgeführt werden. Ein weiterer wichtiger Aspekt ist die starke Abhängigkeit der Ergebnisqualität vom verwendeten Prompt [120]. Daher ist es für eine effiziente Entwicklung und Anwendung von Modellen notwendig, den Prompt zu optimieren, was unter Umständen mit einem erheblichen Aufwand an Ressourcen verbunden ist [120].

Die Literaturübersicht zeigt, dass die Wahl eines geeigneten Ansatzes zur semantischen Analyse von handschriftlichen Dokumentenbildern weiterhin eine offene Forschungsfrage darstellt. Insbesondere wurde der Grund für die geringere Leistungsfähigkeit von HTR-freien Modellen im Vergleich zu HTR-basierten Modellen ungeachtet ihrer theoretischen Vorteile bislang nicht untersucht. In dieser Arbeit wird die Hypothese aufgestellt, dass dies vermutlich auf das Fehlen von externen semantischen Informationen in HTR-freien Ansätzen zurückzuführen ist. Daraus ergibt sich die zentrale Forschungsfrage, ob die Integration von semantischem Weltwissen in HTR-freie Modelle die Diskrepanz zu sequentiellen Ansätzen beheben kann und die theoretischen Vorteile von HTR-freien Modellen wirksam werden. Die effiziente und robuste Integration von semantischem Vorwissen in HTR-freie Modelle ist jedoch auch eine offene Forschungsfrage. Dabei ist das Standardverfahren auf Basis eines selbstüberwachten Trainings von multimodalen Transformer-Modellen ein vielversprechender Ansatz. Jedoch ist dieses Verfahren aufgrund der domänenspezifischen Her-

ausforderungen von Handschriften in der Praxis schwer zu realisieren und erfordert zudem einen hohen Ressourcenaufwand. Daher sind effizientere Ansätze von großem Interesse, die beispielsweise das bereits kodierte semantische Wissen aus textuellen Wortrepräsentationen nutzen und in geeigneter Weise mit der visuellen Ebene verknüpfen.

4 NEURONALE MODELLE ZUR SEMANTISCHEN DOKUMENTENBILDANALYSE

Nachdem im vorherigen Kapitel die relevanten Anwendungen der semantischen Dokumentenanalyse vorgestellt wurden, werden in diesem Kapitel zwei neuronale Modelle zur Realisierung dieser Anwendungen präsentiert. Der erste Ansatz ist ein zweistufiges Verfahren, das state-of-the-art Modelle aus den Forschungsbereichen der *Computer Vision* und des *Natural Language Processing* kombiniert. Dabei wird das Dokumentenbild zunächst mit einem Modell zur Handschrifterkennung (engl.: *Handwritten Text Recognition* (HTR)) in einen maschinenlesbaren Text umgewandelt und auf diesem ein anwendungsspezifisches, textuelles Modell aus dem NLP-Bereich angewendet. Dieser Lösungsansatz wird im Folgenden als HTR-basiertes Verfahren bezeichnet und im Abschnitt 4.1 ausführlich vorgestellt. Ein fundamentaler Nachteil dieses Ansatzes ist, dass sich Fehler aus dem Texterkennungsprozess aufgrund des zweistufigen Verfahrens negativ auf die Leistung der NLP-Modelle auswirken können. Außerdem erfordern die zwei separaten Modelle einen hohen Ressourcen- und Zeitaufwand sowohl beim Training als auch bei der Inferenz [90]. Um diese Probleme zu beheben, wird im Abschnitt 4.2 ein HTR-freies Modell zur semantischen Analyse von handschriftlichen Dokumentenbildern vorgestellt, das auf einer Ende-zu-Ende-Architektur basiert und eine explizite Texterkennung vermeidet.

4.1 HTR-BASIERTES VERFAHREN

Eine sequentielle Kombination aus einem HTR- und einem textuellen NLP-Modell stellt einen intuitiven Ansatz zur semantischen Analyse von handschriftlichen Dokumentenbildern dar. Bei diesem Ansatz wird das Dokumentenbild zunächst mit einem HTR-Modell in maschinenlesbaren Text umgewandelt. Der erzeugte maschinenlesbare Text dient anschließend als Eingabe für ein anwendungsspezifisches NLP-Modell und ermöglicht so die Realisierung der semantischen Aufgabe. In diesem zweistufigen Verfahren erfolgt das Training des HTR- und des NLP-Modells unabhängig voneinander. Zur Realisierung des Ansatzes wird in dieser Arbeit ein wortsegmentiertes HTR-Modell mit state-of-the-art Modellen aus dem NLP-Bereich kombiniert. Eine Übersicht des vorgestellten Ansatzes ist in Abbildung 4.1 visualisiert. Aufgrund der wortsegmentierten Eingabe des HTR-Modells werden die Wortbilder in der Reihenfolge ihres Auftretens im Dokument separat verarbeitet. Dabei ist zu beachten, dass die Wahl eines wortsegmentierten Modells aufgrund der Vergleichbarkeit mit dem HTR-freien Ansatz in dieser Arbeit gewählt wurde und der Texterkenner prinzipiell durch ein beliebiges HTR-Modell unabhängig von der Segmentierung ersetzt werden kann. Die Verwendung einer alternativen Segmentierungsstufe führt jedoch zu Komplikationen bei der Übertragung der Ergebnisse auf die Dokumentenbildebene. Das HTR-Modell aus dieser Arbeit wurde von Kang et al. in [87] veröffentlicht und wird im Abschnitt 4.1.1 ausführlich beschrieben. Die in dieser Arbeit verwendeten Modelle für die semantische Schlüsselwortsuche, die NER und das QA werden im Abschnitt 4.1.2 vorgestellt.

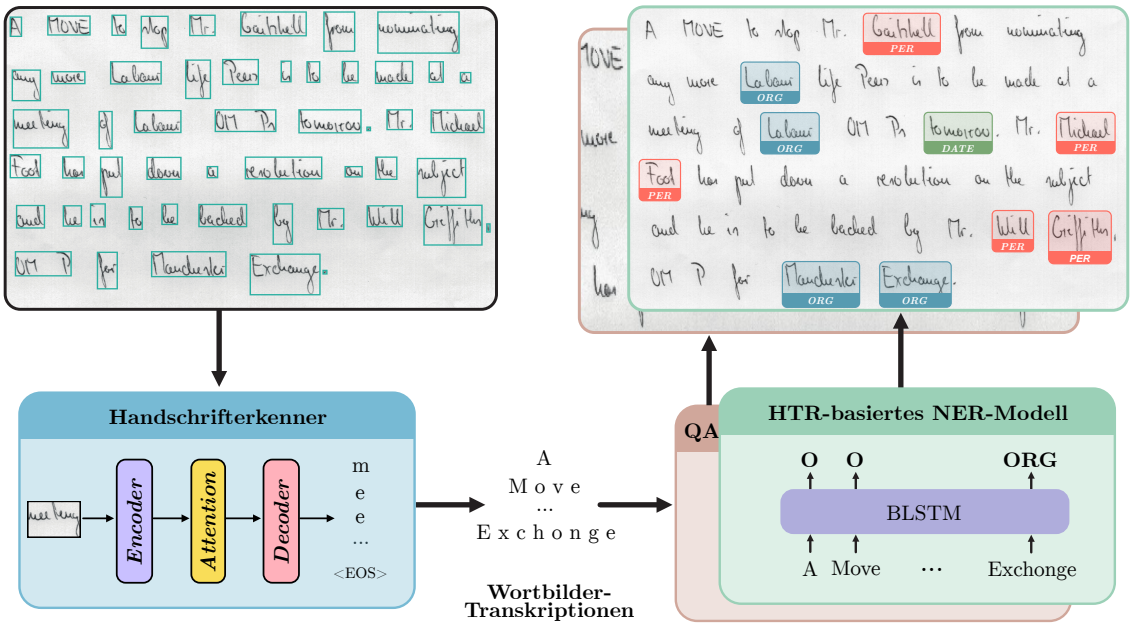


Abbildung 4.1: Ein Überblick über den vorgestellten HTR-basierten Ansatz zur semantischen Analyse von handschriftlichen Dokumentenbildern. Die vorsegmentierten Wortbilder des vorliegenden Dokumentenbildes werden zunächst mit einem HTR-Modell separat transkribiert. Der generierte Text dient anschließend als Eingabe für ein anwendungsspezifisches semantisches Modell.

4.1.1 *Handschrifterkennung*

Die Umwandlung von handschriftlichen Dokumentenbildern in maschinenlesbaren Text wird als Handschrifterkennung bezeichnet. Aufgrund der hohen Variabilität von Handschriften ist deren Transkription ein schwieriges und weiterhin relevantes Forschungsproblem [87, 114]. Die Leistungsfähigkeit von Handschrifterkennern konnte in den letzten Jahren aufgrund der Fortschritte im Bereich des maschinellen Lernens signifikant gesteigert werden [25, 114]. Die Ansätze unterscheiden sich bzgl. des erwarteten Eingabeformates der Dokumente. Im Allgemeinen wird zwischen segmentierungsfreien [184] und segmentierungsbasierten [25, 87, 114] Handschrifterkennern unterschieden. Bei den segmentierungsbasierten Ansätzen wird zusätzlich zwischen wort- [87] und zeilenbasierten [25, 114] Modellen differenziert.

Für einen fairen Vergleich des HTR-freien und des HTR-basierten Ansatzes ist ein wortsegmentiertes Texterkennungsmodell erforderlich. Das von Kang et al. in [87] publizierte Modell bietet vielversprechende Voraussetzungen und benötigt im Gegensatz zu den meisten Ansätzen aus der Literatur weder eine aufwendige Vorverarbeitung der Eingabedaten noch ein Sprachmodell. Das Ende-zu-Ende-Sequenzmodell ist in Abbildung 4.2 visualisiert und basiert auf dem Encoder-Decoder-Paradigma. Dabei extrahiert der Encoder Merkmale aus dem Wortbild und ein Attention-basierter Decoder erzeugt aus diesen Merkmalen iterativ die Transkription des Wortbildes.

Encoder

Der Encoder extrahiert zunächst visuelle Eigenschaften $\boldsymbol{X} \in \mathbb{R}^{(n \times h \times d)}$ des handschriftlichen Wortbildes $\boldsymbol{I}$ mit einem auf dem ImageNet [41] vortrainierten VGG-Netzwerk [183].

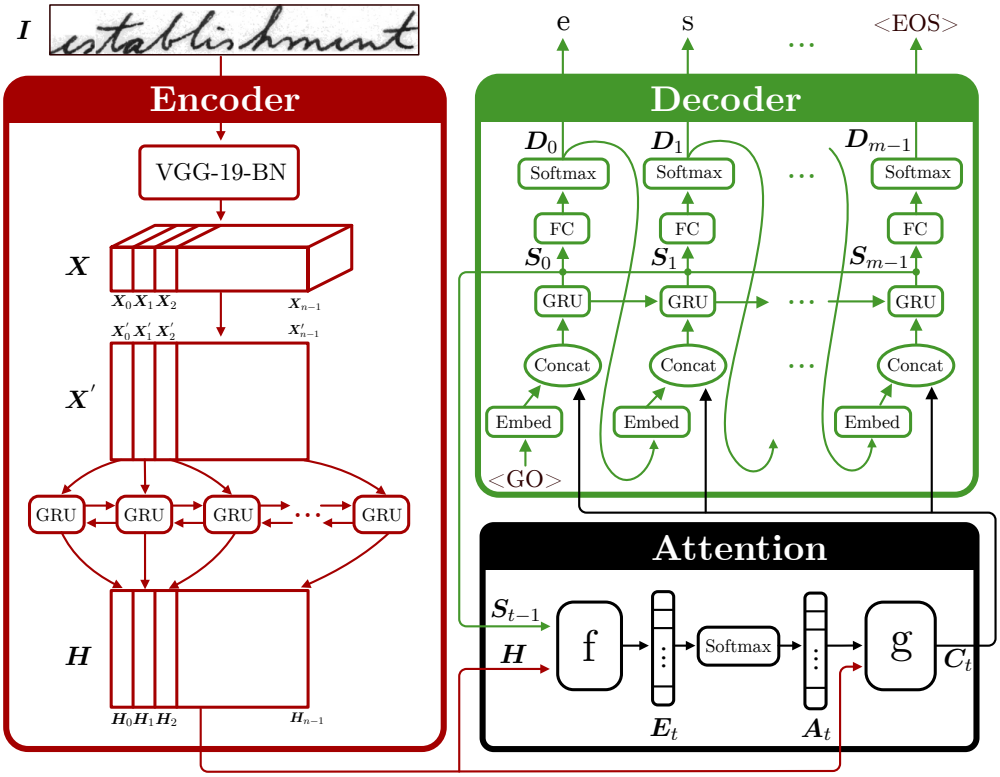


Abbildung 4.2: Die Architektur des Attention-basierten Encoder-Decoder-Modells zur Transkription von handschriftlichen Wortbildern. Die Grafik ist angelehnt an [87].

Dies erzeugt d Merkmalskarten mit einer Breite von n und einer Höhe von h. Motiviert durch die sequentielle Natur von handgeschriebenem Text, werden kontextabhängige Repräsentation für die visuellen Eigenschaften mit einem mehrschichtigen bidirektionalen GRU-Modell kodiert. Für die sequentielle Verarbeitung von $\boldsymbol{X}$ mit einem GRU müssen die Merkmalskarten jedoch zunächst in eine zweidimensionale Repräsentation $\boldsymbol{X}' \in \mathbb{R}^{(n \times c)}$, mit $c = h \cdot d$, überführt werden. Die resultierende Matrix $\boldsymbol{X}'$ kann als eine Sequenz von Merkmalsvektoren $(\boldsymbol{X}'_0, \boldsymbol{X}'_1, \ldots, \boldsymbol{X}'_{n-1})$ interpretiert werden, wobei $\boldsymbol{X}'_i$ mit einem Ausschnitt des Wortbildes $\boldsymbol{I}$ korrespondiert. Die endgültige Kodierung des Wortbildes wird durch die verborgenen Zustände $\boldsymbol{H} \in \mathbb{R}^{(n \times k)}$ des bidirektionalen GRUs erzeugt und kodiert neben den visuellen Informationen auch Kontextinformationen. Hierbei ist k die Dimensionalität der verborgenen Schichten im GRU-Modell.

Attention

Zur Fokussierung des Decoders auf bestimmte Bereiche des Wortbildes wird ein Attention-Mechanismus verwendet. Bei diesem Verfahren wird in jedem Dekodierungsschritt t ein Kontextvektor $\boldsymbol{C}_t \in \mathbb{R}^k$ ermittelt. Dieser Vektor enthält eine kontextsensitive Darstellung der Bildbereiche, welche für die Dekodierung des Zeichens zum Zeitpunkt t von Interesse sind. Die Berechnung des Kontextvektors basiert auf den sogenannten Attention-Gewichten $\boldsymbol{A}_t \in \mathbb{R}^n$, welche die Relevanz jeder Ausgabe des Encoders bei der Erzeugung der Transkription zum Zeitpunkt t quantifizieren. Für die Berechnung dieses Vektors wird die *Bahdanau-Attention* [11] verwendet, welche auf dem im vorhergehenden Zeitschritt

dekodierten Zeichen $\boldsymbol{S}_{t-1} \in \mathbb{R}^k$, sowie der Ausgabe des Encoders $\boldsymbol{H}$ basiert. Zunächst werden sogenannte Alignment-Werte $\boldsymbol{E}_t \in \mathbb{R}^n$ mit

$$\begin{aligned} \boldsymbol{E}_{t,i} &= \mathrm{f}(\boldsymbol{H}_i, \boldsymbol{S}_{t-1}) \\ &= \boldsymbol{w}^\intercal \cdot \underbrace{\tanh\left(\boldsymbol{W} \cdot \boldsymbol{H}_i + \boldsymbol{M} \cdot \boldsymbol{S}_{t-1}\right)}_{=\boldsymbol{Z}_i^{(t)}} \end{aligned} \tag{4.1}$$

berechnet. Diese Werte geben für jedes kodierte Bildsegment des Encoders einen skalaren Wert an, der die relative Relevanz dieses Segments für die Generierung des nächsten Zeichens darstellt. Formal werden die i-te Ausgabe des Encoders $\boldsymbol{H}_i$ und die verborgene Schicht des Decoders $\boldsymbol{S}_{t-1}$ mit den trainierbaren linearen Schichten $\boldsymbol{W}$ und $\boldsymbol{M} \in \mathbb{R}^{(k \times k)}$ auf neue Repräsentationen abgebildet. Eine intermediäre Darstellung $\boldsymbol{Z}_i^{(t)} \in \mathbb{R}^k$ wird durch eine elementweise Addition der neuen Repräsentationen und anschließender Anwendung der tanh-Aktivierungsfunktion erreicht. Schließlich wird der Alignment-Wert $\boldsymbol{E}_{t,i}$ durch das Skalarprodukt zwischen $\boldsymbol{Z}_i^{(t)}$ und einem trainierbaren Gewichtsvektor $\boldsymbol{w} \in \mathbb{R}^k$ bestimmt. Zur besseren Interpretation und Skalierung der Alignment-Werte werden diese mit der Softmax-Funktion in eine Pseudowahrscheinlichkeitsverteilung umgewandelt. Formal werden die Attention-Gewichte $\boldsymbol{A}_t$ für die Dekodierung zum Zeitpunkt t mit

$$\boldsymbol{A}_t = \mathrm{softmax}(\boldsymbol{E}_t) \tag{4.2}$$

berechnet. Der Kontextvektor $\boldsymbol{C}_t$ wird durch eine elementweise Multiplikation der Attention-Gewichte mit den Ausgaben des Encoders erzeugt:

$$\begin{aligned} \boldsymbol{C}_t &= \mathrm{g}(\boldsymbol{A}_t, \boldsymbol{H}) \\ &= \sum_{i=0}^{n-1} \boldsymbol{H}_i \cdot \boldsymbol{A}_{t,i} \end{aligned} \tag{4.3}$$

Decoder

Das Ziel der Dekodierung ist die Erzeugung einer Sequenz von Indizes $\{\mathbf{y}^{(0)}, \ldots, \mathbf{y}^{(l-1)}\}$, welche das im handschriftlichen Wortbild $\boldsymbol{I}$ enthaltene Wort repräsentiert. Jeder Index korrespondiert dabei mit einem Zeichen aus einem festgelegten Vokabular $\mathbf{V}$. Neben den möglichen Zeichen enthält das Vokabular an den ersten Positionen die Sonderzeichen $\langle GO \rangle$, $\langle PAD \rangle$ und $\langle EOS \rangle$. Hierbei ist $\langle GO \rangle$ das Startsignal, $\langle PAD \rangle$ ein Zeichen zum Auffüllen der Sequenz bis zur maximalen Dekodierungslänge m und $\langle EOS \rangle$ das Endsignal, welches das Ende des Dekodierungsprozesses angibt. Für jedes Zeichen aus dem Vokabular existiert eine vektorielle Repräsentation $\boldsymbol{E}_i \in \mathbb{R}^k$, die über eine Art trainierbare Zuordnungstabelle $\boldsymbol{E} \in \mathbb{R}^{(|\mathbf{V}| \times k)}$ festgelegt wird. Für die Dekodierung des Zeichens an der Position t wird zunächst eine kontextsensitive Repräsentation $\boldsymbol{S}_t \in \mathbb{R}^k$:

$$\boldsymbol{S}_t = \begin{cases} \mathrm{gru}(\boldsymbol{E}_0 \parallel \boldsymbol{H}_{n-1}), & \text{für } t = 0 \\ \mathrm{gru}(\boldsymbol{E}_{\mathbf{y}^{(t-1)}} \parallel \boldsymbol{C}_t), & \text{für } 1 \leq t < m \end{cases} \tag{4.4}$$

mittels eines unidirektionalen GRU-Modells erzeugt. Dabei basiert die initiale Dekodierung auf der Repräsentation des $\langle GO \rangle$-Zeichens und der letzten Encoder-Ausgabe. Für alle nachfolgenden Zeitschritte wird die Ausgabe auf Basis der Repräsentation des im vorherigen Zeitschrittes vorhergesagten Indizes $\mathbf{y}^{(t-1)} \in \{0, \ldots, |\mathbf{V}| - 1\}$ und dem aktuellen

Kontextvektor $\boldsymbol{C}_t$ bestimmt. Die kontextsensitive Repräsentation zum Zeitpunkt t wird anschließend mit einer trainierbaren Gewichtsmatrix $\boldsymbol{U} \in \mathbb{R}^{(|\mathbf{V}| \times k)}$ auf die Vokabulargröße abgebildet. Auf das Ergebnis wird eine Softmax-Funktion angewendet, sodass $\boldsymbol{D}_t$ mit

$$\boldsymbol{D}_t = \text{softmax}(\boldsymbol{U} \cdot \boldsymbol{S}_t) \tag{4.5}$$

eine Pseudowahrscheinlichkeitsverteilung über das Vokabular darstellt. Für die finale Dekodierung des t-ten Ausgabesymbols wird der Index $\boldsymbol{y}_t$ des wahrscheinlichsten Symbols aus dem Vokabular mit

$$\boldsymbol{y}_t = \underset{i \in \{0,\ldots,|\mathbf{V}|-1\}}{\text{argmax}} \boldsymbol{D}_{t,i} \tag{4.6}$$

berechnet. Der Decoder produziert so lange eine Zeichenausgabe, bis entweder das $\langle EOS \rangle$-Symbol vorhergesagt wird oder die maximale Anzahl an Zeitschritten erreicht ist.

Das Modell wird in Bezug auf die Kullback-Leibler-Divergenz [99] optimiert. Um die Generalisierungsfähigkeit des Modells zu erhöhen, wird zudem ein sogenannter *Label-Smoothing*-Ansatz [196] verwendet. Dabei handelt es sich um eine Regularisierungstechnik, die ein Rauschen in die Repräsentationen der Gold-Standard Annotationen einführt und damit die Überkonfidenz in der Vorhersage der Netzwerke reduziert. Formal werden die Nullen in der 1-aus-$|\mathbf{V}|$-Kodierung durch $\frac{0.4}{|\mathbf{V}|}$ und die Einsen durch $1 - \frac{|\mathbf{V}|-1}{|\mathbf{V}|} \cdot 0.4$ ersetzt.

4.1.2 *HTR-basierte semantische Modelle*

Im Folgenden werden die Modelle für die semantische Schlüsselwortsuche, die NER und das QA vorgestellt. Ein großer Vorteil des HTR-basierten Ansatzes gegenüber dem HTR-freien ist die Verwendung von state-of-the-art NLP-Modellen ohne explizite Anpassung der Architekturen. Daher handelt es sich bei den in dieser Arbeit verwendeten Ansätzen um bewährte Verfahren aus dem NLP-Bereich.

SEMANTISCHE SCHLÜSSELWORTSUCHE

Das Ziel der semantischen Schlüsselwortsuche ist die Erzeugung einer Retrieval-Liste, in der die segmentierten Wortbilder aus einer vorgegebenen Sammlung von Dokumenten nach ihrer semantischen Ähnlichkeit zu einer benutzerdefinierten Anfrage geordnet sind. Für die Bestimmung der semantischen Ähnlichkeit zwischen einem Wortbild und einer gegebenen Anfrage werden die Wortbilder der Kollektion zunächst mit dem HTR-Modell in eine textuelle Repräsentation umgewandelt. Aufgrund der textuellen Repräsentation der Wortbilder kann die semantische Ähnlichkeit zwischen einer Anfrage und den Wortbildern der Dokumentensammlung auf der Basis eines textuell vortrainierten semantischen Worteinbettungsmodells aus dem NLP-Bereich (z.B. FastText [18]) definiert werden. Dabei wird die Ähnlichkeit zwischen einer gegebenen Anfrage und einem Wortbild durch die Kosinusähnlichkeit ihrer Repräsentationen $\boldsymbol{a} \in \mathbb{R}^n$ und $\boldsymbol{b} \in \mathbb{R}^n$ im semantischen Vektorraum mit

$$\text{dcos}(\boldsymbol{a}, \boldsymbol{b}) = \frac{\boldsymbol{a}^\intercal \cdot \boldsymbol{b}}{\|\boldsymbol{a}\| \cdot \|\boldsymbol{b}\|} \tag{4.7}$$

ermittelt. Die Ausgabe erfolgt in Form einer Ergebnisliste, in der alle Wortbilder aus der Datenbank in absteigender Reihenfolge nach ihrer Ähnlichkeit mit der Suchanfrage sortiert sind.

Exkurs 3: *Robustly optimized BERT approach* (RoBERTa)

Das RoBERTa-Modell ist eine optimierte Variante des *Bidirectional Encoder Representations from Transformers* (BERT)-Modells [42]. Dazu werden in [122] die Einflüsse wichtiger Hyperparameter des BERT-Modells evaluiert und geeignet gewählt. Eine Erhöhung der Trainingsiterationen mit mehr Daten und größeren Batches im Vortraining führt dabei auf allen getesteten NLP-Benchmarks zu besseren Ergebnissen. Zusätzlich wird die Leistung des Modells durch eine Erhöhung der Anzahl der Tokens im Vokabular und die Verwendung von *Byte-Pair-Encodings* [178] gesteigert. Im Vortraining erwies sich zudem die Verwendung der *Next Sentence Prediction* (NSP)-Aufgabe als nachteilig und eine dynamische Variante der *Masked Language Modelling* (MLM)-Aufgabe als am besten geeignet.

NAMED ENTITY RECOGNITION

Ansätze die auf Transformer-Architekturen basieren erzielen derzeit bei den meisten NER-Benchmarks die besten Resultate [243]. Im Vergleich zu LSTM-basierten Modellen führen sie jedoch nur zu marginalen Leistungsverbesserungen und haben dabei einen erhöhten Speicher- sowie Berechnungsaufwand [4, 243]. Aus Ressourcengründen und zur besseren Vergleichbarkeit basiert das in dieser Arbeit verwendete NER-Modell auf einer ähnlichen LSTM-Architektur wie die HTR-freien NER-Ansätze aus der Literatur [166, 206, 207]. Konkret besteht die vorgestellte NER-Architektur, wie in Abbildung 4.3 visualisiert, aus einer Kombination eines Modells zur Wortbildeinbettung und einem mehrschichtigen BLSTM. Der Ansatz überführt die Wortbilder des Dokuments $\mathbf{D} = \{\mathbf{d}^{(0)}, \ldots, \mathbf{d}^{(n-1)}\}$ zunächst in d-dimensionale Wortbildrepräsentationen $\boldsymbol{C} \in \mathbb{R}^{(n \times d)}$. Dazu werden die Wortbilder mit dem HTR-Modell sequentiell transkribiert und mit einem vortrainierten Worteinbettungsmodell, speziell dem *Robustly optimized BERT approach* (RoBERTa)-Modell [122], in eine Sequenz von kontextsensitiven Wortrepräsentationen umgewandelt. Weiterführende Informationen zum RoBERTa-Modell sind in Exkurs 3 aufgeführt. Ein zweischichtiges BLSTM lernt die Beziehungen zwischen den Repräsentationen und ermöglicht so die Extraktion kontextsensitiver Informationen zur Klassifikation der Wortbilder. Für jedes Wortbild $\mathbf{d}^{(k)}$ wird eine Pseudowahrscheinlichkeitsverteilung $\boldsymbol{Y}_k \in \mathbb{R}^{|\mathbf{K}|}$ über die möglichen Entitätsklassen $\mathbf{K}$ erzeugt. Dafür wird eine Kombination aus einer linearen vollvernetzten Schicht und einer Softmax-Funktion auf die Ausgaben des BLSTMs angewendet. Für das Wortbild $\mathbf{d}^{(k)}$ wird die Entität $\hat{\boldsymbol{y}}_k$ mit der höchsten Pseudowahrscheinlichkeit vorhergesagt:

$$\hat{\boldsymbol{y}}_k = \underset{i \in \{0,\ldots,|\mathbf{K}|-1\}}{\operatorname{argmax}} \boldsymbol{Y}_{k,i} \tag{4.8}$$

Das Modell wird vollüberwacht mit der Kreuzentropie auf die Vorhersage der NEs optimiert. Dabei werden die NE-Annotationen über eine 1-aus-$|\mathbf{K}|$-Kodierung repräsentiert. Jedes Wortbild kann genau einer Entitätsklasse zugeordnet werden, da für unbekannte Entitäten eine Art Rückweisungsklasse existiert.

QUESTION ANSWERING

Die Beantwortung von natürlichsprachlichen Fragen auf der Grundlage einer Sammlung von Dokumenten kann insbesondere bei großen Kollektionen zu einem hohen Rechen- und Ressourcenaufwand führen. Daher werden in den meisten QA-Ansätzen vor der Beantwortung der Frage zunächst effiziente Retrieval-Ansätze benötigt, welche die Dokumenten-

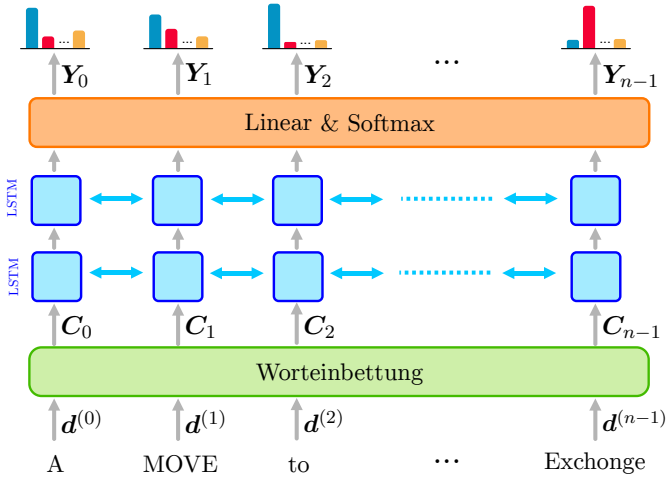


Abbildung 4.3: Ein Überblick über die Architektur des textbasierten NER-Systems. Hierbei dienen die HTR-Ergebnisse des zu analysierenden Dokumentenbildes als Eingabedaten für das Modell.

kollektion auf eine geringe Anzahl von fragerelevanten Dokumenten reduziert. Der HTR-basierte QA-Ansatz in dieser Arbeit verwendet das *Term Frequency-Inverse Document Frequency* (TF-IDF)-Verfahren [130], um die relevantesten Dokumente für die gegebene Frage aus der Sammlung zu ermitteln. Das TF-IDF-Verfahren ist ein häufig verwendeter Ansatz im IR-Bereich und berechnet für ein Wort $\mathbf{w}$ und ein Dokument $\mathbf{D}$ einen numerischen Wert, der die Wichtigkeit des Wortes für das Dokument in Abhängigkeit von der gesamten Dokumentenkollektion repräsentiert. Der Wert setzt sich aus der *Term Frequency* ($\mathrm{tf}(\mathbf{w}, \mathbf{D})$) und der *Inverse Document Frequency* ($\mathrm{idf}(\mathbf{w})$) zusammen:

$$\mathrm{tfidf}(\mathbf{w}, \mathbf{D}) = \mathrm{tf}(\mathbf{w}, \mathbf{D}) \cdot \mathrm{idf}(\mathbf{w}) \tag{4.9}$$

Dabei entspricht die *Term Frequency* der Anzahl der Vorkommen des Wortes $\mathbf{w}$ im Dokument $\mathbf{D}$. Die *Inverse Document Frequency* stellt eine Form des Informationsgehalts eines Wortes dar und wird für das Wort $\mathbf{w}$ mit

$$\mathrm{idf}(\mathbf{w}) = \log\left(\frac{m}{m_w}\right) \tag{4.10}$$

ermittelt. Hierbei ist m die Anzahl der Dokumente in der Kollektion und m_w die Anzahl der Dokumente, in denen das Wort $\mathbf{w}$ vorkommt. Abschließend wird die Relevanz des Dokuments $\mathbf{D}$ für die Frage $\mathbf{Q}$ mit der Summe der TF-IDF-Werte für die in der Frage vorkommenden Wörter bestimmt:

$$\mathrm{rel}(\mathbf{Q}, \mathbf{D}) = \sum_{\mathbf{w} \in \mathbf{Q}} \mathrm{tfidf}(\mathbf{w}, \mathbf{D}) \tag{4.11}$$

Für die k relevantesten Dokumente werden anschließend separat die Antworten bezüglich der Frage ermittelt. Der Aufbau der in dieser Arbeit verwendeten QA-Architektur ist in Abbildung 4.4 dargestellt und besteht aus der Kombination eines Transformer-Encoder-Modells und einer linearen Schicht zur Lokalisierung der Antwort im Dokument. Das Encoder-Modell basiert auf einem vortrainierten BERT-Modell[1]. Hierbei wurden die 24-Encoder-Schichten der Transformer-Architektur auf die Sprachmodellierung vortrainiert

1 https://huggingface.co/bert-large-cased-whole-word-masking-finetuned-squad

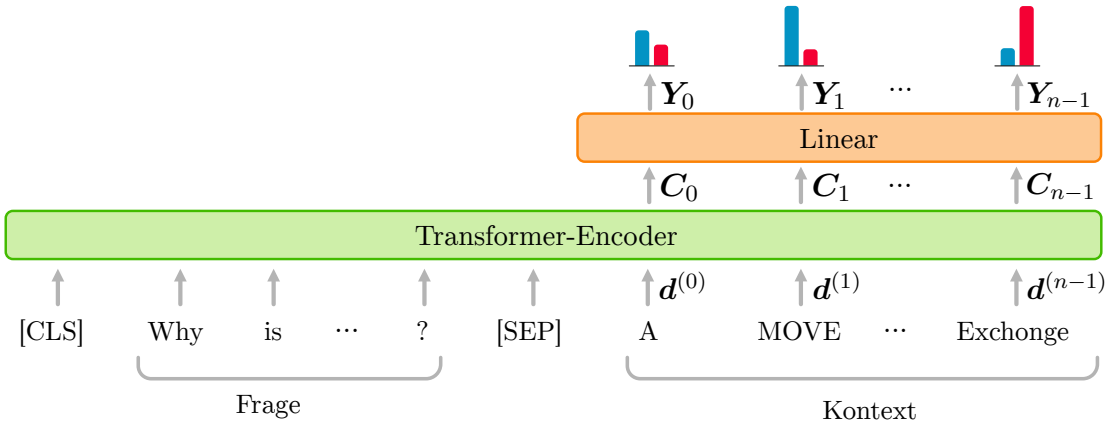


Abbildung 4.4: Die Architektur des textuellen QA-Modells. Der Kontext besteht aus den HTR-Transkriptionen der im Dokumentenbild enthaltenen Wortbilder.

und mit dem *Stanford Question Answering Dataset* (SQuAD) [156] an die QA-Aufgabe angepasst. Die transkribierten Wortbilder des Dokuments $\mathbf{D} = \{\mathbf{d}^{(0)}, \ldots, \mathbf{d}^{(n-1)}\}$ werden zunächst mit dem Encoder-Modell in d-dimensionale Wortrepräsentationen $\boldsymbol{C} \in \mathbb{R}^{(n \times d)}$ umgewandelt. Anschließend wird auf jede dieser n Repräsentationen eine lineare Schicht angewendet. Die Ausgabe ist eine Repräsentation $\boldsymbol{Y} \in \mathbb{R}^{(n \times 2)}$, die für jedes Wortbild einen Start- und Endwert kodiert. Die Antwort wird auf Basis dieser Werte aus dem vorgegebenen Kontext extrahiert. Sie beginnt beim Wort mit dem höchsten Startwert und endet beim Wort mit dem höchsten Endwert im Dokument. Formal wird der Startwert s und der Endwert e mit

$$s = \underset{i \in \{0,\ldots,n-1\}}{\operatorname{argmax}} \boldsymbol{Y}_{i,0} \tag{4.12}$$

$$e = \underset{i \in \{0,\ldots,n-1\}}{\operatorname{argmax}} \boldsymbol{Y}_{i,1} \tag{4.13}$$

berechnet. Zusätzlich zur extrahierten Antwort wird eine Konfidenz benötigt. Hierzu wird zunächst die Softmax-Funktion auf die Ausgaben des QA-Modells mit

$$\boldsymbol{s} = \operatorname{softmax}(\boldsymbol{Y}_{:,0}) \tag{4.14}$$

$$\boldsymbol{e} = \operatorname{softmax}(\boldsymbol{Y}_{:,1}) \tag{4.15}$$

angewendet. Anschließend wird die Konfidenz z durch die Multiplikation der Pseudowahrscheinlichkeiten für die vom Modell vorhergesagten Start- und Endindizes mit

$$z = \boldsymbol{s}_s \cdot \boldsymbol{e}_e \tag{4.16}$$

berechnet. Hierbei repräsentiert $\boldsymbol{s}_s$ die Wahrscheinlichkeit des Startindex und $\boldsymbol{e}_e$ die des Endindex. Die finale Antwort des Systems ist der Textbereich mit der höchsten Konfidenz aus den k relevantesten Dokumenten.

4.2 HTR-FREIES VERFAHREN

Im Folgenden wird ein HTR-freier Ansatz zur semantischen Analyse von handschriftlichen Dokumentenbildern vorgestellt. Durch die Verwendung einer Ende-zu-Ende-Architektur

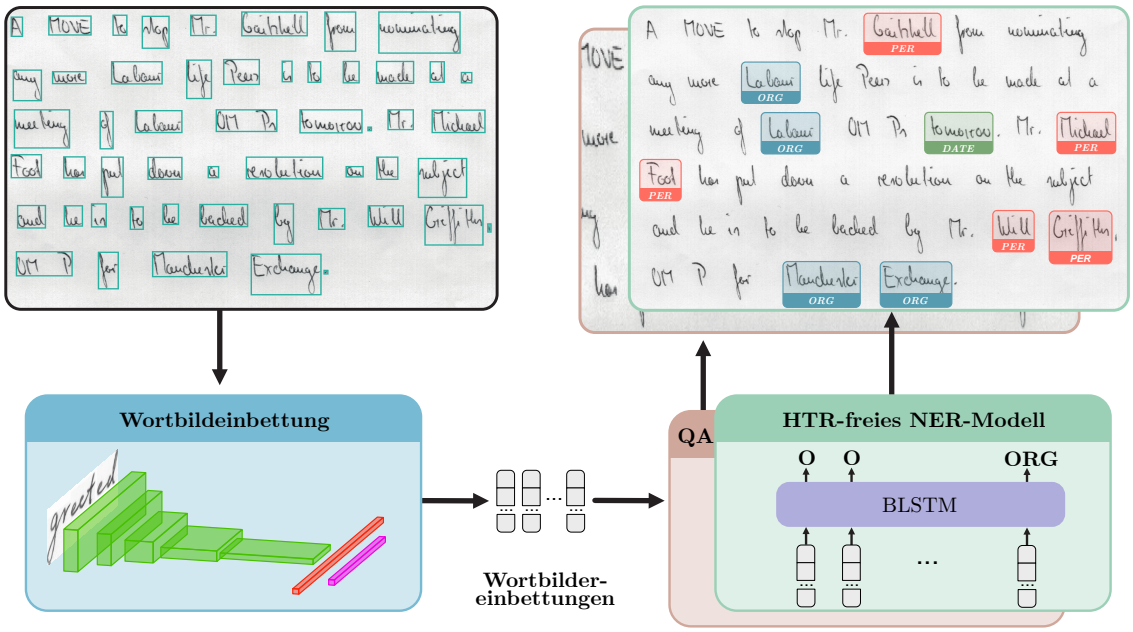


Abbildung 4.5: Ein Überblick über den vorgestellten HTR-freien Ansatz zur semantischen Analyse von handschriftlichen Dokumentenbildern. Die vorsegmentierten Wortbilder des vorliegenden Dokumentenbildes werden mit einem Faltungsnetzwerk separat in vektorielle Repräsentationen überführt. Die Wortbildrepräsentationen dienen anschließend als Eingabe für ein anwendungsspezifisches semantisches Modell.

und der Vermeidung einer expliziten Texterkennung wird das Problem der Fehlerfortpflanzung aus dem HTR-basierten Ansatz gelöst. Der HTR-freie Ansatz ist in Abbildung 4.5 visualisiert und umfasst die Bereiche der Wortbildeinbettung und der anwendungsspezifischen Architekturen zur HTR-freien semantischen Dokumentenbildanalyse. Das Modell erwartet das zu analysierende Dokument als eine Liste von vorsegmentierten Wortbildern. Die Wortbilder werden zunächst in der Reihenfolge ihres Auftretens im Dokument mit einem CNN in vektorielle Darstellungen überführt. Die Wortbildrepräsentationen dienen anschließend als Eingabe für ein anwendungsspezifisches semantisches Modell. Nachfolgend werden die einzelnen Bestandteile dieses Ansatzes vorgestellt. Insbesondere werden die Architektur der Wortbildeinbettung im Abschnitt 4.2.1 und die auf vektorieller Eingabe basierten Modelle zur semantischen Schlüsselwortsuche, zur NER und zum QA detailliert im Abschnitt 4.2.2 präsentiert.

4.2.1 *Wortbildeinbettung*

Das Ziel der Wortbildeinbettung ist die Umwandlung handschriftlicher Wortbilder in vektorielle Darstellungen mit vorgegebener Dimensionalität $d \in \mathbb{N}_{>0}$. Zur Realisierung dieser Transformation haben sich neuronale Architekturen etabliert, die aus einem Faltungs- und einem Klassifikationsteil bestehen [94, 98, 190, 191, 233]. Das gegebene Wortbild wird zunächst durch die Faltungsschichten des Netzwerks und einer anschließenden Pooling-Operation in einen kompakten Merkmalsvektor mit fester Dimensionalität umgewandelt. Der Merkmalsvektor dient als Eingabe für den Klassifikationsteil, der mit einem MLP die gewünschte Dimensionalität d erzeugt. Die Anzahl der Neuronen in der Ausgabeschicht des Netzwerks legt dabei die Dimensionalität fest.

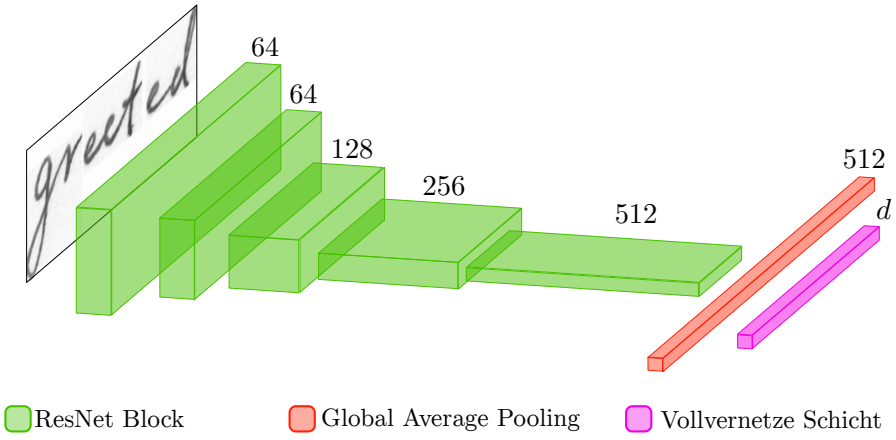


Abbildung 4.6: Die Architektur zur Transformation eines handschriftlichen Wortbildes in eine d-dimensionale Vektorrepräsentation. Die Faltungsschichten des Modells sind analog zum ResNet34 gewählt. Die erzeugten Merkmalskarten werden mit dem GAP in einen 512-dimensionalen Vektor überführt und anschließend mit einem MLP auf eine benutzerdefinierte Ausgabedimensionalität d abgebildet. Die Zahlen über den Blöcken repräsentieren die Anzahl der Filter bei den Faltungen bzw. die Anzahl der Neuronen in den vollvernetzten Schichten.

In dieser Arbeit werden zwei Methoden zur Wortbildeinbettung vorgestellt und evaluiert. Der erste Ansatz ist die Standardvorgehensweise bei HTR-freien Modellen aus der Literatur [1, 167, 206]. Hierbei werden die Parameter des Netzwerks zufällig initialisiert und erst während des Trainings der NLP-Aufgabe angepasst. Der zweite Ansatz nutzt das Vorhandensein von Textinformationen in Wortbildern und trainiert das Netzwerk zur Wortbildeinbettung zunächst auf die Vorhersage von textuellen Wortrepräsentationen, bevor es auf die NLP-Aufgabe angepasst wird. Dabei haben sich Ansätze etabliert, die Bilder und Texte in einen gemeinsamen Vektorraum projizieren [94, 190]. Am weitesten verbreitet ist die Verwendung eines textuellen Worteinbettungsraums (z.B. PHOC) und die Abbildung eines Wortbildes in diesen Raum mit einem CNN [98, 191, 233]. Insbesondere ResNets haben sich für die Vorhersage von Worteinbettungen auf der Basis handschriftlicher Wortbilder etabliert und liefern state-of-the-art Ergebnisse auf den meisten WS-Benchmarks [98, 190]. Entgegen dem aktuellen Trend bei Klassifikationsmodellen, in denen der Klassifikationsteil nur aus einer vollvernetzten Schicht besteht, wird bei der Vorhersage von Wortbildrepräsentationen ein vergleichsweise großes Klassifikationsnetzwerk verwendet [94, 190, 191]. Als Beispiel besitzt das PHOCResNet [190] über 100 Millionen anpassbare Parameter, wobei die Faltungsschichten für 23.7% und die Klassifikationsschichten für 76.3% der Gewichte verantwortlich sind. Die Kombination aus einer hohen Anzahl von Parametern und einer geringen Anzahl von Trainingsdaten bei Datensätzen mit handschriftlichen Dokumentenbildern kann zu einer Überanpassung an die Trainingsdaten führen und insbesondere beim Ende-zu-Ende-Training problematisch werden. In dieser Arbeit wird daher eine optimierte Netzwerkkonfiguration für die Vorhersage von semantischen Worteinbettungen vorgestellt. Eine Reduktion der Parameter hat zudem den Vorteil, dass die Modelle schneller und mit weniger Ressourcen trainiert werden können.

Die Architektur des in dieser Arbeit verwendeten Ende-zu-Ende-Modells zur Wortbildeinbettung ist in Abbildung 4.6 visualisiert. Hierbei werden zunächst Merkmale des Eingabebildes mit den Faltungsschichten des ResNet34 extrahiert und diese mit dem GAP

in einen 512-dimensionalen Vektor transformiert. Das GAP berechnet den arithmetischen Mittelwert für jede der Merkmalskarten aus der letzten Schicht des Faltungsnetzwerks und bildet daraus einen Vektor, dessen Dimensionalität der Anzahl der Merkmalskarten entspricht. Auf die Ausgabe der Pooling-Operation wird eine ReLU-Aktivierungsfunktion angewendet. Anschließend dient dieser Merkmalsvektor als Eingabe für eine vollvernetze Schicht, welche eine Repräsentation mit der vorgegebenen Dimensionalität d erzeugt. Aufgrund des Dynamikbereichs ($\mathbb{R}$) von vortrainierten semantischen Worteinbettungen aus dem NLP-Bereich wird keine Aktivierungsfunktion auf die Ausgabe des Netzwerks angewendet.

4.2.2 *HTR-freie semantische Modelle*

Für die semantische Analyse von Dokumentenbildern ohne eine explizite Texterkennung werden geeignete Architekturen benötigt, die auf vektorieller statt textueller Eingabe basieren. Die in dieser Arbeit vorgestellten Ansätze bauen dabei in der Regel auf Architekturen aus dem NLP-Bereich auf und werden speziell an die Aufgabenstellung angepasst. Für die sequentielle Verarbeitung von Wortbildern werden etablierte BLSTM-Architekturen verwendet, auch wenn state-of-the-art Modelle aus dem NLP-Bereich auf Transformern basieren. Ein Grund für diese Wahl ist der hohe Bedarf an Trainingsdaten bei Transformern. Zudem ist der wesentliche Nachteil von BLSTMs gegenüber Transformern der Informationsverlust bei der Verarbeitung langer Sequenzen. Dies ist jedoch bei Dokumentenbildern aufgrund der begrenzten Anzahl von Wörtern pro Dokumentenbild wenig relevant. Im Folgenden werden die in dieser Arbeit verwendeten semantischen Modelle im Detail vorgestellt.

4.2.2.1 *Semantische Schlüsselwortsuche*

Der HTR-freie Ansatz zur semantischen Schlüsselwortsuche basiert im Allgemeinen auf dem bewährten Verfahren der gemeinsamen Unterraumrepräsentation (engl.: *Common Subspace Representation*) [8]. Die Vorgehensweise dieses Verfahrens ist in Abbildung 4.7 dargestellt. Hierbei werden maschinenlesbare Wörter und handschriftliche Wortbilder in einen gemeinsamen Vektorraum überführt, wobei die Distanz im Vektorraum ihrer semantischen Ähnlichkeit entspricht. In dieser Arbeit wird der Abstand mit der Kosinusähnlichkeit berechnet (siehe Formel 4.7). Die Ausgabe der HTR-freien Schlüsselwortsuche ist eine Retrieval-Liste, bei der alle Wortbilder aus einer gegeben Kollektion absteigend nach ihrer semantischen Ähnlichkeit zur Anfrage sortiert sind. Die Anfrage kann aufgrund der Einbettung von Texten und Bildern in denselben Vektorraum sowohl durch ein maschinenlesbares Wort (QbS) als auch durch ein Bild (QbE) repräsentiert werden.

Der gemeinsame Vektorraum wird durch ein vortrainiertes semantisches Worteinbettungsmodell aus dem NLP-Bereich gebildet. Folglich ist die Abbildung maschinenlesbarer Wörter in den gemeinsamen Vektorraum weitgehend trivial. Dagegen stellt die Transformation handschriftlicher Wortbilder in vordefinierte semantische Repräsentationen eine anspruchsvolle Aufgabe dar. Aufbauend auf den Forschungsergebnissen der syntaktischen Schlüsselwortsuche wird für die Abbildung von Wortbildern in dieser Arbeit ein neuronales Faltungsnetzwerk eingesetzt. Dieses Netzwerk wird darauf trainiert, für ein gegebenes Wortbild die semantische Repräsentation des im Wortbild enthaltenen Textes vorherzusagen. Dabei wird die Repräsentation durch das textuelle Worteinbettungsmodell vorgegeben. Konkret wird das im Abschnitt 4.2.1 beschriebene neuronale Netzwerk für

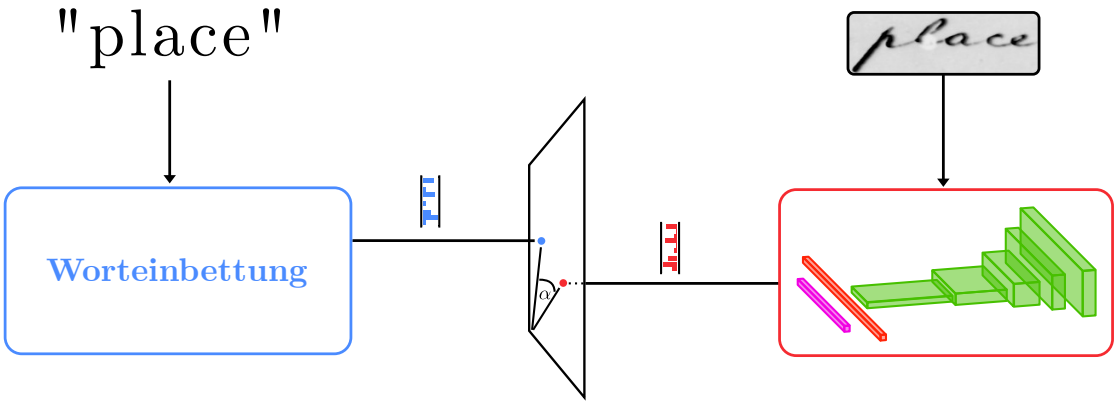


Abbildung 4.7: Ein Überblick über den vorgestellten Ansatz zur Realisierung einer semantischen Schlüsselwortsuche. Es wird eine Abbildung von vorsegmentierten Wortbildern mit einem Faltungsnetzwerk in einen textuellen semantischen Vektorraum gelernt. Die Parameter des vortrainierten Worteinbettungsmodells werden dabei nicht angepasst. Die Distanz (α) zwischen zwei Elementen im Vektorraum entspricht ihrer semantischen Ähnlichkeit. Die Grafik ist angelehnt an [193].

die Abbildung verwendet und die Ausgabedimensionalität entsprechend dem vortrainierten semantischen Worteinbettungsraum angepasst. Für das Training steht eine annotierte Stichprobe zur Verfügung, die aus vorsegmentierten Wortbildern und deren Transkriptionen besteht. Das textuelle Worteinbettungsmodell wird während des Trainings nicht optimiert.

4.2.2.2 *Named Entity Recognition*

Die HTR-freie NER-Architektur für handschriftliche Dokumentenbilder basiert auf der Kombination einer Wortbildeinbettung und eines Klassifikationsnetzwerks. Dabei folgt die Architektur und das Training im Allgemeinen dem in dieser Arbeit vorgestellten HTR-basierten NER-Ansatz und wird an die Verarbeitung vektorieller statt textueller Eingaben angepasst. Der Hauptunterschied zwischen den beiden Ansätzen besteht in der Einbettung der Wortbilder. Hierbei wird jedes Wortbild $\mathbf{d}^{(i)}$ des Dokuments $\mathbf{D} = \{\mathbf{d}^{(0)}, \ldots, \mathbf{d}^{(n-1)}\}$ mit dem im Abschnitt 4.2.1 vorgestellten Wortbildeinbettungsmodell in eine vektorielle Repräsentation $\boldsymbol{C}_i \in \mathbb{R}^d$ überführt. Analog zum HTR-basierten NER-Ansatz werden diese Repräsentationen anschließend mit einem zweistufigen BLSTM in kontextsensitive Wortbildrepräsentationen transformiert. Auf die Ausgabe des BLSTMs wird eine vollvernetzte Schicht mit anschließender Softmax-Funktion angewendet. Dies erzeugt eine Pseudowahrscheinlichkeitsverteilung über die möglichen Entitätsklassen $\mathbf{K}$ für jedes Wortbild des Dokuments. Die endgültige Klassifikation eines Wortbildes erfolgt analog zum HTR-basierten Modell, indem die Entität mit der höchsten Pseudowahrscheinlichkeit ausgewählt wird.

Um die Generalisierungsfähigkeit des Modells zu verbessern, wird während des Trainings ein Label-Smoothing-Ansatz angewendet. Formal wird eine geglättete Verteilung mit dem Ansatz aus [206] berechnet. Hierbei werden alle Werte aus der 1-aus-$|\mathbf{K}|$-Kodierung durch zufällig gezogenen Werte $\boldsymbol{y}' \in \mathbb{R}^{|\mathbf{K}|}$ aus einer Normalverteilung mit einem Mittelwert von $\mu = \frac{0.25}{|\mathbf{K}|}$ und einer Standardabweichung von $\sigma = \frac{\mu}{5}$ ersetzt. Die ursprüngliche Eins der Kodierung wird durch $1 - \sum_{i=0}^{|\mathbf{K}|-1} \boldsymbol{y}'_i$ ersetzt, sodass die Summe aller Elemente der geglätteten Verteilung wieder 1 entspricht.

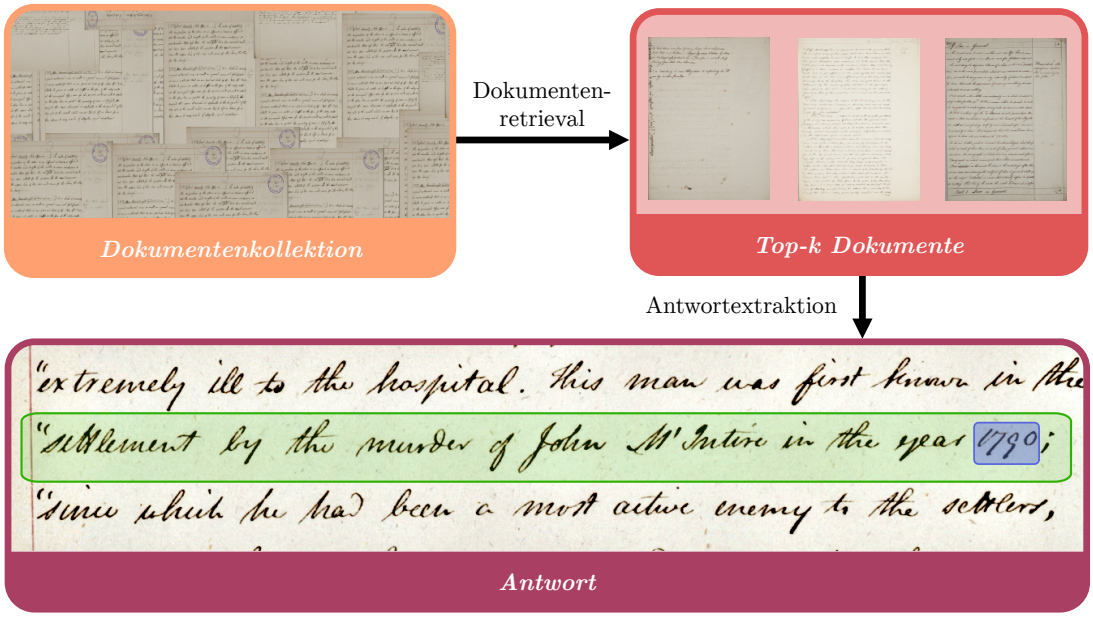


Abbildung 4.8: Ein Überblick über die Pipeline zur Beantwortung von Fragen auf Basis einer Kollektionen von Dokumentenbildern. Für die textuelle Frage „In welchem Jahr wurde John McIntire ermordet?“ identifiziert ein Retrieval-Modell zunächst die k relevantesten Dokumentenbilder aus einer gegebene Sammlung, die zur Beantwortung der Frage dienen. Anschließend wird ein Bildbereich auf Zeilenebene (grün) bzw. Wortebene (blau) aus einem dieser k Dokumentenbilder als Antwort zurückgegeben.

4.2.2.3 *Question Answering*

Die in dieser Arbeit verwendete Pipeline zur HTR-freien Beantwortung von Fragen auf Basis einer Kollektionen von handschriftlichen Dokumentenbildern ist in Abbildung 4.8 dargestellt und umfasst die beiden Komponenten Dokumentenretrieval und Antwortextraktion. Im Gegensatz zu QA-Modellen aus dem NLP-Bereich erzeugt der vorgestellte Ansatz keine Antwort in maschinenlesbarer Form, sondern liefert einen Ausschnitt des Dokumentenbildes, der die Antwort enthält. Dazu wird analog zu [133] ein Retrieval-basierter QA-Ansatz verwendet, der einen Bildbereich auf Zeilenebene aus einem Dokumentenbild extrahiert und diesen als Antwort zurückgibt.

Dokumentenretrieval

Um die relevantesten Dokumente zu einer Anfrage in einer gegebenen Dokumentenkollektion zu bestimmen, werden die Ähnlichkeiten zwischen jedem Wortbild eines Dokuments $\mathbf{D}$ und jedem Fragewort aus der vorverarbeiteten Anfrage $\mathbf{Q}$ berechnet. Für die Bestimmung der Ähnlichkeiten werden die textuellen Fragewörter und die Wortbilder in einen gemeinsamen Vektorraum eingebettet. Das Vorgehen ist analog zur semantischen Schlüsselwortsuche und überführt Wortbilder mit einem CNN in einen textuell vortrainierten Vektorraum. Für alle vektoriellen Fragewörter $\boldsymbol{q} \in \mathbf{Q}$ wird die größte Übereinstimmung in Bezug auf die Repräsentationen der Wortbilder $\boldsymbol{w} \in \mathbf{D}$ mit der Kosinusähnlichkeit berechnet. Die Grundidee dieser Berechnung ist die Überprüfung des Vorhandenseins oder

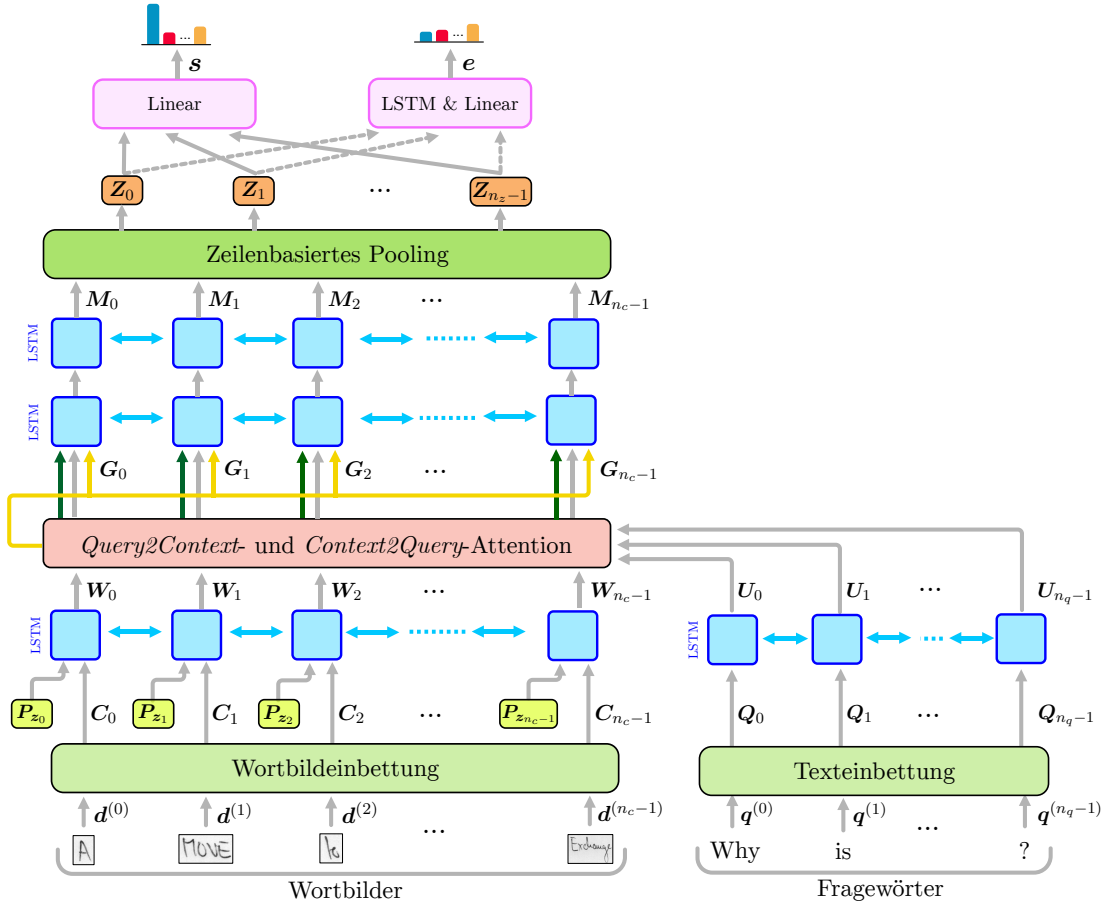


Abbildung 4.9: Die angepasste *Bidirectional Attention Flow* (BIDAF)-Architektur für HTR-freies QA auf Zeilenebene. Die Grafik ist inspiriert von [180].

Fehlens eines bestimmten Fragewortes im Dokument. Die Gesamtähnlichkeit zwischen der Anfrage $\mathbf{Q}$ und einem Dokument $\mathbf{D}$ wird mit

$$\text{docscore}(\mathbf{D}, \mathbf{Q}) = \frac{1}{|\mathbf{Q}|} \cdot \sum_{\boldsymbol{q} \in \mathbf{Q}} \max_{\boldsymbol{w} \in \mathbf{D}} (\text{dcos}(\boldsymbol{w}, \boldsymbol{q})) \tag{4.17}$$

berechnet. Dazu wird zunächst für jedes Fragewort die maximale Kosinusähnlichkeit zu den Wortbildern des Dokuments bestimmt und anschließend der Mittelwert über diese Werte gebildet. Schließlich werden alle Dokumente der Kollektion auf Grundlage der berechneten Werte in absteigender Reihenfolge sortiert und die ersten k Dokumente als Ergebnis zurückgegeben.

Antwortextraktion

Bei dem vorgestellten HTR-freien QA-Modell handelt es sich um eine angepasste Variante der textuellen BIDAF-Architektur [180] aus dem NLP-Bereich. Im Gegensatz zur originalen BIDAF-Architektur werden die textuellen Eingaben durch vektorielle Eingaben ersetzt und die Ausgabe von Wort- auf Zeilenebene geändert (siehe Abbildung 4.9). Die im Folgenden als *BIDAF-Line* bezeichnete Architektur ist in sechs Bereiche unterteilt: Worteinbettung, Kontexteinbettung, *Attention-Flow*, Modellierung, Zeileneinbettung und Ausgabe. Im ersten Schritt des Modells werden die maschinenlesbaren Wörter der Frage und die

Wortbilder des Dokuments in Vektorrepräsentationen umgewandelt. Anschließend werden auf der Grundlage dieser Repräsentationen mehrere Matrixoperationen durchgeführt, um die in der Frage und im Dokument enthaltenen Informationen zu kombinieren. Das Ergebnis dieser Operationen ist eine fragespezifische Repräsentation für jedes Wortbild des Dokuments. Die Wortbilder werden entsprechend ihrer Zeilenzugehörigkeit gruppiert und in eine gemeinsame Zeilenrepräsentation umgewandelt. Schließlich wird die fragespezifische Zeilendarstellung in zwei Pseudowahrscheinlichkeitsverteilungen transformiert, welche die Start- und Endzeile der Antwort im Dokument bestimmen.

WORTEINBETTUNG

In der Worteinbettungsschicht werden alle Wortbilder aus einem gegebenen Dokument $\mathbf{D} = \{\mathbf{d}^{(0)}, \ldots, \mathbf{d}^{(n_c-1)}\}$, sowie die textuellen Fragewörter $\mathbf{Q} = \{\mathbf{q}^{(0)}, \ldots, \mathbf{q}^{(n_q-1)}\}$ in einen gemeinsamen d_e-dimensionalen Vektorraum überführt. Dafür werden die Wortbilder mit dem Faltungsnetzwerk aus Abschnitt 4.2.1 in einen textuellen Vektorraum abgebildet. Die Fragewörter werden mit dem entsprechenden vortrainierten textuellen Worteinbettungsmodell in eine Repräsentation $\boldsymbol{Q} \in \mathbb{R}^{(n_q \times d_e)}$ überführt. Für die Wortbilder im Dokument werden zusätzlich die Zeilenkorrespondenzen unter Verwendung der Kodierungsstrategie, $pe : \mathbb{R} \to \mathbb{R}$, aus [42] modelliert und mit der entsprechenden Wortdarstellung konkateniert. Der Vorgang kann formal anhand der Formeln

$$\begin{aligned} \mathrm{pe}(pos, j) &= \begin{cases} \sin(\frac{pos}{10000^{j/d_z}}) & \text{wenn } j \equiv 0 \pmod 2 \\ \cos(\frac{pos}{10000^{(j-1)/d_z}}) & \text{sonst} \end{cases} \\ \boldsymbol{P}_m &= \{\mathrm{pe}(m, k) \mid k \in \{0, \ldots, d_z - 1\}\} \\ \widehat{\boldsymbol{C}}_i &= \boldsymbol{C}_i \parallel \boldsymbol{P}_{\boldsymbol{z}_i} \end{aligned} \tag{4.18}$$

nachvollzogen werden, wobei d_z die Dimensionalität der Zeilenkodierung, $\boldsymbol{P}_{\boldsymbol{z}_i} \in \mathbb{R}^{d_z}$ die Zeilenrepräsentation für das i-te Wortbild aus $\mathbf{D}$ und $\boldsymbol{C}_i \in \mathbb{R}^{d_e}$ dessen Repräsentation ist. Dabei entspricht $\boldsymbol{z}_i \in \{0, \ldots, n_z - 1\}$ der Zeilennummer des Wortbildes an der Position i und wird extern zur Verfügung gestellt.

KONTEXTEINBETTUNG

Die Ausgabe der Worteinbettungsschicht ist eine kontextunabhängige Darstellung der Dokumenten- und Fragewörter. Die Beantwortung einer Frage erfordert jedoch ein tieferes semantisches Verständnis der Frage bzw. des Dokumenteninhalts. Darüber hinaus muss die Zeilenzugehörigkeit der Wortbilder im Dokument sowie deren Beziehungen zueinander modelliert werden. Daher wird ein Einbettungsmechanismus benötigt, der eine kontextabhängige Wortrepräsentation erzeugen kann. In diesem Modell wird zur Kodierung der Kontextinformationen ein BLSTM verwendet. Das Ergebnis der kontextuellen Einbettung ist die Matrix $\boldsymbol{W} \in \mathbb{R}^{(n_c \times h)}$ für den Kontext und die Matrix $\boldsymbol{U} \in \mathbb{R}^{(n_q \times h)}$ für die Frage, welche formal mit

$$\begin{aligned} \boldsymbol{W} &= \mathrm{blstm}_c(\widehat{\boldsymbol{C}}) \\ \boldsymbol{U} &= \mathrm{blstm}_q(\boldsymbol{Q}) \end{aligned} \tag{4.19}$$

bestimmt werden. Hierbei werden die Beziehungen zwischen den Repräsentationen der einzelnen Wortbilder des Dokuments bzw. der Fragewörter mit zwei separaten BLSTM-Modellen, $\mathrm{blstm}_c : \mathbb{R}^{(n_c \times d_z + d_e)} \to \mathbb{R}^{(n_c \times h)}$ bzw. $\mathrm{blstm}_q : \mathbb{R}^{(n_q \times d_e)} \to \mathbb{R}^{(n_q \times h)}$, extrahiert und so kontextabhängige Repräsentationen erzeugt.

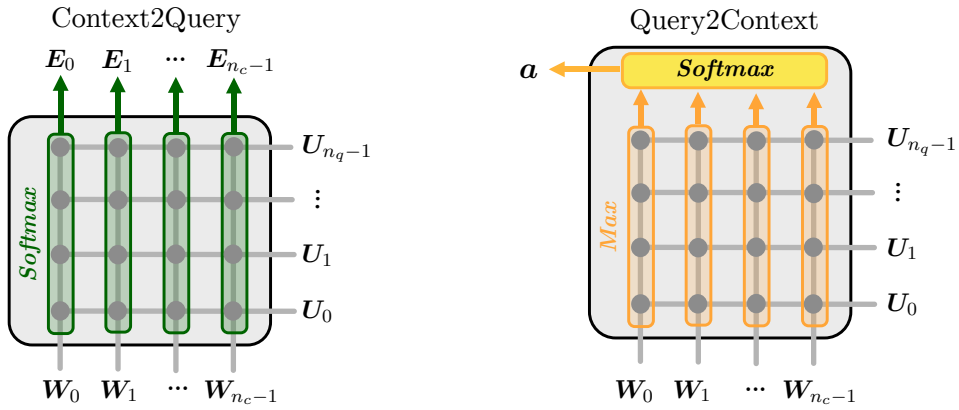


Abbildung 4.10: Eine Übersicht über die Attention-Ansätze zur Extraktion der Beziehungen zwischen den Wortbilddarstellungen des Dokuments $\boldsymbol{W}$ und den Fragewörtern $\boldsymbol{U}$ im BIDAF-Modell. Die grauen Punkte visualisieren die Ähnlichkeiten zwischen den Wortbildern und den Fragewörtern. Die Grafik ist angelehnt an [180].

ATTENTION-FLOW

Das Hauptziel dieses Schrittes ist die Extraktion und Kombination von Informationen aus den kontextabhängigen Dokumenten- und Fragerepräsentationen. Die Ausgabe sind frageabhängige Wortbildrepräsentationen $\widehat{\boldsymbol{W}} \in \mathbb{R}^{(n_c \times h)}$ und $\widehat{\boldsymbol{U}} \in \mathbb{R}^{(n_c \times h)}$. Für die Zusammenführung der Informationen wird der Attention-Mechanismus verwendet. Dieser basiert auf einer gemeinsamen Ähnlichkeitsmatrix $\boldsymbol{S} \in \mathbb{R}^{(n_c \times n_q)}$ zwischen den kontextbezogenen Einbettungen der Wortbilder und Fragewörter. Dabei entspricht

$$\boldsymbol{S}_{t,j} = \mathrm{dcos}(\boldsymbol{W}_t, \boldsymbol{U}_j) \tag{4.20}$$

der Kosinusähnlichkeit zwischen der Repräsentation des Wortbildes an der Position t und dem Fragewort an der Position j. Die Attention-Werte werden auf Basis der *Context2Query*- und *Query2Context*-Attention bestimmt. Context2Query berechnet eine Matrix $\boldsymbol{E} \in \mathbb{R}^{(n_c \times n_q)}$ mit

$$\boldsymbol{E}_t = \mathrm{softmax}(\boldsymbol{S}_t), \tag{4.21}$$

welche die Relevanz der Fragewörter für die Wortbilder darstellt. Hierbei werden die Attention-Gewichte für das Wortbild an der Position t durch Anwendung der Softmax-Funktion auf die t-te Zeile von $\boldsymbol{S}$ bestimmt. Query2Context erzeugt hingegen einen Vektor $\boldsymbol{a} \in \mathbb{R}^{n_c}$ mit

$$\boldsymbol{a} = \mathrm{softmax}(\{\max_{j \in \{0,\ldots,n_q-1\}} \boldsymbol{S}_{t,j} \mid t \in \{0,\ldots,n_c-1\}\}), \tag{4.22}$$

wobei der resultierende Vektor angibt, welches Wortbild einem der Fragewörter am ähnlichsten und damit für die Beantwortung der Frage entscheidend ist. Formal wird für jedes Wortbild zunächst der maximale Ähnlichkeitswert in $\boldsymbol{S}$ berechnet und anschließend die Softmax-Funktion auf die Ergebnisse angewendet. Zur besseren Übersicht sind die Berechnungen der Attention-Ansätze in Abbildung 4.10 visualisiert. Mit den Attention-Gewichten werden anschließend die frageabhängigen Wortbildrepräsentationen $\widehat{\boldsymbol{W}}$ und $\widehat{\boldsymbol{U}}$ erzeugt. Hierbei repräsentiert $\widehat{\boldsymbol{U}}$ die Relevanz jedes Fragewortes für jedes Wortbild und $\widehat{\boldsymbol{W}}$ fasst die Informationen zu den wichtigsten Wortbildern im Dokument bezüglich der Frage zusammen. Formal wird $\widehat{\boldsymbol{U}}_t$ für das Wortbild an der Position t mit

$$\widehat{\boldsymbol{U}}_t = \sum_{j=0}^{n_q-1} \boldsymbol{U}_j \cdot \boldsymbol{E}_{t,j} \tag{4.23}$$

bestimmt. Die frageabhängigen Wortbildrepräsentationen $\widehat{\boldsymbol{W}}$ werden durch eine n_c-fache Replikation des Vektors $\widehat{\boldsymbol{w}} \in \mathbb{R}^h$ bestimmt. Dieser wird formal mit

$$\widehat{\boldsymbol{w}} = \sum_{k=0}^{n_c-1} \boldsymbol{W}_k \cdot \boldsymbol{a}_k \tag{4.24}$$

berechnet und stellt eine gewichtete Summe der kontextsensitiven Wortbildrepräsentationen in Abhängigkeit der Attention-Gewichte $\boldsymbol{a}$ dar.

MODELLIERUNG

In dieser Phase werden die generierten Wortbildrepräsentationen aus den vorherigen Schichten zunächst in eine gemeinsame Darstellung $\boldsymbol{G} \in \mathbb{R}^{(n_c \times 4 \cdot h)}$ mit

$$\boldsymbol{G}_t = \boldsymbol{W}_t \parallel \widehat{\boldsymbol{U}}_t \parallel \boldsymbol{W}_t \odot \widehat{\boldsymbol{U}}_t \parallel \boldsymbol{W}_t \odot \widehat{\boldsymbol{W}}_t \tag{4.25}$$

überführt. Die Matrix enthält alle Informationen aus $\boldsymbol{W}$, $\widehat{\boldsymbol{W}}$ und $\widehat{\boldsymbol{U}}$. Jeder Spaltenvektor in $\boldsymbol{G}$ entspricht einer vektoriellen Darstellung eines Wortbildes aus $\mathbf{D}$, welche die relevanten Frage- und Kontextinformationen kodiert. Die Repräsentation dient als Eingabe für ein zweistufiges BLSTM-Modell, $\text{blstm}_{dq} : \mathbb{R}^{(n_c \times 4 \cdot h)} \rightarrow \mathbb{R}^{(n_c \times h)}$, welches die Beziehungen zwischen der Frage und dem Kontext modelliert und die Ausgabe $\boldsymbol{M} \in \mathbb{R}^{(n_c \times h)}$ erzeugt.

ZEILENEINBETTUNG

Aufgrund der Umstellung der Modellausgabe von Wort- auf Zeilenebene wird in diesem Schritt die Wortrepräsentation $\boldsymbol{M}$ auf die Anzahl der Zeilen, n_z, im Dokument reduziert. Dazu werden die Wortrepräsentationen entsprechend ihrer Zeilenzugehörigkeit im Dokument summiert und ergeben die Zeilenrepräsentation $\boldsymbol{Z} \in \mathbb{R}^{(n_z \times h)}$. Dabei wird die Zeilenzugehörigkeit extern bereitgestellt.

AUSGABE

Das Ausgabeformat des Systems besteht aus einer Anfangs- und einer Endzeile im Dokument, sodass alle Zeilen zwischen diesen beiden Werten die Antwort auf die gestellte Frage definieren. Dazu wird die Zeilenrepräsentation $\boldsymbol{Z}$ in zwei Pseudowahrscheinlichkeitsverteilungen $\boldsymbol{s} \in \mathbb{R}^{n_z}$ und $\boldsymbol{e} \in \mathbb{R}^{n_z}$ überführt, welche für jede Zeile des Dokuments bestimmen, ob es sich um die Start- bzw. Endzeile der Antwort handelt. Die Bestimmung der Startzeilenverteilung wird durch die Anwendung einer vollvernetzten Schicht mit einem Ausgabeneuron auf jede Zeilendarstellung des Dokuments erreicht. Für die Ermittlung der Endzeilenverteilung werden die Zeilendarstellungen zunächst mit einem BLSTM verarbeitet und auf dessen Ausgaben eine vollvernetzte Schicht mit einem Ausgabeneuron angewendet. Die Start- und Endzeile der Antwort werden durch den Index mit dem höchsten Wert in der jeweiligen Verteilung bestimmt.

Da für alle Dokumente aus dem Retrieval eine Antwort ermittelt wird, ist eine Entscheidung über die endgültige Antwort des QA-Systems erforderlich. Zu diesem Zweck wird in dieser Arbeit ein Konfidenzwert $z \in \mathbb{R}$ mit

$$z = \max_{i \in \{0,\ldots,n_z-1\}} \boldsymbol{s}_i + \max_{i \in \{0,\ldots,n_z-1\}} \boldsymbol{e}_i \tag{4.26}$$

für jedes Dokument definiert, welcher der Summe der Aktivierungen für die vorhergesagten Start- und Endzeilenindizes entspricht. Der Bildausschnitt mit dem höchsten Konfidenzwert wird schließlich als Antwort des HTR-freien QA-Systems für die vorliegende Frage ausgegeben.

5 CROSS-MODALE WISSENSDESTILLATION

Eines der Hauptziele dieser Arbeit ist die Integration von semantischem Weltwissen in HTR-freie Modelle und die Beantwortung der Forschungsfrage, ob dies die Defizite im Vergleich zu HTR-basierten Ansätzen aus der Literatur beheben kann. Zur Beantwortung dieser Forschungsfrage wurde im vorherigen Kapitel bereits ein HTR-basiertes und ein HTR-freies Modell vorgestellt. In diesem Kapitel wird ein effizienter und robuster Ansatz zur Integration von semantischem Vorwissen in das HTR-freie Modell präsentiert.

Die Lösung semantischer Aufgaben erfordert in der Regel ein Verständnis des Textinhalts, insbesondere der semantischen Eigenschaften von Wörtern. Für maschinenlesbare Texte existieren bereits leistungsfähige Modelle, welche die semantischen Eigenschaften von Wörtern in einem kontinuierlichen Vektorraum kodieren [182]. Im NLP-Bereich konnte der Vorteil für die Verwendung dieser vortrainierten semantischen Worteinbettungen bereits erfolgreich demonstriert werden. In nahezu allen Benchmarks erzielen Ansätze, die auf semantischen Worteinbettungen zurückgreifen, signifikant bessere Ergebnisse als solche, die ausschließlich auf aufgabenspezifischen Trainingsdaten basieren [42, 182]. Diese Erkenntnisse werden derzeit noch nicht im Bereich der Handschrift angewendet, was vermutlich auf das Fehlen von Modellen zur Vorhersage semantischer Eigenschaften für Wortbilder zurückzuführen ist. Insbesondere im HTR-freien Kontext werden die Modellparameter üblicherweise zufällig initialisiert und Ende-zu-Ende mit den Trainingsdaten des Benchmarks an die semantische Aufgabe angepasst [1, 133, 207]. Die Übertragung des Lernverfahrens zur semantischen Repräsentation von maschinenlesbaren Wörtern auf handschriftliche Wortbilder ist jedoch nicht trivial [90, 110]. Insbesondere die hohe Variabilität von Handschrift im Vergleich zu maschinenlesbaren Texten und die relativ geringe Anzahl repräsentativer handschriftlicher Daten mit Segmentierungs- und Textannotationen erschweren die Übertragung des Ansatzes. Darüber hinaus erfordert die Umsetzung eines solchen Lernverfahrens erhebliche finanzielle und technische Ressourcen. So haben textbasierte semantische Modelle bereits Parameter im niedrigen Milliardenbereich, wobei die zusätzlich zu verarbeitenden visuellen Informationen diese Anforderungen nochmals erheblich erhöhen [90, 110].

Der in dieser Arbeit vorgestellte Ansatz zur HTR-freien semantischen Repräsentation von handschriftlichen Wortbildern vermeidet ein ressourcenintensives Training. Stattdessen wird das bereits kodierte semantische Wissen aus vortrainierten textuellen Worteinbettungsmodellen effizient zur Bestimmung der Wortbildrepräsentationen verwendet. Dazu wird eine Abbildung von Wortbildern mit einem neuronalen Modell in einen textuellen semantischen Worteinbettungsraum aus dem NLP-Bereich gelernt. Die semantischen Eigenschaften der Wörter müssen somit nicht von Grund auf gelernt werden, sondern lediglich eine HTR-freie Abbildung von Wortbildern auf die semantische Repräsentation des in ihnen enthaltenen Textes. Dies ermöglicht nach dem Training die HTR-freie Vorhersage von semantischen Eigenschaften für handschriftliche Wortbilder. Das Konzept dieses Ansatzes beruht auf der Tatsache, dass die Semantik eines handschriftlichen Wortbildes durch den in ihm enthaltenen Text definiert ist. Konkret wird in dieser Arbeit das CNN-Modell aus dem vorgestellten HTR-freien Ansatz zur Realisierung der Abbildungsfunktion verwen-

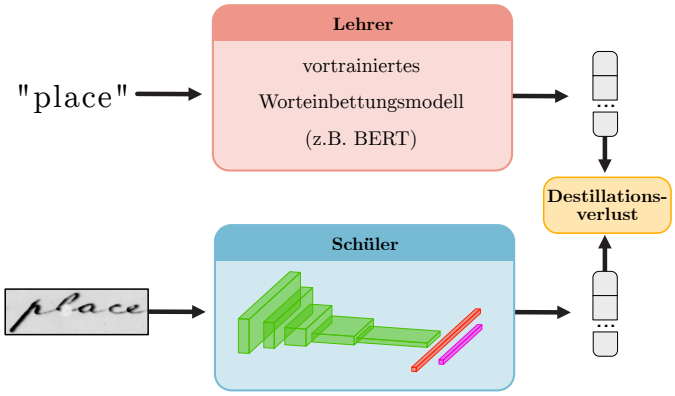


Abbildung 5.1: Ein Überblick des Ansatzes zur cross-modalen Wissensdestillation. Semantische Informationen werden durch ein vortrainiertes textuelles Worteinbettungsmodell kodiert. Das Ziel ist die Vorhersage einer Wortbildeinbettung mit dem Schülermodell, die der Repräsentation des Lehrermodells für den zugehörigen Text des Wortbildes möglichst ähnlich ist. Der Destillationsverlust misst die Abweichung zwischen den erzeugten Repräsentationen des Lehrer- und Schülermodells.

det. Auf diese Weise können die gelernten semantischen Informationen beim Training des HTR-freien Modells für anwendungsspezifische Aufgaben berücksichtigt werden.

Aufgrund der HTR-freien Anforderung ist es notwendig, das semantische Wissen aus dem textuellen in den visuellen Bereich modalitätsübergreifend zu transferieren. Dieser Prozess wird in der Literatur als cross-modale Wissensdestillation (engl.: *Cross-modal Knowledge Distillation*) [64] bezeichnet und ist in Abbildung 5.1 dargestellt. Im Allgemeinen wird bei diesem Verfahren ein Lehrermodell (engl.: *Teacher Model*) auf einer Eingabemodalität vortrainiert und dessen Wissen auf ein Schülermodell (engl.: *Student Model*) übertragen, das eine abweichende Modalität als Eingabe verwendet. In dieser Arbeit dient ein vortrainiertes Worteinbettungsmodell aus dem NLP-Bereich als Lehrer und das im Abschnitt 4.2.1 vorgestellte CNN-Modell als Schüler. Die Eingabe des Lehrers ist ein maschinenlesbares Wort, während die Eingabe des Schülers durch ein handgeschriebenes Wortbild repräsentiert wird. Das Schülermodell wird mit einem vollständig überwachten Lernverfahren trainiert. Dazu steht ein annotierter Trainingsdatensatz zur Verfügung, der aus handgeschriebenen Wortbildern und deren textuellen Annotationen besteht. Die Annotationen werden zur Erstellung der semantischen Zielrepräsentationen für die Wortbilder auf Basis des Lehrermodells verwendet. Das Ziel ist die Bestimmung einer semantischen Wortbildrepräsentation für ein gegebenes Wortbild mit dem Schülermodell, die der vorhergesagten Repräsentation des Lehrermodells für die textuelle Annotation des Wortbildes möglichst ähnlich ist. Dazu wird der sogenannte Destillationsverlust (engl.: *Distillation Loss*) verwendet, der die Abweichung zwischen der Vorhersage des Schülermodells ($\hat{\boldsymbol{y}} \in \mathbb{R}^d$) und des Lehrermodells ($\boldsymbol{y} \in \mathbb{R}^d$) erfasst und minimiert. Der mittlere quadratische Fehler wird in dieser Arbeit als Verlustfunktion verwendet und ist für die vorliegenden Vorhersagen mit

$$\text{mse}(\boldsymbol{y}, \hat{\boldsymbol{y}}) = \frac{1}{d} \sum_{i=0}^{d-1} (\boldsymbol{y}_i - \hat{\boldsymbol{y}}_i)^2 \tag{5.1}$$

definiert. Durch die Minimierung dieses Fehlers wird das Schülermodell zunehmend genauer in Bezug auf dieselbe Vorhersage wie das Lehrermodell und überführt damit das semantische Wissen aus dem textuellen Bereich in die visuelle Handschriftdomäne.

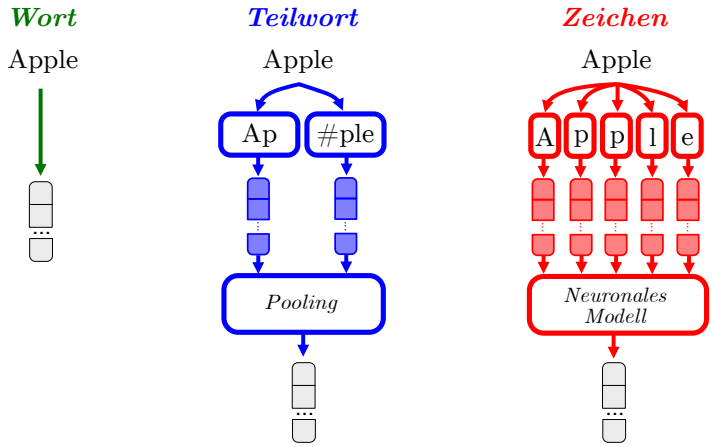


Abbildung 5.2: Ein Überblick über die am häufigsten verwendeten Ansätze zur Kodierung von Eingabewörtern in semantischen Worteinbettungsmodellen aus dem NLP-Bereich. Die Grafik ist inspiriert von [49].

Ein wesentlicher Parameter dieses Destillationsansatzes ist die Wahl eines geeigneten Lehrermodells. Zur Auswahl des Modells werden im Abschnitt 5.1 vielversprechende Worteinbettungsmodelle aus dem NLP-Bereich vorgestellt. Wie nachfolgend ausführlich beschrieben wird, ergibt sich eine Einschränkung dieses Destillationsansatzes für Wortbilder, deren textuelle Annotationen nicht im Training des Destillationsprozesses vorkamen. Um dieses Problem zu mildern, wird im Abschnitt 5.2 eine annotationsfreie Destillationsstrategie vorgestellt, die synthetisch erzeugte Wortbilder verwendet, um die Menge der nicht im Training vorkommenden Wörter zu minimieren. Ein weiterer Lösungsansatz für dieses Problem wird im Abschnitt 5.3 vorgestellt. Hierbei wird eine geeignete Kombination aus einer semantischen und syntaktischen Worteinbettung verwendet, um im Falle einer fehlerhaften semantischen Vorhersage zumindest die orthographischen Informationen des im Wortbild enthaltenen Textes zu repräsentieren. Abschließend werden im Abschnitt 5.4 drei Strategien zur Integration des Schülermodells in den HTR-freien Ansatz vorgestellt.

5.1 SEMANTISCHE WORTEINBETTUNGSVERFAHREN

Wie bereits erwähnt, ist die Wahl eines geeigneten Lehrermodells ein zentraler Parameter bei der Wissensdestillation. Dabei sollte das Schülermodell die Ausgabefunktion des Lehrermodells bestmöglich approximieren und nicht nur die semantischen Repräsentationen für die Wörter aus dem Training auswendig lernen. Dazu wird ein Lehrermodell benötigt, welches neben der Kodierung von leistungsfähigen semantischen Wortrepräsentationen auch eine hohe Korrelation zwischen den orthografischen und semantischen Informationen von Wörtern aufweist. Dieser Zusammenhang ermöglicht die Vorhersage semantischer Informationen auch für nicht im Destillationsprozess enthaltene Wörter und erlaubt eine effiziente Approximation der Abbildungsfunktion mit einer begrenzten Anzahl annotierter Trainingsdaten.

Die semantischen Worteinbettungsmodelle aus dem NLP-Bereich verfolgen konzeptionell unterschiedliche Strategien zur Kodierung eines Eingabewortes. Die Ansätze haben eine hohe Auswirkung auf die Korrelation zwischen den orthografischen Merkmalen von Wörtern und ihrer semantischen Repräsentation [5, 150]. Im Allgemeinen wird zwischen wort-, teilwort- und zeichenbasierten Ansätzen unterschieden (siehe Abbildung 5.2). Der

wortbasiert Ansatz bietet eine intuitive und leistungsstarke semantische Kodierung, wobei für jedes Wort eine eigene Wortrepräsentation trainiert wird [222]. Der Ansatz kann jedoch nur für Wörter aus einem vorgegebenen Vokabular Repräsentationen ermitteln und ist daher anfällig gegenüber Rechtschreibfehlern und Wortneubildungen [18, 150, 224]. Zudem erfordert ein derartiges System aufgrund der meist umfangreichen Anzahl von Wörtern im Vokabular einen erheblichen Speicherbedarf [137]. Der zeichenbasierte Ansatz verarbeitet die Zeichen der Eingabe zunächst individuell und kombiniert deren Einbettungen anschließend zu einer gemeinsamen Wortrepräsentation. Die Kombination erfolgt in der Regel mit einem neuronalen Modell [5, 49, 91]. Im Vergleich zum wortbasierten Ansatz ist der Speicherbedarf sehr gering und es können auch Wörter außerhalb des Trainingsvokabulars verarbeitet werden [5]. Ein weiterer Vorteil des zeichenbasierten Ansatzes ist seine hohe Robustheit gegenüber Rechtschreibfehlern [49]. Allerdings kodieren diese Arten von Einbettungen eine unzureichende semantische Qualität im Vergleich zu den wortbasierten Modellen [49]. Der teilwortbasiert Ansatz interpoliert zwischen der wort- und zeichenbasierten Kodierung [238]. Das Vokabular besteht aus den häufigsten verwendeten Wörtern aus einer Trainingsmenge, sowie aus Teilwörtern und Einzelbuchstaben. Dadurch ist es möglich, leistungsfähige semantische Repräsentationen zu kodieren und gleichzeitig Wörter außerhalb des Vokabulars mit möglichst wenigen Elementen zu rekonstruieren [42, 238].

Die textuellen Worteinbettungsverfahren aus der Literatur können grundsätzlich in statische [18, 137] und kontextbasierte [5, 42, 149] Methoden unterteilt werden. Statische Methoden generieren Worteinbettungen unabhängig von ihrem Kontext und bilden somit ein Wort immer auf die gleiche Vektordarstellung ab. Diese Limitierung ist nicht realitätsnah, da ein Wort je nach Kontext verschiedene Bedeutungen haben kann, z.B. Bank als Kreditinstitut oder Sitzmöglichkeit. Ungeachtet der theoretischen Vorteile und der signifikant besseren Ergebnisse bei der Verwendung von kontextsensitiven gegenüber statischen Einbettungen auf fast allen Benchmarks im NLP-Bereich [50], werden im Bereich der semantischen Analyse von handschriftlichen Dokumentenbildern bisher nur statische Worteinbettungen verwendet [98, 214, 233]. Dies ist vor allem darauf zurückzuführen, dass bereits die Abbildung von Wortbildern zu statischen Worteinbettungen eine komplexe Aufgabe darstellt und nach wie vor eine offene Forschungsfrage ist [214]. Eine potentielle Lösung zur Nutzung kontextabhängiger Ansätze für die statische Wortbildrepräsentation stellt die Extraktion von statischen Worteinbettungen aus kontextsensitiven semantischen Repräsentationen dar [50]. Diese extrahierten Wortrepräsentationen übertreffen statische Ansätze auf einer Vielzahl von semantischen Benchmarks [50]. Basierend auf diesen Erkenntnissen erfolgt die Auswahl des Lehrermodells in dieser Arbeit auf Basis der bekanntesten und am weitesten verbreiteten statischen (Word2Vec [137] und FastText [18]) und kontextsensitiven (Flair [4], ELMo [149] und BERT [42]) Worteinbettungsverfahren des NLP-Bereichs.

WORD2VEC

Das Word2Vec-Verfahren [137] ist einer der ersten Ansätze im NLP-Bereich zur Modellierung von semantischen Eigenschaften für Wörter mit kompakten Vektorrepräsentationen. Die Extraktion der semantischen Eigenschaften basiert auf der Verteilungshypothese [173]. Diese besagt, dass Wörter, die in demselben Kontext vorkommen, semantisch zusammengehören. Zur Realisierung des Verfahrens werden Abbildungen von Wörtern mit einem neuronalen Netzwerk und einem Trainingskorpus auf reellwertige Vektoren gelernt, sodass der Abstand zwischen zwei Wörtern im erzeugten Vektorraum ihrer semantischen

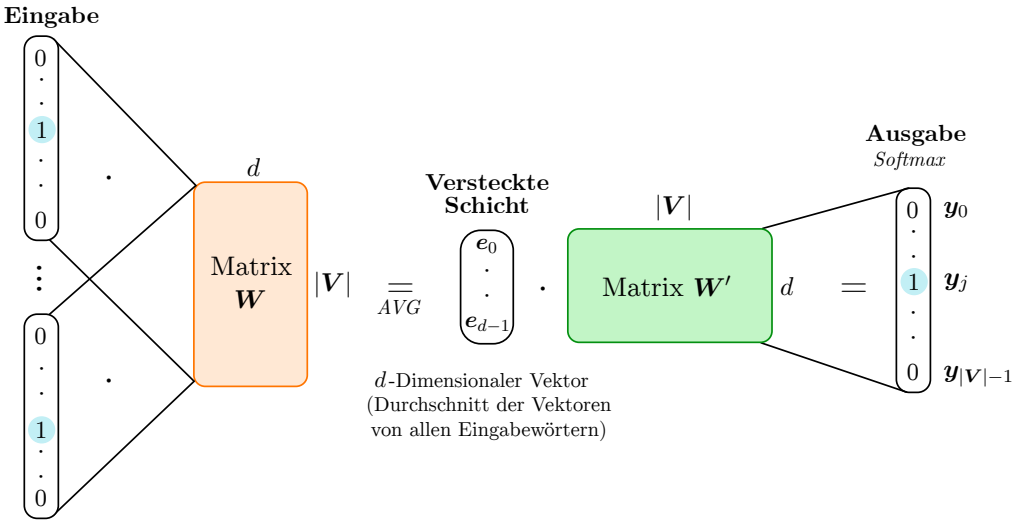


Abbildung 5.3: Ein Überblick über die CBoW-Architektur, die in den Ansätzen Word2Vec und FastText zum Lernen semantischer Worteinbettungen verwendet wird. Die Grafik ist angelehnt an [229].

Ähnlichkeit entspricht. Der Trainingskorpus $\mathbf{K} = \{\mathbf{w}^{(0)}, \ldots, \mathbf{w}^{(k-1)}\}$ besteht ausschließlich aus nicht annotiertem, maschinenlesbarem Text und wird in Beispiele der Form (Kontext, Zielwort) unterteilt. Der Kontext wird als eine Menge von Wörtern definiert, die in einem Fenster von fester Größe c um das Zielwort vorkommen.

Für das Training kann zwischen den beiden neuronalen Netzwerken *Continuous Bag-of-Words* (CBoW) oder *Skip-gram* [137] ausgewählt werden. Obwohl sich die Modelle in ihren konzeptionellen Ansätzen deutlich unterscheiden, erzielen sie auf den meisten Benchmarks nahezu identische Ergebnisse [137]. Das Skip-gram-Modell wird auf die Vorhersage der Kontextwörter für das gegebene Zielwort trainiert, während das CBoW-Modell das Zielwort anhand der Kontextwörter vorhersagt. In der Praxis bietet letzteres sowohl eine kürzere Trainingszeit als auch qualitativ hochwertigere Wortdarstellungen [137]. Die Architektur dieses Modells beruht auf einem zweischichtigen neuronalen Netzwerk mit einer verborgenen Schicht (siehe Abbildung 5.3). Die Ein- und Ausgabewörter werden mit einer 1-aus-$|\mathbf{V}|$-Kodierung anhand eines festen Vokabulars $\mathbf{V}$ repräsentiert. Die Anzahl der Neuronen in der Eingabe- und Ausgabeschicht ist gleich $|\mathbf{V}|$. Die Kontextwörter werden mit einer vordefinierte Fenstergröße c aus einer gegebenen Wortfolge extrahiert und sind für die Position t der Wortfolge mit

$$\mathbf{F}^{(t)} = \{\mathbf{w}^{(t-c)}, \ldots, \mathbf{w}^{(t-1)}, \mathbf{w}^{(t+1)}, \ldots, \mathbf{w}^{(t+c)}\} \tag{5.2}$$

definiert. Jedes Kontextwort wird zunächst mit der linearen Schicht $\boldsymbol{W} \in \mathbb{R}^{(|\mathbf{V}| \times d)}$ auf eine d-dimensionale Wortrepräsentation abgebildet. Dabei sei $\boldsymbol{e}^{(\mathbf{w})} = \boldsymbol{W}_{i_\mathbf{w}} \in \mathbb{R}^d$ die Eingaberepräsentation des Wortes $\mathbf{w}$, wobei $i_\mathbf{w}$ der Index des Wortes $\mathbf{w}$ in $\mathbf{V}$ ist. Anschließend werden die Repräsentationen gemittelt und so in einen gemeinsamen Kontextvektor $\boldsymbol{m}^{(t)} \in \mathbb{R}^d$ mit

$$\boldsymbol{m}^{(t)} = \frac{1}{|\mathbf{F}^{(t)}|} \cdot \sum_{-c \leq j \leq c, j \neq 0} \boldsymbol{e}^{(\mathbf{w}^{(t+j)})} \tag{5.3}$$

überführt. Dieser Kontextvektor wird schließlich mit einer weiteren linearen Schicht $\boldsymbol{W}' \in \mathbb{R}^{(d \times |\mathbf{V}|)}$ und einer Softmax-Funktion in eine Pseudowahrscheinlichkeitsverteilung über $\mathbf{V}$ überführt. Die Verteilung gibt für jedes Wort des Vokabulars die Wahrscheinlichkeit an, im

gegebenen Kontext das Zielwort zu sein. Dies führt in Analogie zur Verteilungshypothese dazu, dass Zielwörter, die im gleichen Kontext vorkommen, auch ähnliche Repräsentationen haben. Das Training des Modells für den Trainingskorpus $\mathbf{K}$ und der Fensterbreite c erfolgt durch die Minimierung der Funktion

$$\mathrm{l}(\mathbf{K}, c) = -\ \frac{1}{|\mathbf{K}| - 2 \cdot c} \cdot \sum_{t=c}^{|\mathbf{K}|-c-1} \mathrm{P}(\mathbf{w}^{(t)} \mid \mathbf{F}^{(t)}\ ) \tag{5.4}$$

Diese Funktion führt zu einer Maximierung der mittleren logarithmischen Wahrscheinlichkeiten für die Vorhersage des Zielwortes in Abhängigkeit der Kontextwörter. Die Pseudowahrscheinlichkeit für die Vorhersage des Zielwortes $\mathbf{w}^{(t)}$ in Abhängigkeit der Kontextwörter $\mathbf{F}^{(t)}$ ist mit

$$\mathrm{P}\left(\mathbf{w}^{(t)} \,\middle|\, \mathbf{F}^{(t)}\right) = \frac{\exp(\boldsymbol{a}^{(\mathbf{w}^{(t)})} \cdot \boldsymbol{m}^{(t)})}{\sum_{l=0}^{|\mathbf{V}|-1} \exp(\boldsymbol{a}^{(\mathbf{w}^{(l)})} \cdot \boldsymbol{m}^{(t)})} \tag{5.5}$$

definiert. Dabei sei $\boldsymbol{a}^{(\mathbf{w})} = \boldsymbol{W}'^{\mathsf{T}}_{:,i_{\mathbf{w}}} \in \mathbb{R}^{(1\times d)}$ die gelernte Ausgaberepräsentation des Wortes $\mathbf{w}$ und $\boldsymbol{m}^{(t)}$ die gemeinsame Vektorrepräsentation der Kontextwörter aus $\mathbf{F}^{(t)}$.

Nach dem Training wird für eine Wort $\mathbf{w}$ die Ausgaberepräsentationen $\boldsymbol{a}^{(\mathbf{w})}$ als semantische Worteinbettung verwendet. Normalerweise kann eine Vorhersage nur für Wörter aus dem Vokabular durchgeführt werden. In der Praxis erfolgt die Kodierung von Wörtern außerhalb des Vokabulars jedoch durch eine sogenannte *out-of-vocabulary*-Repräsentation, die dem zeilenweisen Mittelwert von $\boldsymbol{W}'$ entspricht. Diese Darstellung kodiert wenig bis gar keine semantische Information und erlaubt keine Unterscheidung von Wörtern außerhalb des Vokabulars.

In dieser Arbeit wir ein vortrainiertes Word2Vec-Modell[1] mit 1.8 Milliarden Parametern und einer Ausgabedimensionalität von 300 verwendet. Das Modell wurde mit der CBoW-Architektur auf einer Teilmenge des *Google-News*-Datensatzes mit über 100 Milliarden Wörtern, einer Vokabulargröße von drei Millionen Wörtern und einer Fenstergröße von 5 trainiert.

FASTTEXT

Das Word2Vec-Verfahren berechnet für jedes Wort des Vokabulars eine individuelle Repräsentation und ignoriert dabei die interne Struktur von Wörtern. Dies führt zu einer unzureichenden Darstellung von semantischen Eigenschaften für Wörter außerhalb des Vokabulars [18]. Zur Lösung dieses Problems wurde mit FastText [18] eine Erweiterung von Word2Vec realisiert, welche es ermöglicht, auch für nicht im Vokabular vorkommende Wörter unterscheidbare semantische Vektoren vorherzusagen. FastText nutzt dazu die interne Struktur von Wörtern in morphologischen Sprachen, deren Semantik häufig auf der Basis der enthaltenen Teilwörter bestimmt werden kann [18]. Die Architektur und das Training sind analog zum Word2Vec-Verfahren, wobei lediglich das Vokabular um eine Menge von Teilwörtern (*n-gramme*) erweitert wird. Für das Wort *fasttext* und $n = 3$ ergeben sich z.B. die Teilwörter *<fa , fas, ast, stt, tte, tex, ext, xt>* und *<fasttext>*. Für die endgültige Wortrepräsentation $\boldsymbol{e}^{(\mathbf{w})} \in \mathbb{R}^d$ des Wortes $\mathbf{w}$ werden die Teilwörter zunächst in ihre Vektorrepräsentationen überführt und anschließend mit

$$\boldsymbol{e}^{(\mathbf{w})} = \sum_{\mathbf{k} \in \mathbf{G}^{(\mathbf{w})}} \boldsymbol{z}^{(\mathbf{k})} \tag{5.6}$$

1 Word2Vec-Modell: https://code.google.com/archive/p/word2vec/

aufsummiert. Dabei sei $\mathbf{G}^{(\mathbf{w})}$ die Menge der Teilwörter aus dem Vokabular die in Wort $\mathbf{w}$ vorkommen und $\boldsymbol{z}^{(\mathbf{k})} = \boldsymbol{W}'_{:,i_\mathbf{k}} \in \mathbb{R}^d$ die gelernte Repräsentation des Teilworts $\mathbf{k} \in \mathbf{G}^{(\mathbf{w})}$, wobei $i_\mathbf{k}$ der Index von $\mathbf{k}$ im Vokabular $\mathbf{V}$ ist.

In dieser Arbeit wird ein auf dem *Common-Crawl*-Datensatz[2] vortrainiertes FastText-Modell[3] mit 1.2 Milliarden Parametern und einer Ausgabedimensionalität von 300 verwendet. Dieses wurde mit der CBoW-Architektur, einem Vokabular von 2 Millionen Wörtern, Teilwörtern der Länge 5 und einer Fenstergröße von 5 trainiert.

ELMO

Word2Vec und FastText erzeugen statische Wortrepräsentationen und ignorieren damit die Mehrdeutigkeit von Wörtern in unterschiedlichen Kontexten. Einer der ersten Ansätze zur Lösung dieses Problems ist das *Embeddings from Language Model* (ELMo)-Verfahren [149], welches neben den syntaktischen und semantischen Eigenschaften eines Wortes auch dessen Kontext kodiert. Dazu werden Wortrepräsentationen auf der Basis eines vortrainierten Sprachmodells extrahiert. Die Architektur des verwendeten Sprachmodells besteht im Wesentlichen aus einem *Character-Level Convolutional Neural Network* (CharCNN) [91] zur Umwandlung der Eingabewörter in statische Wortrepräsentationen und einem mehrschichtigen BLSTM zur Modellierung des Kontextes. Dazu wird jedes Wort aus der Eingabe $\mathbf{K} = \{\mathbf{w}^{(0)}, \ldots, \mathbf{w}^{(k-1)}\}$ zunächst mit dem CharCNN in eine kontextunabhängige Darstellung mit fester Dimensionalität transformiert. Formal wird das Wort $\mathbf{w}^{(i)}$ in Abhängigkeit der enthaltenen Zeichen $\mathbf{Z}^{(i)} = \{\mathbf{c}^{(0)}, \ldots, \mathbf{c}^{(l-1)}\}$ in eine Matrix $\boldsymbol{C}^{(i)} \in \mathbb{R}^{(l \times d)}$ überführt, wobei d die Dimensionalität der lernbaren Zeicheneinbettungen ist. Anschließend werden n-gram ähnliche Informationen mittels Faltungsoperationen zwischen $\boldsymbol{C}$ und mehreren lernbaren Filtern mit unterschiedlichen Filtergrößen berechnet. Durch die Anwendung einer Pooling-Operation auf die Filterergebnisse wird eine Repräsentation pro Wort mit fester Dimensionalität erreicht. Diese statischen Wortrepräsentationen dienen als Eingabe für das BLSTM und ermöglichen die Modellierung kontextabhängiger Repräsentationen der Eingabewörter. Für die Vorhersage des i-ten Wortes im Training wird die i-te Ausgabe des BLSTMs mit einer linearen Schicht auf einen Vektor abgebildet. Die Dimensionalität des Vektors entspricht der Größe des vordefinierten Vokabulars $\mathbf{V}$ und jede Dimension korrespondiert mit einem Wort aus $\mathbf{V}$. Das Modell wird auf der Grundlage einer gegebenen Wortsequenz bidirektional auf die Vorhersage des nächsten Wortes trainiert und dabei die logarithmische Wahrscheinlichkeit für die Vorwärts- und Rückwärtsrichtung maximiert:

$$\mathrm{l}(\mathbf{K}) = \sum_{t=0}^{k-1} \left(\log \left(\mathrm{P}\left(\mathbf{w}^{(t)} | \mathbf{w}^{(0)}, \ldots, \mathbf{w}^{(t-1)} \right) \right) + \log \left(\mathrm{P}\left(\mathbf{w}^{(t)} | \mathbf{w}^{(t+1)}, \ldots, \mathbf{w}^{(n-1)} \right) \right) \right) \tag{5.7}$$

Es existieren verschiedene Möglichkeiten zur Extraktion von Wortrepräsentationen auf Grundlage der mehrstufigen BLSTMs und des CharCNNs. In der Praxis haben sich neben der Verwendung von einzelnen Schichten auch die Konkatenation und die gewichtete elementweise Summe der Ausgaben bewährt [149].

In dieser Arbeit wird ein auf dem *One-Billion-Word-Corpus* [26] vortrainiertes ELMo-Modell[4] mit 93.6 Millionen Parametern verwendet. Das Modell erzeugt Wortrepräsentatio-

2 FastText-Modell: https://commoncrawl.org/
3 crawl-300d-2M-subword: https://fasttext.cc/docs/en/english-vectors.html
4 ELMo-Modell: https://allenai.org/allennlp/software/elmo

nen mit einer Dimensionalität von 1024, wobei diese aus der ersten Schicht des BLSTMs extrahiert werden.

FLAIR

Flair [5] ist ein effizientes Sprachmodell auf Zeichenebene zum Erlernen kontextabhängiger Wortrepräsentationen, das sowohl syntaktische als auch semantische Eigenschaften erfasst. Das Training des Sprachmodells erfolgt im Gegensatz zum ELMo-Modell auf Zeichenebene und nicht auf Wortebene. Das Modell wird auf die Vorhersage des nächsten Zeichens $\mathbf{c}^{(t)}$ trainiert, wobei dessen Kontext über ein mehrschichtiges BLSTM kodiert wird. Beim Vorwärtsmodell erfolgt die Vorhersage auf der Zeichenfolge $\{\mathbf{c}^{(0)}, \ldots, \mathbf{c}^{(t-1)}\}$ und beim Rückwärtsmodell auf der Zeichenfolge $\{\mathbf{c}^{(t+1)}, \ldots, \mathbf{c}^{(k-1)}\}$. Das Modell basiert auf einem festen Vokabular $\mathbf{V}$. Jedem Zeichen $\mathbf{c} \in \mathbf{V}$ wird eine zufällige, aber anpassbare Vektorrepräsentation $\boldsymbol{e}^{(\mathbf{c})} \in \mathbb{R}^d$ zugeordnet. Im ersten Schritt werden die Zeichen der Eingabe mit $\boldsymbol{e}^{(\mathbf{c})} = \boldsymbol{W}_{i_\mathbf{c}}$ in ihre Repräsentationen überführt, wobei $i_\mathbf{c}$ der Index des Zeichens $\mathbf{c}$ in $\mathbf{V}$ ist. Anschließend werden mit diesen Eingaben die Ausgaben des Vorwärts- ($\boldsymbol{F} \in \mathbb{R}^{(k \times h)}$) und Rückwärtsmodells ($\boldsymbol{B} \in \mathbb{R}^{(k \times h)}$) bestimmt, wobei h die Ausgabedimensionalität der LSTMs ist. Für die Vorhersage des nächsten Zeichens wird während des Trainings eine lineare Schicht $\boldsymbol{P} \in \mathbb{R}^{(h \times |\mathbf{V}|)}$ zusammen mit einer Softmax-Funktion auf die Ausgaben angewendet, wobei jedes Ausgabeneuron mit einem Zeichen des Vokabulars korrespondiert. Die Parameter des Modells werden auf Grundlage der Kreuzentropie optimiert.

Nach dem Training werden die Worteinbettungen mit den BLSTM-Schichten des trainierten Sprachmodells bestimmt. Dazu werden die Wörter der Eingabe zunächst anhand der Leerzeichen separiert, wobei $t_i \in \{0, \ldots, k-1\}$ im Folgenden als die Position des ersten Zeichens des i-ten Wortes definiert wird. Die Repräsentation eines Wortes ergibt sich durch die Konkatenation der korrespondierenden Zustände vom Vorwärts- und Rückwärtsmodell. Für das Vorwärtsmodell wird dafür der verborgenen Zustand nach dem letzten Zeichen des Wortes und für das Rückwärtsmodell der Zustand vor dem ersten Zeichen des Wortes extrahiert. Formal wird die kontextuelle Repräsentation $\boldsymbol{w}^{(i)} \in \mathbb{R}^{2 \cdot h}$ des i-ten Wortes mit

$$\boldsymbol{w}^{(i)} = \boldsymbol{F}_{t_{i+1}-1} \mid\mid \boldsymbol{B}_{t_i-1} \tag{5.8}$$

bestimmt. Im Rahmen dieser Arbeit wird ein auf dem *One-Billion-Word-Corpus* vortrainiertes Sprachmodell[5] verwendet. Das Flair-Modell hat 18.3 Millionen Parameter und erzeugt Wortrepräsentationen mit einer Dimensionalität von 4096.

BERT

Die Berücksichtigung von Wortrepräsentationen aus Sprachmodellen, die auf großen Textkorpora vortrainiert wurden und auf einer Transformer-Architektur basieren, führen auf den meisten NLP-Benchmarks zu state-of-the-art Ergebnissen [42, 155]. Diese sogenannten Transformer-Encoder werden mit mehreren selbstüberwachten Aufgaben vortrainiert und können so kontextsensitive semantische Informationen auf Wort- und Satzebene kodieren. Eines der bekanntesten und meistgenutzten Encoder im NLP-Bereich ist das *Bidirectional Encoder Representations from Transformers* (BERT)-Modell [42]. Dieser Ansatz basiert auf einer bidirektionalen Transformer-Architektur. Die Architektur besteht aus stapelweise angeordneten Encoder-Blöcken (siehe Abschnitt 2.6), die wiederum weitgehend aus vollvernetzten Schichten und Attention-Heads zusammengesetzt sind. In der Praxis hat sich die Verwendung von 12 und 24 Encoder-Blöcken etabliert.

5 Flair-Modell: https://flair.informatik.hu-berlin.de/resources/embeddings/flair

Das BERT-Modell basiert auf einem festen Vokabular $\mathbf{V}$ von sogenannten Tokens, die neben ganzen Wörtern auch Teilwörter und Zeichen repräsentieren. Im ersten Schritt wird jedes Eingabewort mit einem Tokenizer in die kleinstmögliche Folge von Tokens bezüglich eines gegebenen Vokabulars zerlegt. Daraus resultiert für die Eingabe $\{\mathbf{w}^{(0)}, \ldots, \mathbf{w}^{(k-1)}\}$ eine Sequenz von Tokens $\{\mathbf{t}^{(0)}, \ldots, \mathbf{t}^{(l-1)}\}$, mit $l \geq k$ und $\mathbf{t}^{(i)} \in \mathbf{V}$. Für die Verarbeitung der Token mit dem Transformer muss die Eingabe zunächst in eine vektorielle Repräsentation $\boldsymbol{E} \in \mathbb{R}^{(l \times d_i)}$ überführt werden. Dabei existiert für jedes Token aus dem Vokabular eine lernbare Token-Repräsentation mit einer frei wählbaren Dimensionalität d_i. Da Transformer keine sequentielle Struktur wie RNNs aufweisen, ist eine zusätzliche Informationen zur Reihenfolge der Eingaben erforderlich. Hierbei hat sich der Einsatz von trigonometrischen Positionseinbettungen $\boldsymbol{P} \in \mathbb{R}^{(l \times d_i)}$ (siehe Formel 4.18) als besonders effektiv erwiesen [42]. Aufgrund der Verarbeitung von Eingabeformaten mit mehreren getrennten Texten für einige NLP-Aufgaben, wie z.B. beim QA mit Frage und Kontext, existieren zudem sogenannte Segmenteinbettungen $\boldsymbol{S} \in \mathbb{R}^{(l \times d_i)}$, welche die verschiedenen Eingabebereiche repräsentieren und damit unterscheidbar machen. Die finale Repräsentation der Eingabe $\hat{\boldsymbol{E}} \in \mathbb{R}^{(l \times d_i)}$ wird durch die elementweise Addition der Positions-, Token- und Segmenteinbettungen mit

$$\hat{\boldsymbol{E}} = \boldsymbol{E} + \boldsymbol{P} + \boldsymbol{S} \tag{5.9}$$

berechnet. Auf Grundlage der Eingabe bestimmt das Encoder-Modell die Ausgabe $\boldsymbol{O} \in \mathbb{R}^{(l \times d_o)}$. Je nach der eingesetzten Transformer-Architektur variiert die Ausgabegröße pro Token, d_o, und weist typischerweise eine Dimensionalität von 768 oder 1024 auf.

Für das selbstüberwachte Vortraining werden die Verfahren der maskierten Sprachmodellierung (MLM) und der Vorhersage des nächsten Satzes (engl.: *Next Sentence Prediction* (NSP)) eingesetzt. Das Ziel der MLM ist die Vorhersage eines zufällig maskierten Tokens im gegebenen Kontext, wobei das Modell sowohl den linken als auch den rechten Kontext des Tokens erfassen kann. In der Praxis erfolgt die Maskierung der Token heuristisch, indem 15% der Token zufällig maskiert werden. Das zu maskierende Token wird in 80% der Fälle durch das spezielle Token [MASK] und in den restlichen Fällen durch ein zufälligen Token aus $\mathbf{V}$ ersetzt. Ein MLP mit einer Ausgabegröße von $|\mathbf{V}|$ wird jeweils auf die Ausgaben der [MASK]-Token angewendet und auf die Vorhersage des ursprünglichen Tokens trainiert. Die NSP-Aufgabe ermöglicht das Erlernen von Beziehungen zwischen Sätzen, indem das System darauf trainiert wird, für einen gegebenen Satz A zu entscheiden, ob der Satz B ein Folgesatz ist und die Sätze somit semantisch zusammengehören. Für dieses Verfahren existiert ein spezielles Token [SEP], das als Trennzeichen zwischen den beiden Sätzen A und B fungiert. Zusätzlich wird das Klassifikations-Token [CLS] als aggregierte Repräsentation der Eingabe am Anfang jeder Eingabesequenz verwendet. Die Klassifikation der Zusammengehörigkeit der beiden Sätze erfolgt durch Anwendung eines MLPs auf die Ausgaberepräsentation des [CLS]-Tokens.

Für die Extraktion einer Wortrepräsentation aus dem Modell stehen mehrere Ansätze zur Verfügung, wobei die letzten Schichten des Modells meist die relevantesten Eigenschaften für NLP-Aufgaben beinhalten [42]. Die Zerlegung der Eingabewörter in mehrere Teilwörter zu Beginn der Architektur erfordert eine Strategie zur Extraktion einer Wortdarstellung für jedes Wort. Wird ein Eingabewort durch den Tokenizer in eine Folge von Teilwörtern zerlegt, so dient in dieser Arbeit die Wortdarstellung des ersten Teilwortes als Gesamtwortdarstellung.

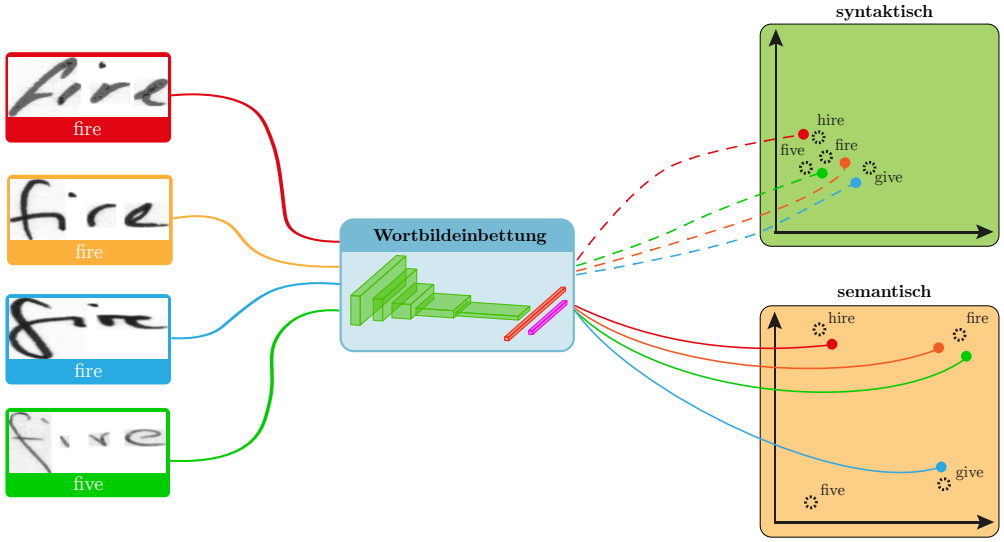


Abbildung 5.4: Die Herausforderungen bei der Abbildung von Wortbildern in einen semantischen Vektorraum und der Vergleich mit einer syntaktischen Abbildung. Obwohl die ersten drei Wortbilder in diesem Beispiel die Transkription „*fire*“ haben, werden die meisten dieser Bilder aufgrund ihrer visuellen Erscheinung anderen Wörtern im semantischen Raum zugeordnet. In der syntaktischen Repräsentation hingegen liegen alle Wortbilder des Beispiels nah an der Zielrepräsentation.

In dieser Arbeit wird ein vortrainiertes BERT-Modell[6] verwendet, das mit den MLM- und NSP-Aufgaben auf dem *Books-Corpus* [253] sowie extrahierten Textpassagen aus der englischen Wikipedia selbstüberwacht trainiert wurde. Das Modell umfasst 24 Encoder-Schichten, wobei die verborgenen Schichten eine Dimensionalität von 1024 aufweisen und die Anzahl der Attention-Heads auf 16 festgelegt ist. Insgesamt besteht das Modell aus 336 Millionen Parametern und erzeugt Worteinbettungen mit einer Dimensionalität von 1024.

5.2 ANNOTATIONSFREIE WISSENSDESTILLATION

Eine der größten Herausforderungen für die cross-modale Wissensdestillation in dieser Arbeit besteht darin, dass Wortbilder, deren Transkriptionen eine geringe Editierdistanz aufweisen, häufig auf stark variierende Repräsentationen abgebildet werden müssen. Diese Herausforderung wird in Abbildung 5.4 veranschaulicht, wobei visuell ähnliche Wortbilder im semantischen Worteinbettungsraum weit voneinander entfernt sind, während sie im syntaktischen Einbettungsraum nahe beieinander liegen. Dies liegt vor allem daran, dass die semantische Repräsentation eines Wortes in der Regel nicht mit der visuellen Erscheinung des Wortes korrespondiert. Diese fehlende Beziehung verhindert weitgehend die Vorhersage einer geeigneten semantischen Repräsentation für ein unbekanntes Wort auf der Grundlage seiner orthographischen Merkmale. Die Bestimmung semantischer Repräsentationen für Wortbilder, deren Annotationen nicht in der Trainingsmenge enthalten sind, ist daher schwierig bis unmöglich [214]. Deshalb ist es wichtig, möglichst viele Wörter in das Training des cross-modalen Destillationsansatzes einzubeziehen.

6 BERT-Modell:https://huggingface.co/bert-large-cased

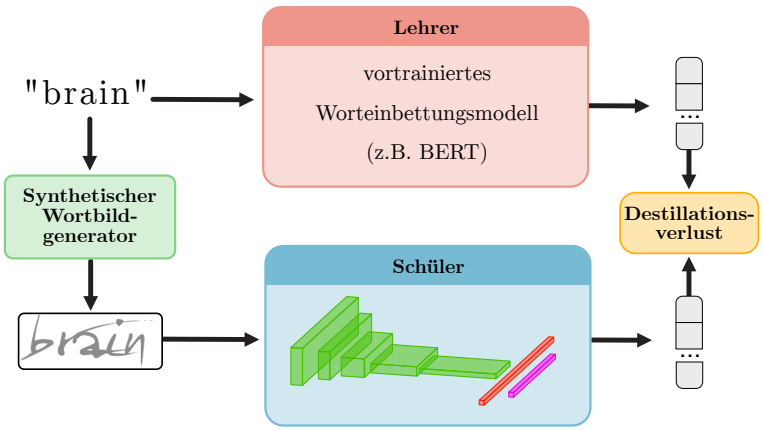


Abbildung 5.5: Ein Überblick des annotationsfreien Ansatzes zur cross-modalen Wissensdestillation. Synthetisch erzeugte Wortbilder werden für die Destillation verwendet, um den Mangel an manuell annotierten Wortbildern zu kompensieren.

Ein zentrales Problem im Bereich der handschriftlichen Dokumentenbilder ist jedoch die Verfügbarkeit großer Datensätze mit manuell annotierten Wortbildern. Eine mögliche Lösung stellt die Integration von synthetisch generierten Wortbildern dar. Synthetische Daten werden in der Dokumentenanalyse vor allem im Bereich der annotationsfreien Schlüsselwortsuche bzw. Texterkennung [94, 236, 237] und beim Vortraining von neuronalen Modellen [94, 114] eingesetzt. Dabei kann das Training mit synthetischen Wortbildern die Generalisierungsfähigkeit der Modelle verbessern [94] und den Bedarf an realen annotierten Daten verringern oder sogar erübrigen [236, 237]. Um diese Forschungsergebnisse auf den vorgestellten Ansatz zu übertragen, wird ein annotationsfreies Verfahren zur cross-modalen Wissensdestillation vorgestellt (siehe Abbildung 5.5). Dieser Ansatz basiert nicht auf einem vorgegebenen Korpus von manuell annotierten Wortbildern, sondern ausschließlich auf synthetisch generierten Wortbildern. Das Syntheseverfahren ermöglicht eine effiziente Generierung großer Datenmengen mit annotierten handschriftlichen Wortbildern und reduziert somit die Anzahl der nicht im Training aufgetretenen Wörter. Die grundsätzliche Intention dieses Verfahrens ist, dass mit einer ausreichend großen Menge an synthetischen Wortbildern der semantische Raum adäquat abgedeckt wird und somit nur wenige Wortbildrepräsentationen falsch vorhergesagt werden. Die geringe Anzahl fehlerhafter Vorhersagen kann von einem HTR-freien Modell intern verarbeitet und anhand der Kontextinformationen korrigiert werden.

Der Hauptunterschied zwischen dem annotationsfreien cross-modalen Ansatz zur Wissensdestillation und dem vorgestellten traditionellen Ansatz liegt in der Eingabephase. Die übrige Vorgehensweise ist analog zum traditionellen Ansatz. Dabei benötigt das annotationsfreie Modell lediglich ein maschinenlesbares Wort als Eingabe, während das traditionelle Modell ein manuell annotiertes Wortbild voraussetzt. Somit wird für die Wissensdestillation kein Datensatz mit annotierten Wortbildern sondern lediglich ein Vokabular von maschinenlesbaren Wörtern benötigt. Das zur Generierung verwendete Vokabular ist für die Qualität der Ergebnisse von entscheidender Bedeutung [235]. In dieser Arbeit wird ein Lexikon aus den am häufigsten vorkommenden Wörtern der englischen Sprache verwendet, da das Auftreten dieser Wörter in zukünftigen Dokumenten besonders wahrscheinlich ist. Zudem wird ein Bildgenerator benötigt, welcher möglichst realistische handschriftliche Wortbilder erzeugt. Eine kostengünstige und effektive Generierung von annotierten synthe-

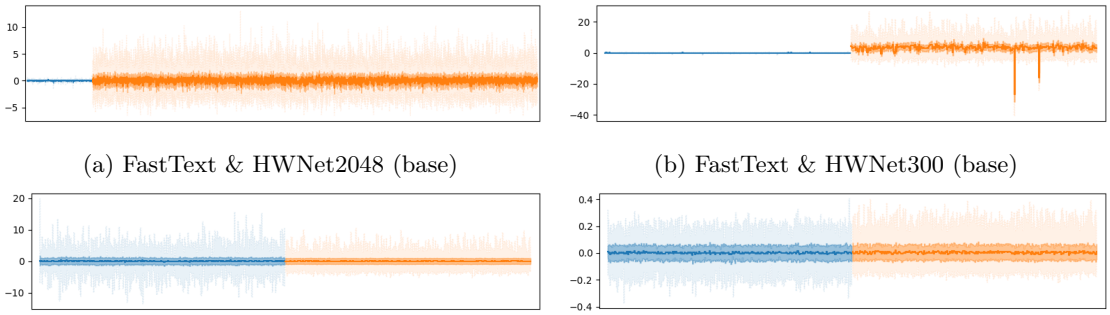
tischen Wortbildern ist mit computerbasierten Schriftarten, sogenannten *TrueType-Fonts*, realisierbar [96]. Diese sind in großer Zahl online verfügbar, sodass eine hohe Vielfalt an Schreibstilen simuliert werden kann. Eine Annäherung an reale handschriftliche Wortbilder wird durch eine Einführung zusätzlicher Variabilität in den Generierungsprozess erreicht [235]. Wichtige Parameter sind die Schriftgröße und -intensität, die Verzerrung mit affinen Transformationen und die Integration von Artefakten durch Filteroperationen.

5.3 KOMBINATION VON SEMANTISCHEN UND SYNTAKTISCHEN WORTEINBETTUNGEN

Die Verwendung von synthetisch generierten Wortbildern kann die Robustheit der semantischen Wortbildeinbettung zwar verbessern, aber das generelle Problem der fehlerhaften Abbildung von Wortbildern in einen semantischen Raum nicht lösen. Eines der Hauptprobleme bei der fehlerhaften Vorhersage ist, dass die orthografischen Informationen des im Wortbild enthaltenen Wortes verloren gehen. Ein Ansatz zur Lösung dieses Problems ist die Kombination von semantischen und syntaktischen Repräsentationen [98]. Durch deren Kombination ist auch bei einer fehlerhaften semantischen Vorhersage die Struktur des Wortes verfügbar und kann von einem nachfolgenden NLP-Modell berücksichtigt werden. Die zugrundeliegende Motivation basiert auf der robusten Vorhersage von syntaktischen Repräsentationen für Wortbilder (siehe Abbildung 5.4). Anhand des Beispiels wird verdeutlicht, dass visuell ähnliche Wortbilder eine ähnliche syntaktische Wortrepräsentation aufweisen und daher auch bei einer fehlerhaften Abbildung im syntaktischen Worteinbettungsraum nahe an der Zielrepräsentation liegen.

Eine intuitive Lösung für die Kombination stellt die Konkatenation einer semantischen und einer syntaktischen Wortrepräsentation dar [98]. Hierbei wird ein gegebenes Wortbild zunächst mit zwei separaten neuronalen Modellen in eine semantische und syntaktische Wortbildrepräsentation überführt. Anschließend werden die beiden Repräsentationen konkateniert und ergeben die neue Wortbildrepräsentation. Die simple Konkatenation der Repräsentationen ist jedoch aufgrund der teilweise sehr unterschiedlichen Charakteristika der Darstellungen, wie z.B. der Dimensionalitäten und Statistiken (Mittelwerte, Quantile und Varianzen), problematisch und führt häufig zu einer Dominanz bezüglich der semantischen oder syntaktischen Eigenschaften. Dies ist beispielhaft in Abbildung 5.6a dargestellt, wobei FastText als semantische und HWNetv2 als syntaktische Wortrepräsentation verwendet wird.

Wenn die konkatenierte Wortdarstellung lediglich als Eingabe für ein neuronales Netzwerk dient, werden die unterschiedlichen Charakteristika in der Regel keine wesentliche Auswirkung haben, da das Netzwerk diese beim Training ausgleichen bzw. berücksichtigen kann. Für Aufgaben wie die semantische Schlüsselwortsuche sind diese Abweichungen jedoch problematisch, da in den meisten Ansätzen lediglich die Ähnlichkeit zwischen zwei Repräsentationen ermittelt wird. Da eine der Einbettungen dominiert, werden semantische oder syntaktische Informationen häufig nicht in die Ähnlichkeitsbewertung einbezogen, was dazu führt, dass die Vorteile einer Kombination verloren gehen. Zur Lösung dieses Problems werden in dieser Arbeit vor der Konkatenation mehrere Normalisierungsschritte auf die Repräsentationen angewendet. Dadurch können Unterschiede in den statistischen Charakteristika der Darstellungen stark reduziert und Informationen gleichberechtigt berücksichtigt werden. In diesem Zusammenhang bietet das HWNetv2 ideale Voraussetzungen für die syntaktische Wortrepräsentation, da es bei nahezu allen Benchmarks im Bereich



(a) FastText & HWNet2048 (base)

(b) FastText & HWNet300 (base)

(c) FastText & HWNet300 (std.)

(d) FastText & HWNet300 (std. & l2)

Abbildung 5.6: Mittelwerte, Standardabweichungen und Dynamikbereiche der verschiedenen Einbettungsansätze für die Kombinationen von FastText (blau) und HWNet (orange). Auf die Einbettungen wurden vor deren Konkatenation eine Standardisierung (std.) und eine L2-Normalisierung (l2) angewendet. Die Einbettungen wurden auf der IAM-Testmenge berechnet.

der Schlüsselwortsuche state-of-the-art Ergebnisse erzielt und eine hohe Flexibilität hinsichtlich der Anpassung der Charakteristika für die Wortrepräsentationen aufweist. Dafür werden zunächst die Dimensionen der Repräsentationen angeglichen, indem die Anzahl der Ausgabeneuronen im HWNetv2-Modell von 2048 auf die Dimensionalität der semantischen Worteinbettung geändert wird (siehe Abbildung 5.6b). Zudem werden die Trainingsdaten mit der Z-Transformation standardisiert. Für jede Dimension der semantischen bzw. syntaktischen Repräsentationen werden zunächst der Mittelwert $\boldsymbol{\mu}$ und die Standardabweichung $\boldsymbol{\sigma}$ aus den Trainingsdaten berechnet. Anschließend wird für jede Repräsentation $\boldsymbol{r} \in \mathbb{R}^n$ aus den Trainings-, Validierungs- und Testdaten der Wert der i-ten Dimension mit

$$\boldsymbol{z}_i = \frac{\boldsymbol{r}_i - \boldsymbol{\mu}_i}{\boldsymbol{\sigma}_i} \tag{5.10}$$

standardisiert und ergibt so eine neue Repräsentation $\boldsymbol{z} \in \mathbb{R}^n$. Während die Trainingsdaten mit einem Mittelwert von 0 und einer Standardabweichung von 1 standardisiert sind, weisen die Validierungs- und Testdaten nicht notwendigerweise diese Eigenschaften auf (siehe Abbildung 5.6c).

Obwohl die Standardisierung bereits zu ähnlichen Statistiken bezüglich der Standardabweichung und des Mittelwertes für die beiden Wortdarstellungen führt, weisen die einzelnen Dimensionen von $\boldsymbol{z}$ weiterhin eine hohe Spannweite auf. Durch Normierung der Vektoren mit

$$\boldsymbol{z}_{norm} = \frac{\boldsymbol{z}}{\|\boldsymbol{z}\|_2} = \frac{\boldsymbol{z}}{\sqrt{\sum_{k=0}^{n-1} \boldsymbol{z}_k^2}} \tag{5.11}$$

auf eine L2-Norm von 1 werden die Daten auf eine Einheitslänge skaliert, wobei die Richtung des Vektors erhalten bleibt. Obwohl diese Skalierung keinen Einfluss auf die Kosinusähnlichkeit der einzelnen Einbettungen hat, ändert sich durch diese Operation der Vektoren vor der Konkatenation die Orientierung der gemeinsamen Darstellung. Auch wenn die L2-Norm keine theoretischen Intervallgrenzen garantieren kann, bietet sie in der Praxis oft eine Reduktion des Dynamikbereichs. Die statistischen Charakteristika der semantischen und syntaktischen Repräsentationen nach der Anwendung der Standardisierung und Normalisierung sind beispielhaft in Abbildung 5.6d dargestellt.

Neben der direkten Konkatenation wird in dieser Arbeit eine gewichtete Kombination von semantischen und syntaktischen Informationen vorgestellt. Dies ermöglicht es Benutzern, den Fokus der Darstellung während einer Ähnlichkeitsbewertung stärker auf semantische oder syntaktische Aspekte zu richten. Der Fokus wird durch einen manuell einstellbaren Gewichtungsfaktor $m \in [0, 1]$ bestimmt. Die gewichtete Kombination der syntaktischen und semantischen Einbettung, $\boldsymbol{y}$ bzw. $\boldsymbol{e}$, wird gemäß der Formel

$$\boldsymbol{c} = m \cdot \boldsymbol{e} \,||\, (1 - m) \cdot \boldsymbol{y} \tag{5.12}$$

berechnet. Dabei werden die syntaktische und semantische Einbettung elementweise mit $(1 - m)$ bzw. m multipliziert und anschließend konkateniert.

5.4 ANSÄTZE ZUR INTEGRATION VON SEMANTISCHEM VORWISSEN

In den vorherigen Abschnitten wurde ein optimiertes Verfahren zur Vorhersage von semantischen Repräsentationen für handgeschriebene Wortbilder vorgestellt. Im Folgenden werden Strategien zur Integration des semantischen Wortbildeinbettungsmodells in den HTR-freien Ansatz dieser Arbeit präsentiert.

Im NLP-Bereich haben sich die merkmalsbasierten Verfahren und die *fine-tuning*-Ansätze zur Integration von semantischen Vorwissen etabliert. Der merkmalsbasierte Ansatz verwendet aufgabenspezifische Architekturen, welche die semantischen Repräsentationen der Wörter aus einem gegebenen Dokument als Eingabe erhalten. Beim fine-tuning-Ansatz, wie z.B. BERT, werden hingegen aufgabenspezifische Parameter zu dem universalen Einbettungsmodell hinzugefügt und zusammen mit allen zuvor gelernten Parametern auf die NLP-Aufgabe angepasst. Der fine-tuning-Ansatz führt im Vergleich zum merkmalsbasierten Ansatz bei den meisten NLP-Benchmarks zu besseren Ergebnissen [42]. Dabei hat dieser jedoch den Nachteil, dass sich nicht alle Aufgaben problemlos durch eine simple Anpassung des Einbettungsmodells abbilden lassen, sodass häufig eine aufgabenspezifische Modellarchitektur erforderlich ist [42]. Zudem hat der merkmalsbasierte Ansatz fundamentale Ressourcenvorteile, da die meist großen Einbettungsmodelle nur einmal vortrainiert werden müssen und dann auf der Grundlage der Wortrepräsentationen anwendungsspezifische Modelle mit einer geringen Anzahl an Parametern effizient trainiert werden können [42].

In dieser Arbeit werden drei Ansätze zur Integration der vortrainierten Wortrepräsentationen vorgestellt und evaluiert. Dafür werden bei allen Ansätzen zunächst die Gewichte des Wortbildeinbettungsmodells aus dem HTR-freie Ansatz dieser Arbeit (siehe Abbildung 4.5) mit einem vortrainierten semantischen Modell initialisiert. Beim merkmalsbasierten Ansatz sind die Gewichte des Einbettungsmodells „eingefroren“, sodass nur die Parameter des anwendungsspezifischen NLP-Modells anpassbar sind. Somit dienen die vortrainierten Wortrepräsentation nur als Eingabe und werden nicht optimiert. Beim fine-tuning-Ansatz, sind auch die Gewichte des Einbettungsmodells anpassbar. Dadurch ermöglicht der Ansatz es eventuelle Fehler bei der Vorhersage der semantischen Einbettung zu korrigieren und die Unsicherheit bei der Abbildung zu berücksichtigen. Aufgrund dieser Eigenschaften ist der Ansatz im Allgemeinen für die Verarbeitung von handschriftlichen Dokumentenbildern zu bevorzugen.

Die Kombination aus einer meist geringen Anzahl von annotierten Trainingsdaten für die semantische Aufgabe und einer großen Anzahl von Modellparametern führt häufig zu einer Überanpassung der semantischen HTR-freien Modelle an die Trainingsdaten. Hierbei

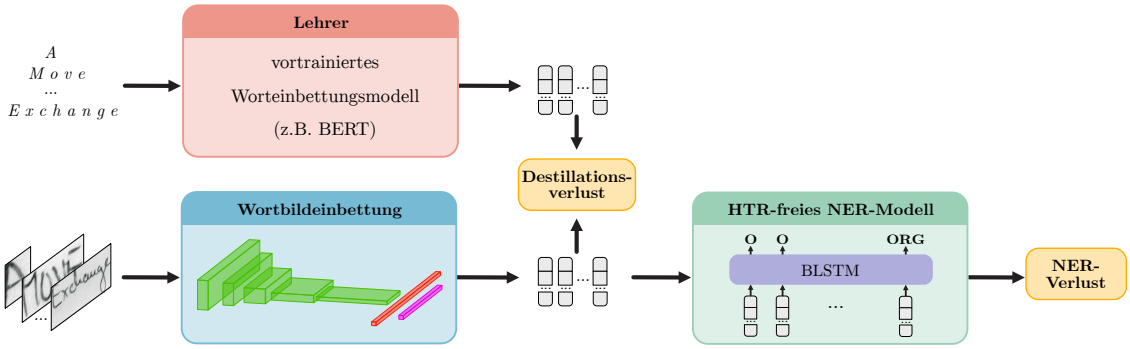


Abbildung 5.7: Ein Überblick über die duale Integrationsstrategie am Beispiel der NER-Aufgabe. Neben dem Training der NER-Aufgabe, wird der HTR-freie Ansatz auf die Wissensdestillation für die gegebenen handschriftlichen Wortbilder trainiert. Die Verlustfunktion der NER-Aufgabe und der Wissensdestillation werden gemeinsam optimiert.

sind insbesondere die hohe Anzahl an Parametern im ResNet problematisch. Um dieses Problem zu mildern, wird ein sogenannter dualer Ansatz vorgestellt, der den fine-tuning-Ansatz um die Optimierung des Destillationsverlustes zusätzlich zur Verlustfunktion für die NLP-Aufgabe erweitert. Der duale Ansatz ist in Abbildung 5.7 dargestellt und minimiert während des Trainings die Summe beider Kriterien. Die Motivation für diesen Ansatz besteht darin, eine Überanpassung der Parameter des Worteinbettungsmodells an die Trainingsdaten zu erschweren, indem eine semantische Repräsentation der Wortbilder bevorzugt wird.

6 EXPERIMENTELLE EVALUATION

Für die semantische Analyse von handschriftlichen Dokumentenbildern werden im Kapitel 4 sowohl ein HTR-freier als auch ein HTR-basierter Ansatz vorgestellt. Das HTR-freie Modell bietet theoretische Vorteile gegenüber dem HTR-basierten Ansatz, führt aber bei den meisten semantischen Benchmarks in der Literatur zu schlechteren Leistungen. Zur Lösung dieses Dilemmas wird in Kapitel 5 die cross-modale Wissensdestillation zur Integration externer semantischer Informationen in HTR-freie Modelle vorgestellt. Vor diesem Hintergrund ergeben sich die folgenden Forschungsfragen, die in dieser Arbeit beantwortet werden sollen:

1. Erfordert die semantische Analyse von handschriftlichen Dokumentenbildern spezielle Verfahren wie HTR-freie Architekturen oder ist eine einfache Kombination aus einem Handschrifterkenner und einem textuellen NLP-System ausreichend?
2. Kann die Integration von externem semantischen Wissen die Leistungsfähigkeit von HTR-freien Modellen verbessern und damit die bestehende Diskrepanz zu HTR-basierten Ansätzen beseitigen?
3. Existieren Kriterien, anhand derer entschieden werden kann, in welchen Fällen ein HTR-freier und in welchen ein HTR-basierter Ansatz besser geeignet ist?

Da es keinen theoretischen Ansatz zur Beantwortung der Forschungsfragen gibt, werden empirische Ansätze in Form von experimentellen Evaluationen verfolgt. Die Experimente werden anhand von drei repräsentativen semantischen Aufgaben durchgeführt, wobei für jede Aufgabe mehrere Benchmarks zur Verfügung stehen. Die Benchmark-Datensätze werden im Abschnitt 6.1 detailliert vorgestellt und erfordern unterschiedliche Metriken und Evaluationsprotokolle, die wiederum im Abschnitt 6.2 spezifiziert werden. Zur Beantwortung der ersten Forschungsfrage werden im Abschnitt 6.3 die Auswirkungen von Fehlern bei der Texterkennung auf die Leistungsfähigkeit von HTR-basierten Modellen evaluiert. In dieser Arbeit wird die Hypothese aufgestellt, dass ein wesentlicher Faktor für die geringe Leistungsfähigkeit von HTR-freien Modellen auf die fehlende Integration externer semantischer Informationen zurückzuführen ist. Zur Überprüfung dieser Hypothese wird zunächst im Abschnitt 6.4 die Relevanz von vortrainierten semantischen Worteinbettungen für NLP-Systeme untersucht. Anschließend werden im Abschnitt 6.5 anhand intrinsischer Evaluationen die wichtigsten Parameter des Verfahrens zur cross-modalen Wissensdestillation festgelegt. Dazu gehören die vorgestellten Optimierungsansätze zur robusten Vorhersage semantischer Wortbildrepräsentationen. Für das optimierte HTR-freie Modell und den HTR-basierten Ansatz werden im Abschnitt 6.6 die Leistungen auf den semantischen Benchmarks vorgestellt und mit verwandten Ansätzen aus der Literatur verglichen. Abschließend werden im Abschnitt 6.7 weiterführende Experimente durchgeführt, wobei insbesondere die Verwendung von HTR-freien und HTR-basierten Modellen in Abhängigkeit von Texterkennungsfehlern diskutiert wird.

Abbildung 6.1: Repräsentative Beispiele von Wortbildern aus den in dieser Arbeit verwendeten Evaluationsdatensätzen.

6.1 DATENSÄTZE

Die semantische Analyse von handschriftlichen Dokumentenbildern ist ein relativ junges Forschungsgebiet, sodass gegenwärtig nur wenige öffentlich verfügbare Benchmark-Datensätze in diesem Bereich zur Verfügung stehen. Bisherige publizierte Modelle aus der Literatur wurden häufig auf nicht öffentlich zugänglichen [22, 202] oder synthetisch generierten [22] Datensätzen trainiert und evaluiert. Dies erschwert den Vergleich und die Entwicklung semantischer Analyseverfahren für reale handschriftliche Dokumentenbilder. Die in dieser Arbeit für die Evaluation ausgewählten Datensätze enthalten sowohl synthetisch generierte als auch echte handgeschriebene Wortbilder. Dabei werden die Ansätze in dieser Arbeit sowohl auf Datensätzen mit modernen als auch historischen handschriftlichen Dokumentenbildern evaluiert. Dies führt zu einer starken Variation zwischen den Wortbildern der verschiedenen Benchmarks (siehe Abbildung 6.1). Die verwendeten Datensätze weisen zudem erhebliche Unterschiede in der Größe des verfügbaren Trainings- und Testmaterials, sowie in der Anzahl der Schreiber auf und ermöglichen somit Rückschlüsse auf eine Vielzahl realer Anwendungsszenarien. Für alle Datensätze liegen *Bounding-Box*-Informationen auf Wortebene vor. Die verfügbaren Datensätze werden zusätzlich zu ihrer jeweiligen NER- bzw. QA-Spezialisierung hinsichtlich der Handschrifterkennung sowie der Schlüsselwortsuche evaluiert.

IAM-Database

Die IAM-DB[1] [131] ist ein zentraler Benchmark für eine Vielzahl von Anwendungen im handschriftlichen Dokumentenbereich und wird in dieser Arbeit sowohl zur Bewertung von Methoden zur Schlüsselwortsuche als auch zur NER verwendet. Die Datenbank basiert auf dem Lancaster-Oslo-Bergen-Korpus [188] und beinhaltet moderne englische Texte aus stark unterschiedlichen Genres. Diese Texte wurden von insgesamt 657 verschiedenen Per-

1 IAM-DB Version 3.0: http://www.fki.inf.unibe.ch/databases/iam-handwriting-database

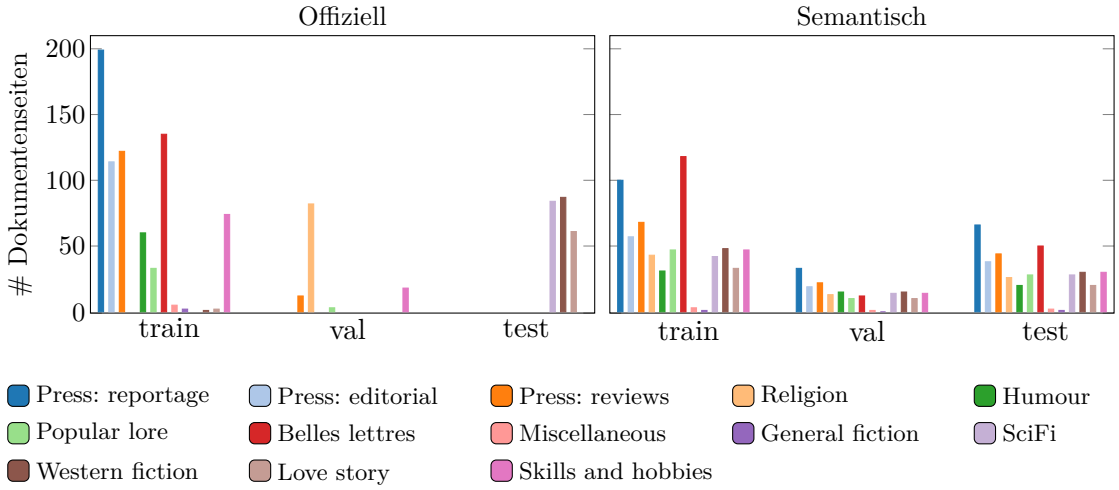


Abbildung 6.2: Die Anzahl der Dokumentseiten pro Genre in den Trainings-, Validierungs- und Testpartitionen. Die Histogramme werden für die offizielle und die angepasste Partitionierung der IAM-Datenbank angezeigt.

sonen in 1539 handschriftliche Dokumente überführt. Die eingescannten Dokumentenbilder umfassen insgesamt 13353 Textzeilen und 115320 Wortbilder. Die offizielle Partitionierung unterteilt die Datenbank in 6161 Zeilen für das Training, 1840 für die Validierung und 1861 für das Testen. Diese Partitionen sind schreiberunabhängig, sodass alle Dokumente eines Schreibers exklusiv zur Trainings-, Validierungs- oder Testmenge zugeordnet sind.

Für den IAM-Datensatz existieren manuelle NE-Annotationen sowie eine optimierte semantische Aufteilung[2] in Trainings-, Validierungs- und Testdaten [215]. Der Bedarf für eine optimierte Partitionierung der Daten wird anhand der Abbildung 6.2 ersichtlich. Die Histogramme verdeutlichen die suboptimale Verteilung der Dokumentengenres für die offizielle Aufteilung und die daraus resultierende mangelnde Repräsentativität der Trainingsdaten für die Test- und Validierungsdaten. Für eine optimierte Verteilung, wurden die Dokumente entsprechend ihrer Genres im Verhältnis $6:1:3$ in Trainings-, Validierungs- und Testmengen aufgeteilt, wobei eine schreiberunabhängige Partitionierung berücksichtigt wurde. Die NE-Annotationen der IAM-DB basieren auf den etablierten 18 Kategorien des *OntoNotes*-Datensatzes [203]. Es existieren zwei Versionen mit 18 und 6 Kategorien, wobei in der reduzierten Variante die 18 Kategorien (siehe Abbildung 6.3) so weit wie möglich zusammengefasst und stark unterrepräsentierte Kategorien ausgeschlossen wurden. Dies führt zu den folgenden sechs NE-Kategorien: Ort, Zeitangabe, Zahl, Person, Organisation (*ORG*) und *NORP* (Nationalität, religiöse oder politische Gruppen). Dabei umfasst die Kategorie „Ort“ die Elemente der „Geopolitischen Entitäten“ (*GPE*) und der „Gebäude“ (*FAC*). Die Kategorie „Zahlen“ beinhaltet die Entitäten „Ordinal“, „Prozent“, „Anzahl“ und „Geld“. In Abbildung 6.3 sind für die beiden Versionen die Anzahl der Wortbilder pro NE-Kategorie in Abhängigkeit der Partitionierung angegeben. Dies verdeutlicht die große Variabilität bezüglich des Vorkommens von Entitäten pro Kategorie, sowie die hohe Ähnlichkeit der Verteilungen zwischen den Partitionen.

2 Semantischer IAM-Split: https://patrec.cs.tu-dortmund.de/resources/

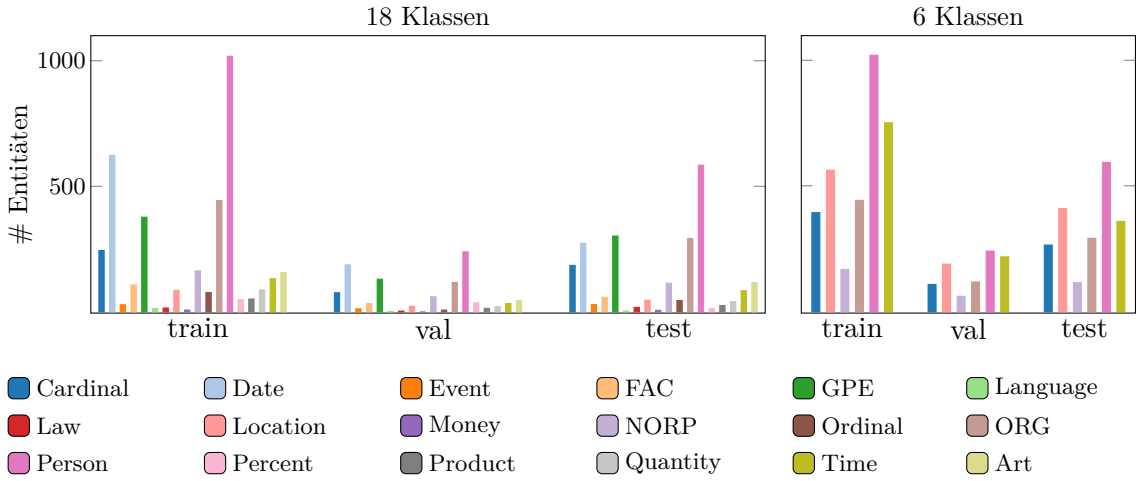


Abbildung 6.3: Die Anzahl der NEs in den Trainings-, Validierungs- und Testpartitionen für die IAM-Datenbank. Die Histogramme sind für die verschiedenen Größen der NE-Vokabulare dargestellt.

Synthetic Groningen Meaning Bank

Der *synthetic Groningen Meaning Bank* (sGMB)-Datensatz[3] [22] ist ein speziell für das NER entwickelter Benchmark und besteht aus synthetisch generierten handgeschriebenen Dokumentenseiten in englischer Sprache. Die synthetisierten Texte und deren Annotationen mit NEs basieren auf dem Korpus der *Groningen Meaning Bank* [20]. Obwohl es sich bei dem Benchmark um synthetische Dokumentenbilder handelt, bietet der Datensatz aufgrund der natürlichsprachlichen Texte und der geringen Anzahl von Entitäten realitätsnahe Voraussetzungen. Der maschinenlesbare Text wurde mit einem öffentlich verfügbaren Syntheseprogramm[4] in Dokumentenbilder überführt. Hierbei wurden die Dokumente mit circa 380 verschiedenen Schriftarten synthetisiert. Zur Nachbildung realistisch gescannter Dokumentenbilder wurden außerdem zufällige Verzerrungs- und Störeffekte auf die synthetisch erzeugten Bilder angewendet. Die offizielle Partitionierung des Datensatzes teilt die Daten in 38048 Trainings-, 5150 Validierungs- und 18183 Testwortbilder auf. Das NE-Vokabular besteht aus den folgenden fünf Kategorien: Geografische Entität, Organisation, Person, Geopolitische Entität und Zeitangabe.

George Washington

Der George Washington (GW)-Datensatz[5] [158] enthält 20 auf englisch verfasste Dokumentenseiten mit Korrespondenzen zwischen George Washington und seinen Mitarbeitern aus dem Jahr 1755. Es wird allgemein angenommen, dass die Dokumente von einer einzigen Person verfasst wurden. Der allgemeine Schreibstil und das Erscheinungsbild der Wortbilder sind daher weitgehend homogen. Es existieren insgesamt 4860 Wortbilder mit 1124 verschiedenen Transkriptionen, wobei diese nur in Kleinbuchstaben vorliegen und keine Satzzeichen enthalten. Für den GW-Datensatz existieren manuelle NE-Annotationen so-

3 sGMB Datensatz: https://github.com/omni-us/research-dataset-sGMB
4 HW-Synthesizer: https://github.com/manucarbonell/handwritten-document-synthesizer
5 George Washington Datensatz: http://ciir.cs.umass.edu/downloads/old/data_sets.html

wie eine optimierte Partitionierung der Dokumentenbilder[6] [215]. Dazu wurde eine optimale Verteilung der Dokumente mit der Antwortmengenprogrammierung (engl.: *Answer Set Programming*) [116] ermittelt, wobei die Dokumente in zwölf Trainings-, zwei Validierungs- und sechs Testseiten aufgeteilt wurden. Die Partitionierung der Dokumente ist dahingehend optimiert, dass die fünf NE-Kategorien, bestehend aus Zahl, Datum, Ort, Organisation und Person, im Verhältnis 6 : 1 : 3 verteilt sind.

HWSQuAD

Der QA-Datensatz HWSQuAD[7] [133] basiert auf dem textuellen *SQuAD1.0*-Datensatz [156] und besteht aus synthetisch generierten handgeschriebenen Dokumentenbildern. Der Textdatensatz ist ursprünglich für eine MRC-Aufgabe definiert worden und wurde von Mathew et al. in [133] zu einer QA-Aufgabe für Dokumentenkollektionen adaptiert. Der synthetische Datensatz besteht aus 20963 Dokumentseiten mit insgesamt 84942 Fragen. Bei der Synthese wurde eine Vielzahl von Parametern und Ansätzen verwendet, um möglichst realistische historische Dokumentenbilder zu erzeugen. Insgesamt enthält der Datensatz über 100 handschriftnahe Schriftarten und mehr als 20 manuskriptähnliche Hintergrundbilder. Die offizielle Partitionierung teilt den Datensatz in 17007 Dokumente für das Training, 1889 für die Validierung und 2067 für den Test auf. Die Trainings-, Validierungs- und Testmengen enthalten jeweils 67887, 7578 und 9477 Fragen. Der Datensatz verfügt sowohl über eine Zeilen- als auch Wortsegmentierung.

BenthamQA

BenthamQA[8] [133] ist ein historischer handschriftlicher QA-Datensatz, dessen Fragen und Antworten mittels *Crowdsourcing* erstellt wurden [221]. Der historische Datensatz enthält 338 Dokumente mit beträchtlichen Variationen im Schreibstil, die im wesentlichen vom englischen Philosophen Jeremy Bentham im 18. und 19. Jahrhundert verfasst wurden. Es kann davon ausgegangen werden, dass die Dokumente nicht ausschließlich von Bentham selbst, sondern auch von mehreren Sekretären verfasst wurden, wodurch eine erhöhte Variabilität in den Handschriften zu beobachten ist. Der Datensatz umfasst eine Testmenge, die aus lediglich 200 Frage-Antwort-Paaren bestehen und auf Grundlage von 94 der 338 Dokumentenbilder beantwortet werden können. Hierbei stehen sowohl eine Zeilen- als auch eine Wortsegmentierung zur Verfügung. Für das Training und die Validierung von QA-Modellen werden standardmäßig die Daten des HWSQuAD-Datensatzes verwendet. Bei der Erstellung von HTR- und WS-Modellen werden zudem die IAM- und GW-Datensätze aufgrund ihrer realen handschriftlichen Wortbilder berücksichtigt.

Synthetischer Datensatz

Die in dieser Arbeit erzeugten synthetischen Datensätze basieren auf den häufigsten Wörtern der englischen Sprache. Die verwendete Wortliste entstammt einer Worthäufigkeitsanalyse von über 70000 Büchern des Gutenberg-Projekts[9]. Die erzeugten Datensätze wei-

6 Semantischer GW Split: https://patrec.cs.tu-dortmund.de/resources/
7 HWSQuAD Datensatz: https://www.docvqa.org/datasets/benthamqa-and-hw-squad
8 BenthamQA Datensatz: https://www.docvqa.org/datasets/benthamqa-and-hw-squad
9 https://en.wiktionary.org/wiki/Wiktionary:Frequency_lists/English/Project_Gutenberg

sen eine unterschiedliche Anzahl von Wörtern auf, um den Einfluss der Vokabulargröße auf die Leistungsfähigkeit von Modellen zur semantischen Analyse von handgeschriebenen Dokumentenbildern zu evaluieren. Für jedes Wort des Vokabulars werden 50 Trainings- und 4 Testbilder erzeugt. Zur Generierung der synthetischen Wortbilder wird die *Python*-Bibliothek *Pillow*[10] verwendet. Diese erwartet zusätzlich zum Synthesewort eine computerbasierte Schriftart. Die in dieser Arbeit verwendeten 362 handschriftnahen Schriftarten stammen von der Plattform *1001Fonts*[11] und simulieren Handschriften mit unterschiedlichsten Schreibstilen. Die Schriften werden zunächst in disjunkte Trainings- und Testmengen aufgeteilt. Bei jedem Generierungsschritt wird zufällig eine Schrift aus der jeweiligen Menge ausgewählt. Zur besseren Annäherung an reale handschriftliche Wortbilder werden die Schriftgröße (in Pixel) und die Schriftintensität variiert [235]. Diese werden zufällig und gleichverteilt aus den Wertebereichen 40 bis 128 bzw. 0 (schwarz) bis 150 (hellgrau) entnommen. Zusätzlich wird die Hintergrundfarbe der synthetisierten Grauwertbilder in 50% der Fälle als weiß (255) und in den übrigen Fällen durch eine Normalverteilung mit einem Mittelwert von 211 und einer Standardabweichung von 24.7 definiert. Diese Parameter wurden aus der Verteilung der Trainingsdaten in der IAM-DB bestimmt. Kleine Segmentierungsfehler werden durch zufälliges Entfernen oder Auffüllen von Pixeln am Bildrand mit einer Gleichverteilung von -10 bis 20 modelliert. Es werden keine künstlichen Verzerrungen oder Fehler modelliert, da diese beim Training der Modelle durch Augmentierungsverfahren realisiert werden. Neben den Ansätzen zur Erhöhung der Variabilität auf Bildebene werden zusätzliche Anpassungen für das zu synthetisierende Wort vorgenommen. Dazu wird in 1% der Fälle das Wort vollständig in Großbuchstaben dargestellt und in 9% der Fälle der Anfangsbuchstabe des Wortes vertauscht, indem aus einem Kleinbuchstaben ein Großbuchstabe wird und umgekehrt. Außerdem wird in 1% der Fälle ein Sonderzeichen an das Wort angehängt.

6.2 EVALUIERUNGSMETRIKEN

Für die Vergleichbarkeit von Ansätzen ist es von großem Vorteil, wenn die Leistung eines Modells in einem numerischen Wert erfasst werden kann. Dazu sind neben den definierten Benchmarks auch geeignete Metriken zur Bewertung der Ansätze notwendig. Im Folgenden werden die in dieser Arbeit verwendeten und in der Literatur etablierten Metriken sowie Evaluierungsprotokolle für die verschiedenen Aufgabenstellungen dieser Arbeit vorgestellt.

Handschrifterkennung

Die Zeichenfehlerrate (engl.: *Character Error Rate* (CER)) und die Wortfehlerrate (engl.: *Word Error Rate* (WER)) sind die Standardmetriken zur Bewertung von Handschrifterkennungssystemen. Die beiden Metriken beschreiben die mittlere Fehlerrate bei der Erkennung auf Zeichen- bzw. Wortebene. Die Berechnung der Metriken erfolgt anhand einer Liste von Wortbildern, für die sowohl die HTR-Ausgaben als auch die zugehörigen Gold-Standard-Textannotationen verfügbar sind. Für die Bestimmung der CER wird zunächst für jedes Wortbild die Levenshtein-Distanz [111] berechnet. Hierbei wird die minimale Anzahl der erforderlichen Operationen zur Transformation der HTR-Ausgabe in die Annotation bestimmt, wobei die Zeichen entweder ersetzt, gelöscht oder einfügt werden können. Der

10 https://pillow.readthedocs.io/en/stable/reference/index.html
11 https://www.1001fonts.com/handwritten-fonts.html

Distanzwert wird anschließend mit der Wortlänge der Annotation, n_c, normiert. Dadurch wird die CER zu einem Vergleichsmaß, das unabhängig von der Länge des Textes ist. Die CER ist für ein einzelnes Wortbild formal mit

$$\mathrm{cer} = \frac{s_c + d_c + i_c}{n_c} \tag{6.1}$$

definiert, wobei s_c, d_c und i_c die Anzahl der Ersetzungs-, Löschungs- bzw. Einfügeoperationen darstellen. Das endgültige Bewertungsergebnis ist der Mittelwert aller CER-Werte für eine gegebene Wortbildmenge.

Im Allgemeinen erfolgt die Berechnung der WER analog zur CER. Aufgrund der Verarbeitung von vorsegmentierten Wortbilder wird in dieser Arbeit jedoch ein Spezialfall der WER betrachtet. Dabei entspricht die Anzahl der Wörter in der HTR-Ausgabe der Anzahl der Annotationen, sodass das Einfügen und Löschen von Wörtern entfällt. Lediglich das Ersetzen von Wörtern ist relevant, sofern die HTR-Ausgabe von der zugehörigen Annotation abweicht. Die Berechnung der WER erfolgt gemäß der Formel

$$\mathrm{wer} = \frac{s_w}{n_w}, \tag{6.2}$$

wobei s_w der Anzahl der fehlerhaft transkribierten Wörter und n_w der Gesamtzahl der Wortbilder in der Testmenge entspricht.

Semantische Schlüsselwortsuche

Für eine Bewertung von Methoden zur semantischen Schlüsselwortsuche ist sowohl eine semantische als auch eine syntaktische Metrik erforderlich. Hierbei hat sich die traditionelle Schlüsselwortsuche in Kombination mit der *mean Average Precision* (mAP) [130] als syntaktisches und die Wortanalogie (engl.: *Word Analogy* (WA)) [137] als semantisches Qualitätsmaß bewährt.

MEAN AVERAGE PRECISION

Das generelle Ziel der segmentierungsbasierten Schlüsselwortsuche ist die Transformation einer Menge von Wortbildern $\mathbf{D}$ in eine Liste $\mathbf{R}$, in der alle n Wortbilder aus $\mathbf{D}$ nach ihrer Relevanz bezüglich einer gegebenen Suchanfrage sortiert sind. Für die Schlüsselwortsuche stehen verschiedene Anfrageformate zur Verfügung, von denen in dieser Arbeit die Formate QbE und QbS verwendet werden. In der generierten Ausgabeliste sollten die relevanten Elemente möglichst am Anfang und die irrelevanten Elemente am Ende der Liste platziert werden. In diesem Zusammenhang wird ein Wortbild als relevant definiert, wenn dessen Annotation exakt mit der textuellen Repräsentation der Anfrage übereinstimmt. Formal wird die Relevanz durch die Funktion $\mathrm{r} : \mathbf{D} \to \{0, 1\}$ definiert, wobei ein relevantes Wortbild durch 1 und ein irrelevantes Wortbild durch 0 repräsentiert wird.

Für die Bewertung der Rückgabeliste wird erwartet, dass eine perfekte Anordnung der Wortbilder zu einem Wert von 1 führt. Es wird außerdem erwartet, dass dieser Wert abnimmt, wenn irrelevante Wortbilder in der Rückgabeliste vor den relevanten Elementen platziert sind. Dieser Zusammenhang lässt sich mit der *Average Precision* (AP) modellieren, welche formal durch

$$\mathrm{ap} = \frac{\sum_{k=0}^{n-1} \mathrm{p}(k) \cdot \mathrm{r}(k)}{\sum_{i=0}^{n-1} \mathrm{r}(i)} \tag{6.3}$$

definiert ist. Hierbei wird die *Precision* über verschiedene Teile der Rückgabeliste bestimmt und für die ersten k Wortbilder der Liste mit

$$\mathrm{p}(k) = \frac{tp}{tp + fp} = \frac{\sum_{i=0}^{k-1} \mathrm{r}(i)}{k} \tag{6.4}$$

berechnet. Diese Formel beschreibt das Verhältnis zwischen der Anzahl relevanter und irrelevanter Wortbilder in der Retrieval-Liste. Hierbei entspricht *True Positive* (tp) der Anzahl der Wortbilder mit identischer Annotation zur Anfrage und *False Positive* (fp) der Anzahl der Wortbilder mit abweichender Annotation. Da jedes Wortbild ausschließlich einer der Klassen tp und fp zugeordnet ist, entspricht die Summe dieser Werte der Länge der Retrieval-Liste. Abschließend werden die Werte auf das Intervall $[0, 1]$ normiert. Dazu wird der Normalisierungsfaktor mit $\sum_{i=1}^{n} \mathrm{r}(i)$ berechnet und entspricht der Anzahl der relevanten Wortbilder in $\mathbf{D}$. Das endgültige Bewertungsmaß für den Ansatz ist die mAP, bei der die AP-Werte für alle Anfragen des Benchmarks gemittelt werden.

Die Berechnung der mAP erfolgt in dieser Arbeit analog zum Protokoll aus [8]. In diesem Protokoll wird bei der QbE-Auswertung das erste Element der Rückgabeliste verworfen und ein Wortbild aus der Testmenge genau dann als QbE-Anfrage verwendet, wenn mindestens ein weiteres Wortbild mit der gleichen Annotation in der Testmenge existiert. Bei der QbS-Auswertung werden die Annotationen des Testdatensatzes genau einmal als Anfrage verwendet. Dementsprechend werden Wörter, die mehrfach im Datensatz vorkommen, nur einmal als Anfrage berücksichtigt. Das Protokoll hat zudem die Besonderheit für den IAM-Datensatz, dass sowohl beim QbE- als auch beim QbS-Verfahren nur Anfragen berücksichtigt werden, die nicht Bestandteil der offiziellen Stoppwortliste sind. Aufgrund der geringen Größe des GW-Datensatzes wird eine vierfache Kreuzvalidierung für diesen Benchmark durchgeführt und die Leistung als Mittelwert dieser Validierungen angegeben.

WORTANALOGIE

Zur Bewertung der semantischen Qualität von textuellen Worteinbettungsmodellen haben sich WA-Benchmarks bewährt. Bei diesem Evaluierungsansatz werden manuell vordefinierte Beispiele für semantische Analogien vorgegeben, die vom Modell gelöst werden müssen. Die Grundidee dieser Benchmarks besteht darin, zu überprüfen, ob die Wörter räumlich so im Vektorraum angeordnet sind, dass die Analogien mit einfacher Vektorarithmetik gelöst werden können. Formal sind in der WA-Aufgabe drei Wörter $\mathbf{a}, \mathbf{b}$ und $\mathbf{c}$ gegeben. Das Ziel besteht darin, das vierte Wort $\mathbf{d}$ zu ermitteln, sodass die folgende Bedingung erfüllt ist: $\mathbf{a}$ verhält sich zu $\mathbf{b}$ wie $\mathbf{c}$ zu $\mathbf{d}$. Ein Beispiel wäre „Deutschland“ verhält sich zu „Berlin“ wie „Frankreich“ zu „Paris“.

In dieser Arbeit wird die in [137] veröffentlichte Sammlung von manuell definierten WA-Beispielen verwendet. Zur Realisierung des Benchmarks auf handschriftlichen Wortbildern werden zunächst die Einbettungen für alle Wortbilder aus der Testmenge des gegebenen Datensatzes berechnet. Anschließend werden alle Analogien verworfen, bei denen das Zielwort $\mathbf{d}$ nicht mit mindestens einer Annotation aus der Testmenge übereinstimmt. Für alle übrigen Analogien werden die textuellen semantischen Repräsentationen der Wörter $\mathbf{a}, \mathbf{b}$ und $\mathbf{c}$ bestimmt und die erwartete Position $\hat{\boldsymbol{d}}$ des Lösungswortes mit

$$\hat{\boldsymbol{d}} = \boldsymbol{b} - \boldsymbol{a} + \boldsymbol{c} \tag{6.5}$$

ermittelt. Anschließend wird für die geschätzte Vektorrepräsentation $\hat{\boldsymbol{d}}$ und den Wortbildrepräsentationen aus der Testmenge das Wortbild mit der höchsten Kosinusähnlichkeit

bestimmt. Falls die Annotation des so ermittelten Wortbildes mit dem Zielwort **d** übereinstimmt, ist die Analogie erfüllt. Als abschließendes semantisches Bewertungsmaß wird die Genauigkeit (engl.: *Accuracy*) verwendet, die den Prozentsatz der richtig vorhergesagten Analogien angibt.

Named Entity Recognition

NER-Modelle werden in der Literatur standardmäßig mit den Metriken *Recall* und Precision bewertet. Dabei gibt die Precision den prozentualen Anteil der vom System korrekt erkannten Entitäten und der Recall den Anteil der vom System im Korpus gefundenen Entitäten an. Da es sich in dieser Arbeit um segmentierungsbasierte Ansätze handelt, existiert für jedes Wortbild des Benchmarks genau eine vorhergesagte Entität und eine Gold-Standard Annotation. Bei den NER-Datensätzen ist zu beachten, dass alle Labelmengen eine Rückweisungsklasse (O) besitzen. Diese Klasse wird jedem Wortbild zugewiesen, das nicht zu einer der vordefinierten Entitäten der Labelmenge gehört.

Für jede Entität **e** der Labelmenge wird die Precision (p) analog zur Formel 6.4 und der Recall (r) mit

$$r = \frac{tp}{tp + fn} \tag{6.6}$$

berechnet. Die Grundlage der Formeln bietet die sogenannte Konfusionsmatrix (engl.: *Confusion Matrix*), welche die Anzahl der richtig und falsch klassifizierten Vorhersagen für die Klasse **e** angibt. Von besonderem Interesse sind die Werte der False Positives (fp), der *False Negatives* (fn) und der True Positives (tp) aus dieser Matrix. Ein True Positive liegt vor, wenn die Annotation **e** für ein Wortbild korrekt vorhergesagt wurde. Ein False Positive tritt auf, wenn **e** vorhergesagt wurde, das Wortbild jedoch eine andere Annotation aufweist. Ein False Negative beschreibt ein Wortbild, das mit **e** annotiert ist, für das aber eine andere Entität vorhergesagt wurde. Bei der Berechnung der Metriken können zwei Sonderfälle auftreten. Zum einen kann es vorkommen, dass in der Testmenge kein Wortbild für eine gegebene Klasse vorhanden ist, diese Klasse aber für ein Wortbild vorhergesagt wurde. In diesem Fall ist die Berechnung des Recalls mathematisch nicht definiert und wird in dieser Arbeit als 0 gewertet. Zum anderen ist es möglich, dass für eine Klasse kein Wortbild vorhergesagt wurde, sodass die Berechnung der Precision mathematisch nicht definiert ist und daher in dieser Arbeit mit 0 bewertet wird.

Der F1-Wert ($f \in [0, 1]$) ist ein gewichteter Mittelwert aus der Precision und dem Recall und erzeugt auf diese Weise einen einzigen Wert zur Bewertung der Leistung von NER-Systemen. Formal wird der F1-Wert mit

$$f = \frac{2 \cdot p \cdot r}{p + r} \tag{6.7}$$

berechnet. In der Literatur existieren verschiedene Bewertungsprotokolle, von denen *macro* und *micro* die am häufigsten verwendeten sind. In dieser Arbeit wird der macro-Ansatz verwendet, bei dem die Precision-, Recall- und F1-Werte für jede Entitätsklasse separat berechnet und anschließend mit dem arithmetischen Mittelwert zu einem Wert zusammengefasst werden. Auf diese Weise werden alle Klassen gleichermaßen berücksichtigt und es wird vermieden, dass der Wert durch die Leistung einer häufig vorkommenden Entitätsklasse verzerrt wird. Dies ist besonders wichtig, da im Allgemeinen bei allen NER-Datensätzen eine deutliche Diskrepanz zwischen den NE-Klassen existiert, wobei etwa 90% der Wortbilder der Rückweisungsklasse zuzuordnen sind.

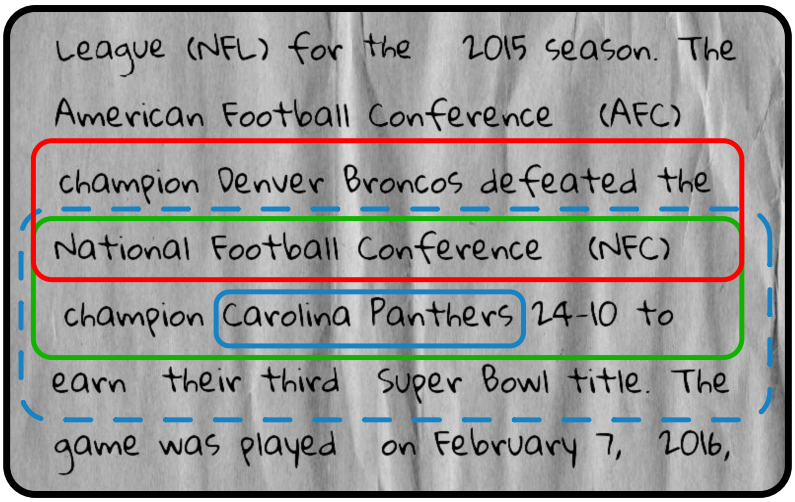


Abbildung 6.4: Ein Beispiel des HWSQuAD-Datensatzes für eine richtig (grün) und eine falsch (rot) vorhergesagte Antwortbox bzgl. der Frage „Welcher Verein verlor den Super Bowl 2016?“. Das Rechteck um das Wort „Carolina Panthers“ stellt die Box **K** dar, während das gestrichelte Rechteck die Box **G** illustriert.

Question Answering

Im Gegensatz zu QA-Modellen aus dem NLP-Bereich erzeugen die in dieser Arbeit vorgestellten Ansätze keine Antwort in einem maschinenlesbaren Format, sondern liefern einen Ausschnitt des Dokumentenbildes, der die Antwort enthält. Dazu wird analog zu [133] ein Bildbereich auf Zeilenebene aus einem Dokumentenbild extrahiert und die Leistung der QA-Systeme mit dem *Double Inclusion Score* (DIS) [133] gemessen. Dieses Verfahren ist an die *Intersection over Union* (IoU) angelehnt und bewertet für jede Frage die Übereinstimmung zwischen dem vorhergesagten Bildbereich des QA-Modells und dem Bildbereich der Gold-Standard Annotation. Ein wichtiges Kriterium der Metrik ist, dass sich der Bildausschnitt aus der Annotation innerhalb des vorhergesagten Bildbereichs befindet. Dieses Kriterium ist jedoch nicht ausreichend, da eine Methode, die das gesamte Bild als Antwort zurückgibt, bereits die Bedingung erfüllt, aber keinen Mehrwert für die Lokalisierung der Antwort im Dokument bietet. Daher wird ein weiteres Kriterium benötigt, um Antworten mit einem zu umfangreichen Kontext zu sanktionieren. Formal werden diese Kriterien bei der Berechnung des DIS für einen vorhergesagten und einen Gold-Standard Bildausschnitt mit

$$d = \frac{|\mathbf{A} \cap \mathbf{K}|}{|\mathbf{K}|} \cdot \frac{|\mathbf{A} \cap \mathbf{G}|}{|\mathbf{A}|} \tag{6.8}$$

umgesetzt. Dabei bestimmt der erste Term implizit, ob die Gold-Standard Antwort im vorhergesagten Bildbereich enthalten ist und der zweite Term sanktioniert Antworten mit zu großen Bildbereichen. Das Verfahren basiert auf der Wort- und Zeilensegmentierung eines gegeben Datensatzes und überführt die Bildausschnitte zunächst in Wortbildmengen. Hierbei repräsentiert die Box **K** die Menge der Wortbilder, welche die Gold-Standard Antwort darstellen. Zudem repräsentiert die Antwortbox **A** die Menge der Wortbilder, die Teil des vom QA-System vorhergesagten Bildausschnitts sind. Die Box **G** enthält alle Wortbilder aus den Zeilen der Box **K**, sowie die Wortbilder aus der Zeile darüber und der Zeile darunter. Ein visuelles Beispiel für die Definitionen der Boxen ist in Abbildung 6.4 dargestellt. Die Vorhersage wird als korrekte Antwort betrachtet, wenn $d \in [0, 1]$ über dem

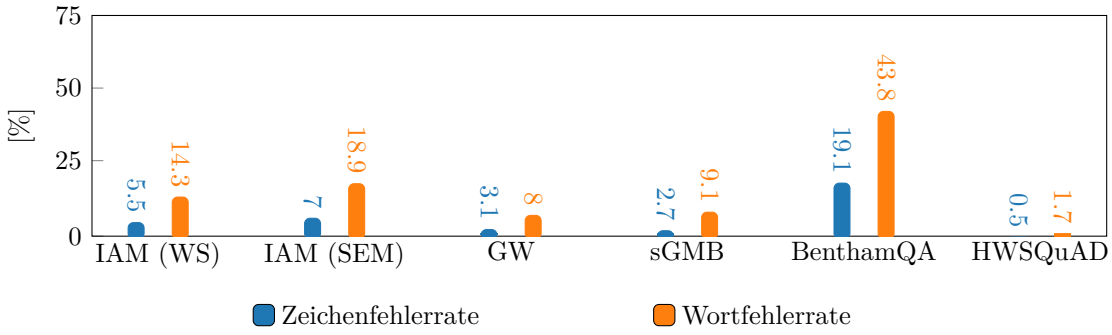


Abbildung 6.5: Die Leistungen der HTR-Modelle auf den Evaluationsbenchmarks. Die Ergebnisse sind in Zeichen- und Wortfehlerrate angegeben.

Schwellwert von 0.8 liegt. Die Leistung des QA-Modells ist definiert als der prozentuale Anteil der korrekt beantworteten Fragen.

6.3 AUSWIRKUNG VON TEXTERKENNUNGSFEHLERN AUF NLP-MODELLE

Die Hauptmotivation für die Verwendung von HTR-freien Ansätzen besteht in der Annahme, dass die Handschrifterkennung zu viele Transkriptionsfehler produziert und sich diese negativ auf nachfolgende textbasierte NLP-Modelle auswirken. In der Literatur existieren bereits eine Reihe von Publikationen, die den Einfluss von OCR-Fehlern auf NLP-Modelle untersuchen und einen negativen Einfluss nachgewiesen haben [19, 65, 66]. Auch wenn der allgemeine Trend wahrscheinlich auf die Handschrifterkennung übertragbar ist, ist eine direkte Ableitung hinsichtlich der Auswirkung von Fehlern aufgrund der höheren Variabilität von Handschriften nicht ohne weiteres möglich. Neben diesen Studien existieren Arbeiten in der Literatur, die einen zweistufigen OCR-basierten Ansatz zur NER erfolgreich auf maschinell gedruckten Dokumentenbildern anwenden. Hierbei wird nur ein marginaler Leistungsverlust festgestellt, wenn das NER-Modell auf die OCR-Ausgaben im Vergleich zu den textuellen Gold-Standard Annotationen angewendet wird [66]. Eine interessante Frage ist daher, ob und wie sich die Fehler von HTR-Modellen auf die Leistung von NLP-Systemen auswirken und ob eine der drei semantischen Aufgaben aus dieser Arbeit besonders robust oder anfällig für Texterkennungsfehler ist.

Zur Beantwortung dieser Forschungsfragen wird die Leistung des HTR-basierten Ansatzes in Abhängigkeit von der CER evaluiert. Dazu werden HTR-Modelle mit einer variablen Anzahl von Trainingsdaten ($1\% - 100\%$) trainiert und auf den NER-, QA- und WA-Benchmarks evaluiert. Die HTR-Modelle dienen als Texterkenner im HTR-basierten Ansatz und erzeugen Erkennungsergebnisse für die semantischen Benchmarks mit unterschiedlichen CERs. Tabelle 6.5 zeigt die Leistung der HTR-Modelle auf den in dieser Arbeit verwendeten Datensätzen unter Berücksichtigung aller Trainingsdaten. Die Transkriptionen dienen anschließend als Eingabe für die auf den Annotationen des Benchmarks vortrainierten NLP-Modelle aus Abschnitt 4.1.2. Zusätzlich zu den Ausgaben der HTR-Modelle wird die Leistung der NLP-Modelle hinsichtlich einer perfekten Texterkennung evaluiert. Dazu werden die Annotationen der Wortbilder als Eingabe für die NLP-Modelle genutzt. Aus Gründen der Übersichtlichkeit wird für jede semantische Aufgabe nur ein repräsentativer Benchmark evaluiert. Hierbei wird der BenthamQA-Datensatz für das QA, der IAM-NER Datensatz mit 6 NE-Kategorien für die NER und der IAM-Datensatz für die Wortanalogie verwendet.

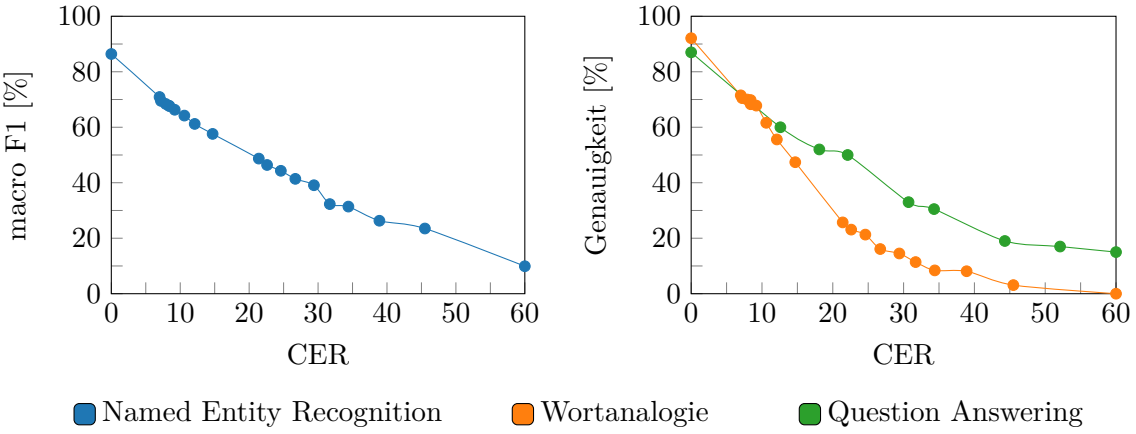


Abbildung 6.6: Der Einfluss von Fehlern bei der Handschrifterkennung auf die Leistungsfähigkeiten der drei NLP-Ansätze. Die QA-Ergebnisse wurden auf dem BenthamQA, die NER-Ergebnisse auf dem IAM-NER(6) und die WA-Ergebnisse auf der IAM-DB produziert.

Die Ergebnisse in Abbildung 6.6 verdeutlichen den starken negativen Einfluss von HTR-Fehlern auf NLP-Modelle. Bei einer fehlerfreien Texterkennung erzielen die semantischen Modelle auf den Benchmarks beeindruckende Ergebnisse. Dabei werden 92.1% der Wortanalogien richtig gelöst, 87% der Fragen richtig beantwortet und ein F1-Wert von 86.4% für die NER erreicht. Allerdings verschlechtern sich die Ergebnisse bei den NER- und WA-Aufgaben bereits bei einer geringen Fehlerrate deutlich. Der absolute Leistungsverlust bei Verwendung der besten Texterkennungsmodelle beträgt bereits etwa 15% für den NER Benchmark und etwa 20% für den WA-Benchmark. Die Leistung der Texterkennungsmodelle liegt für die beiden Benchmarks auf der IAM-DB bei einer CER von $7\% - 60\%$ und für den BenthamQA-Benchmark bei $15\% - 60\%$. Die verhältnismäßig geringe Leistung der HTR-Modelle auf dem BenthamQA-Datensatz ist hauptsächlich auf dem Fehlen repräsentativer Trainingsdaten zurückzuführen. Durch die Verwendung von Datensätzen mit echten Handschriften im Training, wie z.B. IAM und GW, kann die CER für die HTR-Modelle von ca. 30% auf 15% reduziert werden. Trotz der Verbesserung der Erkennungsergebnisse verschlechtert sich die Leistung des QA-Systems auf dem BenthamQA-Benchmark um circa 30% im Vergleich zur Verwendung von Gold-Standard Textannotationen als Eingabe.

Generell ist bei allen getesteten Benchmarks eine Verschlechterung der Leistung der semantischen Modelle mit steigender CER zu beobachten. Die Wortanalogie zeigt eine besonders hohe Anfälligkeit für Texterkennungsfehler und weist teilweise einen exponentiellen Leistungsabfall bei steigender CER auf. Dieser Zusammenhang ist jedoch wenig überraschend, da in der Literatur bereits die besonders hohe Anfälligkeit von semantischen Worteinbettungsmodellen für Rechtschreib- und OCR-Fehler nachgewiesen werden konnte [46, 101]. Ein möglicher Grund liegt in der separaten Verarbeitung der HTR-Ergebnisse im semantischen Modell, wodurch auftretende Erkennungsfehler nicht wie bei den QA- und NER-Modellen über den Kontext berücksichtigt und korrigiert werden können. Die NER- und QA-Modelle zeigen im Vergleich zum WA-Ansatz eine höhere Robustheit, weisen aber dennoch einen negativen linearen Zusammenhang zwischen der CER und der Leistung der semantischen Aufgabe auf. Bemerkenswert ist jedoch, dass trotz der hohen CERs von teilweise über 60% weiterhin ein F1-Wert von 9.9% auf dem NER-Benchmark erreicht wird und 15% der Fragen auf dem QA-Datensatz beantwortet werden können.

Zur formalen Bestimmung des Zusammenhangs zwischen der CER und den Leistungen der semantischen Modelle wird für jede der drei semantischen Aufgaben eine Korrelationsanalyse durchgeführt. Hierbei wird der Korrelationskoeffizient $r \in [-1, 1]$ bestimmt, der ein Maß für die Stärke des Zusammenhangs zwischen zwei Variablen $\mathbf{X} = \{\mathbf{x}^{(0)}, \ldots, \mathbf{x}^{(n-1)}\}$ und $\mathbf{Y} = \{\mathbf{y}^{(0)}, \ldots, \mathbf{y}^{(n-1)}\}$ darstellt. Im vorliegenden Anwendungsfall repräsentiert $\mathbf{X}$ die CERs und $\mathbf{Y}$ die Leistung der semantischen Modelle für jeweils eine der drei Aufgaben. Ist der Korrelationskoeffizient kleiner als Null, besteht ein negativer Zusammenhang zwischen den Variablen. Das bedeutet, dass sich $\mathbf{X}$ und $\mathbf{Y}$ in entgegengesetzter Richtung entwickeln und somit bei einer Erhöhung des Wertes der einen Variable der Wert der anderen Variable sinkt. Ist der Koeffizient größer als Null, liegt eine positive Korrelation vor. Das bedeutet, dass wenn der Wert der einen Variable steigt, auch der Wert der anderen steigt. Bei einem Koeffizienten von Null besteht kein linearer Zusammenhang zwischen $\mathbf{X}$ und $\mathbf{Y}$. Je näher der Korrelationskoeffizient bei 1 oder -1 liegt, desto stärker ist der Zusammenhang zwischen den Variablen. Der Korrelationskoeffizient wird formal mit

$$\mathrm{r}(\mathbf{X}, \mathbf{Y}) = \frac{\mathrm{Cov}(\mathbf{X}, \mathbf{Y})}{\mathrm{Var}(\mathbf{X}) \cdot \mathrm{Var}(\mathbf{Y})} \tag{6.9}$$

berechnet. Hierzu wird die Kovarianz von $\mathbf{X}$ und $\mathbf{Y}$ durch das Produkt der Standardabweichungen der beiden Variablen dividiert. Für die Auswertungen dieser Arbeit ergibt sich für die Leistung des QA-Modells und der entsprechenden CERs ein Korrelationskoeffizient von -0.96, für die NER ein Wert von -0.96 und für die WA ein Wert von -0.93. Somit weisen alle semantischen Aufgaben eine starke negative Korrelation zwischen der CER und der Leistung der semantischen Modelle auf.

Zusammenfassend deuten die Ergebnisse auf einen starken negativen Einfluss von HTR-Fehlern auf die Leistungsfähigkeit semantischer Ansätze hin. Daher kann bereits eine geringe Fehlerrate bei der einfachen Kombination eines HTR- und eines NLP-Modells zu erheblichen Leistungseinbußen bzgl. der semantischen Analyse von handschriftlichen Dokumentenbildern führen.

6.4 RELEVANZ VON VORTRAINIERTEN SEMANTISCHEN WORTEINBETTUNGEN

Zur Reduzierung der Auswirkungen von Handschrifterkennungsfehlern auf nachfolgende NLP-Modelle wurden in der Literatur bereits HTR-freie Ende-zu-Ende-Ansätze veröffentlicht [1, 167, 207]. Obwohl diese HTR-freien Modelle das Problem der Fehlerfortpflanzung lösen können, erzielen sie auf den meisten semantischen Benchmarks in der Literatur deutlich schlechtere Ergebnisse als die HTR-basierten Ansätze [133, 215]. In dieser Arbeit wird die Hypothese vertreten, dass dieser Leistungsunterschied hauptsächlich auf die fehlende Integration von vortrainierten semantischen Worteinbettungen in HTR-freien Modellen zurückzuführen ist. Diese semantischen Worteinbettungen werden auf der Grundlage großer Textkorpora vortrainiert und kodieren damit eine Art externes Weltwissen, das für ein Verständnis semantischer Zusammenhänge von grundlegender Bedeutung ist. Diese externen semantischen Informationen können nicht aus den wenigen Trainingsdaten der Benchmarks gelernt werden. Dies führt vermutlich zu einer geringeren Generalisierungsfähigkeit der HTR-freien Modelle und damit zu einer schlechteren Leistung im Vergleich zu HTR-basierten Ansätzen. Um diese Hypothese zu überprüfen, wird in diesem Experiment zunächst die Relevanz von vortrainierten semantischen Worteinbettungen für die semantische Analyse von handschriftlichen Dokumentenbildern untersucht.

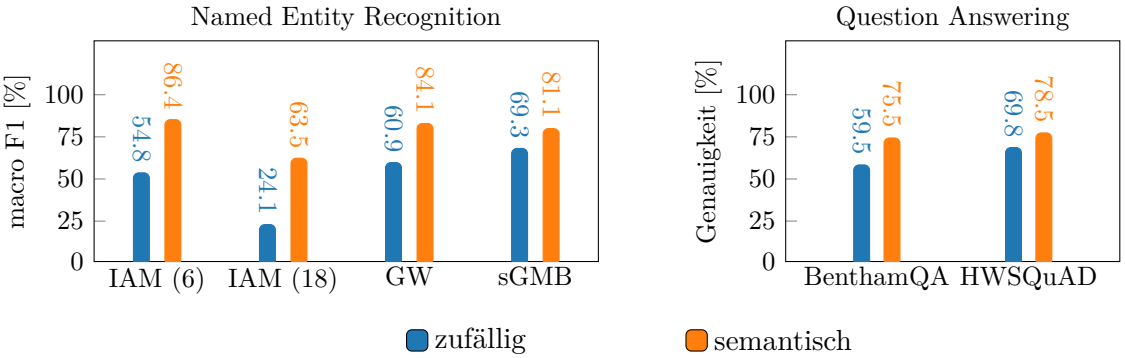


Abbildung 6.7: Einfluss von vortrainierten semantischen Worteinbettungen auf die semantische Analyse von handschriftlichen Dokumentenbildern. Die Leistung der Einbettung wird auf Basis von NER- und QA-Benchmarks bestimmt. Die semantische Einbettung von Wörtern wird mit einem vortrainierten ELMo-Modell und die zufällige Einbettung mit dem Flair-Modell realisiert.

Die Relevanz der semantischen Repräsentationen wird in diesem Experiment durch den Vergleich von NLP-Modellen mit semantischer und zufällig initialisierter Wortrepräsentation auf QA- und NER-Benchmarks bestimmt. Die zufällig initialisierte aber anpassbare Worteinbettung repräsentiert die Bedingungen für HTR-freie Modelle. Um den Einfluss der Texterkennung bei der Bewertung der Relevanz zu vermeiden, erfolgt diese anhand der textuellen Annotationen der Wortbilder. Eine solche Analyse ist jedoch bei state-of-the-art Modellen aus dem NLP-Bereich nur eingeschränkt möglich, da die semantischen Worteinbettungen in diesen Modellen keine externen Eingaben darstellen, sondern durch ein Vortraining des Modells erzeugt werden und somit integraler Bestandteil des Modells selbst sind. Aus diesem Grund werden für dieses Experiment die HTR-freien semantischen Architekturen aus Abschnitt 4.2 verwendet, welche die Wortrepräsentationen als externe Eingabe erwarten. Die Erzeugung der semantischen Einbettungen erfolgt mit dem ELMo-Modell, das eine hohe semantische Qualität aufweist. Diese Einbettung eignet sich jedoch nur bedingt für die Wahl als zufällig initialisiertes Modell, da große Datenmengen zum Erlernen dieser Repräsentationen benötigt werden. Daher wird für die zufällig initialisierten Worteinbettungen das Flair-Modell verwendet, das bereits für kleine Datenmengen gute Wortrepräsentationen erlernen kann [4].

Konkret dienen die Wortrepräsentationen, die aus den Annotationen der Wortbilder eines gegebenen Dokuments stammen, als Eingabe für die HTR-freien, anwendungsspezifischen NLP-Modelle. Diese Modelle werden dann durch ein Ende-zu-Ende-Training an die semantischen Benchmarks angepasst. Die Parameter des ELMo-Modells werden bei dem Training nicht beeinflusst. Für jeden Benchmark ergeben sich daraus zwei Modelle, von denen eines externes semantisches Vorwissen enthält und das andere ohne externe Informationen arbeitet.

Die Ergebnisse des Experiments sind für die NER- und QA-Benchmarks in Abbildung 6.7 dargestellt. Die Resultate verdeutlichen die fundamentale Bedeutung der Integration von externen semantischen Informationen für die semantische Analyse von Dokumenten. Dabei können durch die Berücksichtigung von semantischem Vorwissen auf allen Benchmarks deutliche Leistungssteigerungen erzielt werden. Insbesondere bei den NER-Benchmarks sind die Leistungsunterschiede zwischen den verwendeten Einbettungsverfahren erkennbar. Auf den meisten Benchmarks erreichen die Modelle mit zufällig initialisier-

ten Worteinbettungen bereits verhältnismäßig hohe Ergebnisse. Hierbei variieren die macro F1-Werte auf den NER-Benchmarks zwischen 24.1% und 69.3%. Durch die Verwendung von semantischen Worteinbettungen werden absolute Leistungssteigerungen zwischen 12% und 40% auf diesen Benchmarks erzielt. Die hohe Variabilität der Leistungsgewinne zwischen den Benchmarks ist auf die unterschiedliche Komplexität der Daten zurückzuführen. Insbesondere bei Datensätzen mit komplexen semantischen Beziehungen, wie der IAM-DB mit 18 Kategorien, zeigt sich der Vorteil externer semantischer Informationen.

Für die QA-Benchmarks werden mit den zufällig initialisierten Worteinbettungen bereits zwischen 59.5% und 69.8% der Fragen richtig beantwortet. Durch die Verwendung semantischer Worteinbettungen wird eine absolute Steigerung von 16% auf dem BenthamQA-Datensatz und von 8.7% auf dem HWSQuAD-Datensatz erreicht. Interessanterweise sind die Leistungssteigerungen auf den QA- im Vergleich zu den NER-Benchmarks teilweise deutlich geringer, obwohl die Beantwortung von Fragen die semantisch komplexere Aufgabe darstellt. Eine mögliche Erklärung hierfür ist die Verwendung der BIDAF-Architektur für das QA, die im Vergleich zu state-of-the-art QA-Modellen auf den meisten Benchmarks in der Literatur signifikant schlechtere Leistungen erzielt. Hierbei erreicht das auf der BERT-Architektur basierende QA-Modell aus Abschnitt 4.1.2 eine Genauigkeit von 87% auf dem BenthamQA-Datensatz und von 96% auf dem HWSQuAD-Datensatz. Wie bereits erwähnt, kann dieses Modell aufgrund der vortrainierten Transformer-Architektur nicht für einen adäquaten Vergleich mit zufällig initialisierten Worteinbettungen verwendet werden. Neben der Wahl der Architektur ist die stark unterschiedliche Größe der NER- und QA-Datensätze eine mögliche Erklärung für den geringen Unterschied zwischen semantischen und zufällig initialisierten Worteinbettungen. Die QA-Modelle haben im Training größere Textmengen verarbeitet und sind daher potentiell in der Lage, auch ohne vortrainierte semantische Worteinbettungen wichtige Beziehungen zwischen Wörtern zu extrahieren.

6.5 EVALUATION DER CROSS-MODALEN WISSENSDESTILLATION

Die Ergebnisse aus dem vorherigen Experiment haben gezeigt, dass die Integration von vortrainierten semantischen Worteinbettungen fundamental wichtig für die semantische Analyse von Dokumenten ist und zu einer enormen Leistungssteigerung bei semantischen Aufgaben führen kann. Ein zentraler Bestandteil dieser Arbeit ist daher die Integration von semantischen Vorwissen in das HTR-freie Modell aus Abschnitt 4.2. Dazu wird der im Kapitel 5 vorgestellte cross-modale Destillationsansatz in den folgenden Experimenten evaluiert. Das Verfahren basiert auf einer robusten Abbildung von handschriftlichen Wortbildern in einen vortrainierten semantischen Worteinbettungsraum. Aufgrund der hohen Komplexität dieser Aufgabe werden die im Kapitel 5 eingeführten Optimierungsverfahren zunächst intrinsisch evaluiert und anschließend auf Basis der erzielten Ergebnisse ein optimiertes Modell zur semantischen Wortbildeinbettung für die weiteren Evaluationen dieser Arbeit realisiert. Aus Gründen der Übersichtlichkeit werden die Evaluationen nur für einen repräsentativen Benchmark pro Anwendung durchgeführt, wobei die Ergebnisse für die übrigen Benchmarks im Anhang aufgeführt sind. Abschließend werden die im Abschnitt 5.4 vorgestellten Ansätze zur Integration des optimierten Einbettungsmodells in das HTR-freie Verfahren anhand der NER- und QA-Benchmarks evaluiert.

Die intrinsische Evaluation ist ein etablierter Ansatz im NLP-Bereich, bei dem die Qualität einer Wortrepräsentation unabhängig von spezifischen NLP-Aufgaben getestet

wird [154, 209, 222]. Dieser Ansatz ermöglicht eine effiziente und generische Evaluierung von Wortrepräsentationsverfahren, ohne diese anhand aufwendiger extrinsischer NLP-Benchmarks evaluieren zu müssen. Für diese Arbeit werden intrinsische Evaluationsmethoden benötigt, welche für ein handschriftliches Wortbild die Abweichung der vorhergesagten Repräsentation von dessen Zielrepräsentation im semantischen Raum sowie die semantische Qualität der vorhergesagten Repräsentation bewerten. Die Abweichung wird in dieser Arbeit mit der semantischen Schlüsselwortsuche und die semantische Qualität mit der Wortanalogie bewertet. Trotz der überwiegend beobachteten hohen Korrelation zwischen intrinsischen und extrinsischen Evaluationsergebnissen besteht keine uneingeschränkte Gewährleistung für diese Beziehung. Die Leistung einer Einbettung kann für eine bestimmte extrinsische Aufgabe auf der Grundlage ihrer intrinsischen Bewertungen zwar abgeschätzt, aber nicht garantiert werden [31, 208].

6.5.1 *Analyse semantischer Worteinbettungsverfahren*

Ein zentraler Parameter bei der cross-modalen Wissensdestillation ist die Wahl eines geeigneten Lehrermodells. Das Modell sollte möglichst leistungsfähige semantische Wortrepräsentationen kodieren und eine hohe Korrelation zwischen orthographischen und semantischen Eigenschaften von Wörtern aufweisen. Die Worteinbettungsmodelle aus dem NLP-Bereich verfolgen konzeptionell unterschiedliche Strategien zur Kodierung eines Eingabewortes, die vermutlich einen erheblichen Einfluss auf die Leistungsfähigkeit der cross-modalen Wissensdestillation haben. Daher werden in diesem Experiment die im Abschnitt 5.1 vorgestellten Worteinbettungsverfahren auf ihre Eignung zur semantischen Repräsentation von handschriftlichen Wortbildern evaluiert. Jedes der ausgewählten Worteinbettungsmodelle wird in diesem Experiment als Lehrermodell für die cross-modale Wissensdestillation verwendet und die Leistung der daraus resultierenden Schülermodelle bezüglich der Wortanalogie und der semantischen Schlüsselwortsuche auf der IAM-DB evaluiert. Dazu wird die Architektur des Schülermodells individuell an die textuellen Worteinbettungsmodelle angepasst, indem die Anzahl der Neuronen in der Ausgabeschicht des Netzwerks entsprechend der Ausgabedimensionalität des Lehrermodells gewählt wird.

Die semantischen Zielrepräsentationen für die handschriftlichen Wortbilder werden im Training auf der Grundlage ihrer textuellen Annotationen mit dem Lehrermodell erzeugt. Die Transformation von maschinenlesbaren Wörtern in Vektorrepräsentationen ist für die meisten der Ansätze wohldefiniert. Für die kontextsensitiven BERT- und ELMo-Modelle existieren jedoch verschiedene Verfahren zur Generierung einer Wortrepräsentation, die einen fundamentalen Einfluss auf die semantische Qualität der Repräsentationen haben [50]. Dabei kann sowohl die Ausgabe einzelner Schichten aus den BLSTM- oder Transformer-Modellen als auch deren Kombination als Wortrepräsentation dienen. Zur geeigneten Auswahl einer Wortrepräsentation aus den BERT- und ELMo-Modellen sind in Abbildung 6.8 die semantischen Qualitäten der einzelnen Schichten aus den Modellen visualisiert. Dazu werden die Ausgaben der Schichten auf Basis der textuellen Annotationen aus der IAM-DB bestimmt und deren Qualität mit dem WA-Benchmark bewertet. Bei dem BERT-Modell variieren die Leistungen der 24 Schichten zwischen 87.6% und 35.7%. Die ersten Schichten des Modells realisieren eine leistungsfähige Wortrepräsentation, während die letzten Schichten eine geringe Leistung auf dem Benchmark erzielen. Das ELMo-Modell verfügt lediglich über 3 Ausgabeschichten, deren Ergebnisse auf dem Benchmark zwischen 92.1% und 84.1% variieren. Analog zum BERT-Modell kodieren die vorderen Schichten des Mo-

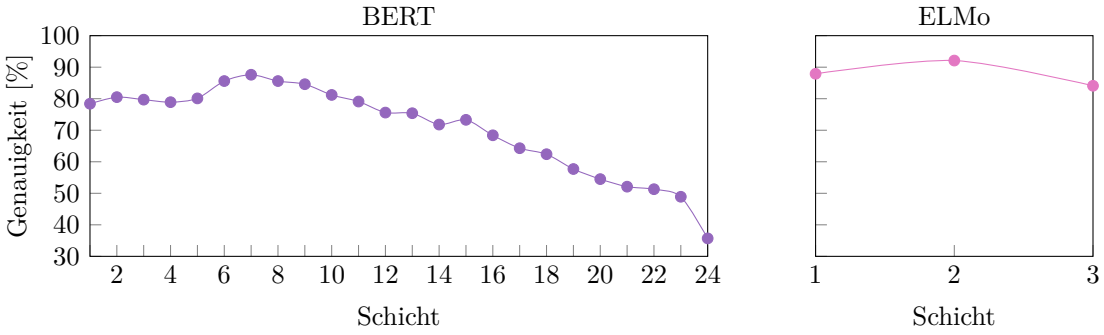


Abbildung 6.8: Die Eignung der einzelnen Schichten in den BERT- und ELMo-Modellen zur semantischen Repräsentation von Wörtern. Die semantische Qualität wird anhand der Wortanalogien aus dem IAM-Benchmark bestimmt.

dells leistungsstarke semantische Eigenschaften. Zusammenfassend zeigen die Ergebnisse analog zu [50], dass die ersten Schichten von kontextsensitiven Worteinbettungsmodellen die statischen Eigenschaften von Wörtern kodieren, während die letzten Schichten hauptsächlich für die Kodierung von kontextsensitiven Informationen zuständig sind. Daher wird in den folgenden Experimenten die Ausgabe der siebten Schicht für das BERT-Modell und die Ausgabe der zweiten Schicht für das ELMo-Modell als Wortrepräsentation verwendet.

In einem ersten Analyseschritt wird die semantische Qualität der Worteinbettungen unabhängig von ihrer Eignung zur Vorhersage auf Basis handschriftlicher Wortbilder ermittelt. Die semantischen Einbettungen der Wörter werden mit den Annotationen der Wortbilder bestimmt und bezüglich des WA-Benchmarks bewertet. Die WA-Werte repräsentieren nicht nur die semantische Qualität der Worteinbettungen, sondern stellen auch eine Obergrenze für die Qualität der Schülermodelle dar. Abbildung 6.9 zeigt die WA-Werte für die ausgewählten semantischen Wortrepräsentationen. Die Relevanz der Modellauswahl für das Lehrermodell wird angesichts der großen Leistungsunterschiede zwischen den Worteinbettungsverfahren deutlich. Mit der syntaktischen Einbettung des HWNetv2 werden lediglich 26.3% der Analogien im Benchmark korrekt aufgelöst. Zudem führt die rein zeichenbasierte Flair-Einbettung auf diesem Benchmark zu einer vergleichsweise schlechten Leistung von 48.3%. Die Word2Vec- und FastText-Verfahren steigern mit einer Genauigkeit von 62.2% bzw. 81.4% die Leistung gegenüber dem Flair-Modell erheblich. Die Wortrepräsentationen der BERT- und ELMo-Modelle führen im Vergleich zu den klassischen semantischen Worteinbettungsverfahren zu einer weiteren Verbesserung der Ergebnisse. Dabei löst das ELMo-Modell 92.1% und das BERT-Modell 87.6% der Wortanalogien korrekt auf.

Neben der semantischen Qualität der Worteinbettungen ist ihre Eignung für die Vorhersage auf Basis handschriftlicher Wortbilder von besonderer Relevanz in dieser Arbeit. Die semantische Qualität wird für die vorhergesagten Repräsentationen der Wortbilder mit der WA und die Abweichung zur Zielrepräsentation mit der semantischen Schlüsselwortsuche bewertet. Die Ergebnisse dieser Analysen sind in Abbildung 6.9 dargestellt. Bezüglich der QbE-Werte unterscheiden sich die Einbettungsverfahren nur geringfügig. Die Word2Vec- und FastText-Modelle erreichen einen mAP-Wert von etwa 80% und die anderen Modelle von etwa 90%. Somit werden Wortbilder mit der gleichen textuellen Annotation in der Regel auf eine ähnliche vektorielle Repräsentation abgebildet. Die Vor- und Nachteile der untersuchten Worteinbettungsverfahren werden anhand der QbS- und WA-Evaluationen ersichtlich. Generell ist eine hohe Korrelation zwischen den WA-Ergebnissen der Wort-

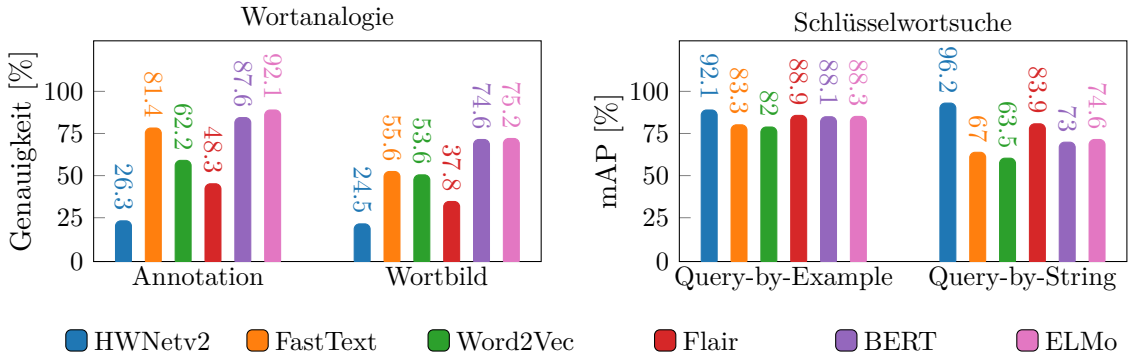


Abbildung 6.9: Die Qualität der untersuchten textuellen Wortrepräsentationsverfahren im Hinblick auf ihre Eignung zur Repräsentation von Semantik für handschriftliche Wortbilder. Dazu wird die semantische Qualität der Repräsentationen sowohl für die Annotationen als auch für die Wortbilder selbst mit dem WA-Benchmark der IAM-DB bestimmt. Die Abweichung zwischen den vorhergesagten Repräsentationen und den Zielrepräsentationen wird mit der QbE- und QbS-basierten Schlüsselwortsuche auf der IAM-DB bewertet.

bildrepräsentationen und den Repräsentationen der Annotationen zu beobachten, wobei die Werte für die Vorhersage auf Wortbildern deutlich schlechter sind. Die syntaktische Repräsentation des HWNetv2-Modells erreicht bei der QbS-basierten Schlüsselwortsuche eine mAP von 96.2%, woraus eine geringe Abweichung zwischen den Vorhersage- und Zielrepräsentationen abgeleitet werden kann. Mit nur 24.5% richtig gelösten Analogien kodiert die Darstellung jedoch wenig bis keine semantische Information. Die rein zeichenbasierte Flair-Einbettung bietet zwar eine Verbesserung der semantischen Qualität gegenüber dem HWNetv2 mit 37.8% aufgelösten Analogien, führt aber gleichzeitig zu einer reduzierten Vorhersagefähigkeit bezüglich der Zielrepräsentationen. Die Word2Vec- und FastText-Modelle weisen mit 53.6% bzw. 55.6% korrekt gelösten Analogien ähnliche semantische Qualitäten auf. Wie erwartet ist aufgrund der unabhängigen Worteinbettungen die Leistung des Word2Vec-Verfahrens im QbS-Benchmark mit 63.5% die niedrigste aller Verfahren. Durch die Berücksichtigung der n-gram-Informationen verbessert FastText die Leistung gegenüber Word2Vec auf dem Benchmark marginal auf 67%. Der beste Kompromiss zwischen der semantischen Qualität der Repräsentation und ihrer Eignung zur Vorhersage auf Wortbildern wird mit den BERT- und ELMo-Modellen erreicht. Diese Modelle lösen 74.6% bzw. 75.2% der Analogien korrekt und erzielen eine mAP von 73% bzw. 74.6% auf dem QbS-Benchmark. Obwohl die BERT-Einbettungen nur geringe Unterschiede zur ELMo-Repräsentation auf dem IAM-Benchmark aufweisen, bietet ELMo das beste Verhältnis zwischen WA- und QbS-Werten auf allen in dieser Arbeit getesteten Benchmarks.

Zusammenfassend zeigen die Ergebnisse der Experimente, dass die Wahl einer geeigneten textuellen Worteinbettung einen grundlegenden Einfluss auf die semantische Repräsentation von Wortbildern hat. Die ELMo-Einbettung bietet den besten Kompromiss zwischen der semantischen Qualität der Repräsentationen und ihrer Eignung für die Wortbildvorhersage und wird daher in den folgenden Experimenten dieser Arbeit als semantische Wortbildrepräsentation verwendet.

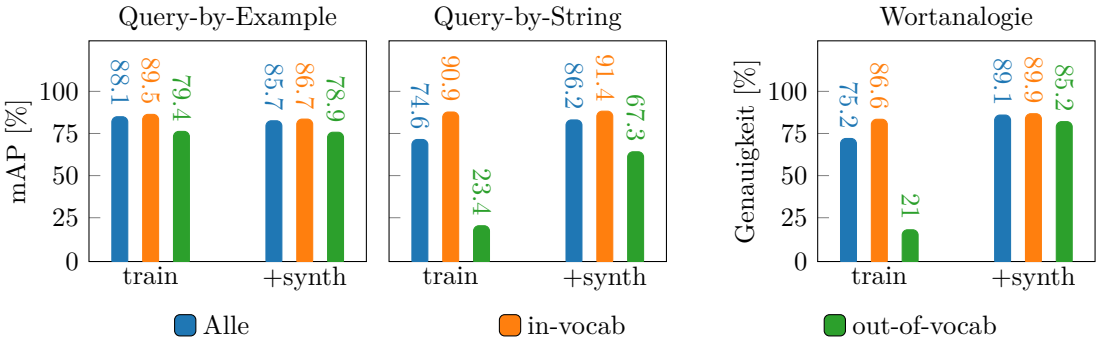


Abbildung 6.10: Die Leistung der vorhergesagten Wortbildrepräsentationen in Abhängigkeit von ihrem Vorkommen im Training des Destillationsprozesses. Die Bewertung erfolgt anhand der QbE-, QbS- und WA-Benchmarks auf der IAM-DB. Hierbei repräsentiert *in-vocab* alle Anfragen bzw. Analogien aus den Benchmarks, die als Annotation im Training des Destillationsprozesses vorkamen und *out-of-vocab* alle anderen Fälle. Zudem bezeichnet *train* die Verwendung aller Wortbilder aus der Trainingsmenge der IAM-DB und *+synth* die Hinzunahme von synthetisch generierten Wortbildern beim Training des Destillationsprozesses.

6.5.2 *Evaluation der annotationsfreien Wissensdestillation*

Die geringe Leistungsfähigkeit der Wortbildeinbettungen auf dem QbS-Benchmark im letzten Experiment deutet auf eine fehlerhafte Abbildung der Wortbilder in die semantischen Einbettungsräume hin. In dieser Arbeit wird die Hypothese aufgestellt, dass die Qualität der Vorhersage einer semantischen Repräsentation für ein Wortbild stark davon abhängt, ob Wortbilder mit der gleichen Annotation im Training des Destillationsprozesses vorkamen. Diese Hypothese beruht auf der Annahme, dass die Semantik eines nicht im Destillationsprozess einbezogenen Wortes nur schwer bestimmt werden kann. Die Annahme ist motiviert durch den fehlenden Kontext bei der Vorhersage der Wortbildrepräsentation und der im Allgemeinen schwachen Korrelation zwischen den orthographischen Merkmalen und den semantischen Repräsentationen. Zur Überprüfung dieser Hypothese wird die Leistung der vorhergesagten semantischen ELMo-Repräsentationen für die Wortbilder aus der Testmenge der IAM-DB in Abhängigkeit von ihrem Vorkommen im Training auf den QbS-, QbE- und WA-Benchmarks untersucht. Dazu werden die Anfragen bzw. Analogien aus den Benchmarks in disjunkte Mengen aufgeteilt. Dabei enthält *in-vocab* diejenigen Anfragen bzw. Analogien, die als Annotation im Training vorkamen und *out-of-vocab* alle anderen Fälle. Die Analyseergebnisse sind in Abbildung 6.10 dargestellt und unterstützen die aufgestellte Hypothese. Die QbE-basierte Schlüsselwortsuche ergibt eine mAP von 89.5% für Anfragebilder, bei denen Wortbilder mit der gleichen Annotation im Training des Destillationsprozesses vorkamen, und 79.4% für Anfragen, bei denen dies nicht der Fall war. Von besonderer Relevanz für die Bewertung der Abweichungen zwischen der vorhergesagten und der Zielrepräsentation eines Wortbildes ist die QbS-basierte Schlüsselwortsuche. Hier ergibt sich eine mAP von 90.9% für *in-vocab*-Anfragen und eine mAP von lediglich 23.4% für die *out-of-vocab*-Anfragen. Dies verdeutlicht den starken Zusammenhang zwischen der Qualität der Wortbildvorhersage und dem Vorhandensein eines Wortbildes mit der gleichen Annotation im Training des Destillationsprozesses. Die Auswertung hinsichtlich der semantischen Qualität der vorhergesagten Wortbildrepräsentationen unterstützt die aufgestellte Hypothese ebenfalls. Dabei werden Wortanalogien, für deren Zielwort mindestens

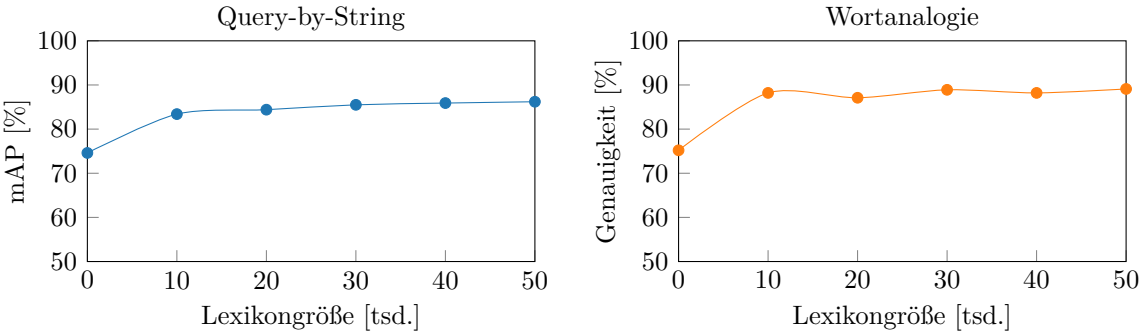


Abbildung 6.11: Der Einfluss von synthetisch generierten Wortbildern bei der Vorhersage der ELMo-Repräsentationen in Abhängigkeit von der Lexikongröße. Die Qualität der Vorhersagen wird mit den WA- und QbS-Benchmarks bewertet. Der Wert 0 auf der x-Achse repräsentiert die Anzahl der Wörter aus den Trainingsdaten der IAM-DB und jeder Wert $x > 0$ die Hinzunahme der $x \cdot 1000$ häufigsten englischen Wörter.

ein Wortbild mit der gleichen Annotation im Training vorkam, zu 86.6% richtig aufgelöst, während Analogien, für die dies nicht der Fall ist, nur zu 21% korrekt gelöst werden.

Um die Anzahl der nicht im Training vorkommenden Wörter zu reduzieren, wird im Folgenden der im Abschnitt 5.2 vorgestellte annotationsfreie Destillationsansatz evaluiert. Das Ziel dieses Ansatzes beruht auf der Annahme, dass mit einer hinreichend großen Menge an synthetisierten Wortbildern der semantische Raum ausreichend abgedeckt wird, sodass nur wenige Wortbildrepräsentationen falsch vorhergesagt werden und diese durch ein HTR-freies Modell auf Basis des gegebenen Kontextes intern korrigiert werden können. Eine interessante Frage in diesem Zusammenhang ist, wie sich die Lexikongröße zur Generierung der synthetischen Wortbilder auf die Leistung der QbS-basierten Schlüsselwortsuche und der WA-Benchmarks auswirkt. Zu diesem Zweck wird die Trainingsmenge der IAM-DB für den cross-modalen Destillationsansatz um synthetische Wortbilder erweitert, die auf unterschiedlichen Lexikongrößen basieren. In Abbildung 6.11 sind die Ergebnisse für die vorhergesagten ELMo-Repräsentationen aus dem Testdatensatz der IAM-DB dargestellt. Diese verdeutlichen den positiven Einfluss der Hinzunahme von synthetisch generierten Wortbildern auf die Leistung der cross-modalen Wissensdestillation. Durch die Hinzunahme erhöht sich die Genauigkeit von 75.2% auf 89.1% für die WA-Aufgabe und die mAP von 74.6% auf 86.2% für die QbS-basierte Schlüsselwortsuche. Die größte Leistungssteigerung wird bei der Berücksichtigung der 10000 häufigsten Wörter erzielt und ändert sich nicht wesentlich bei einer höheren Anzahl von Wörtern. Durch eine einfache Erhöhung der zu synthetisierenden Wortmenge lassen sich in den meisten Fällen nur marginale Verbesserungen erzielen, die sich sogar negativ auf die Ergebnisse auswirken können.

Der Ansatz der annotationsfreien Wissensdestillation verfolgt das Ziel, die Leistung für *out-of-vocab*-Anfragen zu verbessern. Die Ergebnisse in Abbildung 6.10 bestätigen diese Auswirkung, wobei sich die Leistung bei dem WA-Benchmark von 21% auf 85.2% und bei dem QbS-Benchmark von 23.4% auf 67.3% verbessert. Insgesamt steigt die Genauigkeit auf dem WA-Benchmark durch die Hinzunahme der synthetischen Daten von 75.2% auf 89.1% und die mAP auf dem QbS-Benchmark von 74.6% auf 86.2%. Beim QbE-Benchmark ist eine marginale Verschlechterung der mAP von 88.1% auf 85.7% zu beobachten. Dies ist vermutlich auf die Abweichungen zwischen synthetischen und echten handschriftlichen Wortbildern zurückzuführen.

6.5.3 *Evaluation von kombinierten Worteinbettungen*

Der annotationsfreie Destillationsansatz reduziert zwar die Fehler bei der Vorhersage semantischer Wortbildrepräsentationen, kann aber das ursprüngliche Problem der semantischen Wortbildeinbettung aufgrund von Wortneubildungen und des großen Wortschatzes von Sprachen nicht lösen. Eines der Hauptprobleme bei der fehlerhaften Vorhersage semantischer Wortbildrepräsentationen ist der Verlust von Informationen über den im Wortbild enthaltenen Text. Eine mögliche Lösung dieses Problems ist die Kombination von semantischen und syntaktischen Worteinbettungen zur Repräsentation handschriftlicher Wortbilder. Dies ermöglicht es HTR-freien NLP-Modellen, semantische Aufgaben auf der Basis orthographischer Merkmale zu lösen, auch wenn die semantischen Informationen für einen Teil der Wortbilder fehlerhaft sind.

In diesem Experiment wird das im Abschnitt 5.3 vorgestellte Verfahren zur geeigneten Kombination von semantischen und syntaktischen Wortbildeinbettungen evaluiert. Für diesen Ansatz wird die HWNetv2-Repräsentation mit einer Ausgabedimensionalität von 2048 für die syntaktische Wortbildrepräsentation und die ELMo-Repräsentation für die semantische Wortbildrepräsentation verwendet. Die Einbettungen werden hinsichtlich der QbE- und QbS-basierten Schlüsselwortsuche sowie der WA-Aufgabe auf der IAM-DB evaluiert. Dazu werden zwei separate Modelle mit dem im Abschnitt 4.2.1 vorgestellten Wortbildeinbettungsmodell und den Trainingsdaten aus der IAM-DB auf die Vorhersage der syntaktischen und semantischen Repräsentationen trainiert. Auf Basis der Modelle wird jedes Wortbild aus der Testmenge des IAM-Datensatzes in eine semantische und eine syntaktische Wortbildrepräsentation überführt.

Zunächst werden die Auswirkungen der einzelnen Optimierungsschritte aus Abschnitt 5.3 bewertet. Sowohl die syntaktische als auch die semantische Wortbildeinbettung erreichen bereits hohe mAP-Werte auf den QbE- und QbS-Benchmarks. In Bezug auf den QbE-Benchmark unterscheiden sich die Einbettungen nur marginal, wobei das syntaktische Modell einen mAP-Wert von 92.2% und das semantische Modell von 87.8% erreicht. Die Vorhersage der syntaktischen Wortbildrepräsentationen ist mit einer mAP von 96.3% im Vergleich zur semantischen mit 84.3% deutlich robuster. Allerdings weist das semantische Modell mit einer Genauigkeit von 88.2% eine hohe Qualität bei der Lösung von Analogien auf, während das syntaktische Modell nur 23.2% dieser Analogien löst. Bei einer einfachen Konkatenation der beiden Einbettungen wird die gemeinsame Repräsentation vollständig durch das syntaktische Modell dominiert. Dies führt im Vergleich zur rein syntaktischen Einbettung zu einer identischen Leistung auf den Benchmarks. Die Anpassung der Ausgabedimensionalität des HWNetv2-Modells auf 1024 verbessert die WA auf 32.4%. Durch die Standardisierung der Einbettungen ergibt sich eine weitere Verbesserung der WA auf 41.9%. Werden die Repräsentation zusätzlich auf eine L2-Norm von 1 normiert, so kann die konkatenierte Darstellung 65.4% der Analogien korrekt lösen. Bei Anwendung der Optimierungsschritte bleiben die mAP-Werte für die QbS- und QbE-basierte Schlüsselwortsuche nahezu konstant bei 92% bzw. 96%. Insgesamt führen die Optimierungsschritte vor der Konkatenation der beiden Repräsentationen zu einer Steigerung der semantischen Ausdrucksstärke bei nahezu gleichbleibender Leistung auf den QbE- und QbS-Benchmarks.

Ungeachtet der Optimierungen besteht weiterhin eine deutliche Dominanz von syntaktischen Eigenschaften in der gemeinsamen Wortbildrepräsentation. In diesem Experiment wird die Auswirkung des Gewichtungsparameters m aus Formel 5.12 evaluiert, wobei dieser bei einem Wert von 0 eine rein syntaktische und bei einem Wert von 1 eine rein semantische Repräsentation darstellt. Die Ergebnisse werden für die standardisierten und

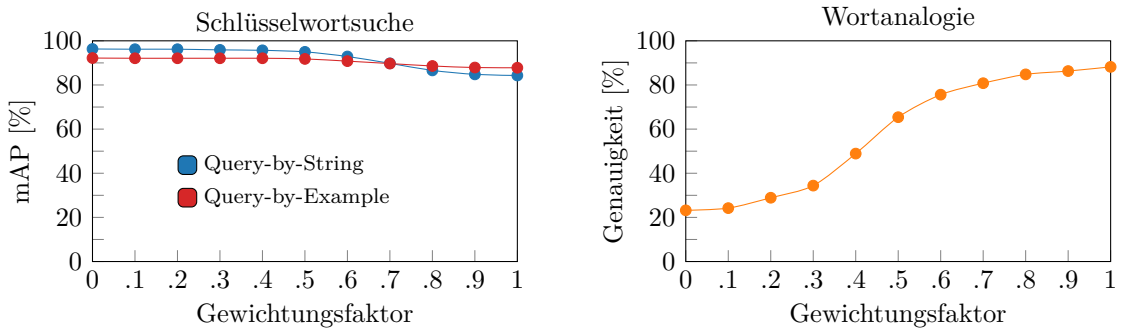


Abbildung 6.12: Auswirkungen der gewichteten Kombination von semantischen und syntaktischen Wortbildrepräsentationen auf Basis der IAM-DB. Die Leistung wird anhand der Schlüsselwortsuche und der Wortanalogie gemessen. Die Werte auf der x-Achse repräsentieren den Gewichtungsfaktor, wobei der Wert 0 für eine rein syntaktische und der Wert 1 für eine rein semantische Wortbildrepräsentation steht.

normalisierten Wortbildeinbettungen angegeben. Die Diagramme in Abbildung 6.12 zeigen den Verlauf der syntaktischen und semantischen Benchmarks bei unterschiedlicher Gewichtung der Repräsentationen. Im Allgemeinen liegen die QbE-Werte weitgehend konstant zwischen 92.2% und 87.8%. Die QbS-Werte sind streng monoton fallend in einem Intervall zwischen 96.3% und 84.3%. Der Verlust ist bis zu einer Gewichtung von 0.5 nur marginal, während die Leistung für alle Werte oberhalb dieses Gewichts annähernd linear abnimmt. Die größten Leistungsunterschiede sind bei den streng monoton steigenden WA-Werten zu beobachten. Hier werden Werte zwischen 23.2% und 88.2% erreicht und weisen einen sigmoidalen Funktionsverlauf auf. Bis zu einem Gewicht von 0.3 und ab einem Gewicht von 0.7 gibt es einen marginalen Anstieg bezüglich der Auflösung der Analogien in der IAM-DB, während zwischen den genannten Bereichen ein wesentlicher Anstieg der WA-Werte zu beobachten ist. Generell ist eine Zunahme des QbS-Wertes mit einer Abnahme des WA-Wertes verbunden und umgekehrt. Daher ist bei der Auswahl einer geeigneten Kombination aus semantischen und syntaktischen Wortrepräsentationen ein Kompromiss zwischen der QbS- und der WA-Leistung notwendig. Für alle getesteten Datensätze zeigt sich, dass eine optimale Balance zwischen diesen Werten bei einer Gewichtung von 0.7 erreicht wird.

6.5.4 *Evaluation von Integrationsansätzen*

Nachdem ein optimiertes Modell zur semantischen Wortbildrepräsentation in den vorherigen Experimenten erarbeitet wurde, werden im Folgenden die im Abschnitt 5.4 vorgestellten Ansätze zur Integration dieses Modells in das HTR-freie Verfahren evaluiert. Die Leistung der Ansätze wird mit den NER- und QA-Benchmarks ermittelt. Die vorgestellten Verfahren basieren auf einem vortrainierten semantischen Wortbildeinbettungsmodell zur Vorhersage von ELMo-Repräsentationen. Dazu wird das Modell zunächst mit dem Verfahren der cross-modalen Wissensdestillation auf dem synthetischen Datensatz vortrainiert. Anschließend wird es mit den Wortbildern aus den Trainingsdaten der entsprechenden Benchmarks angepasst. Wenn möglich, wird zusätzlich zu den Ansätzen eine zufällige Initialisierung des Wortbildeinbettungsmodells im HTR-freien Verfahren als eine Art *Baseline* für die Integrationsansätze evaluiert.

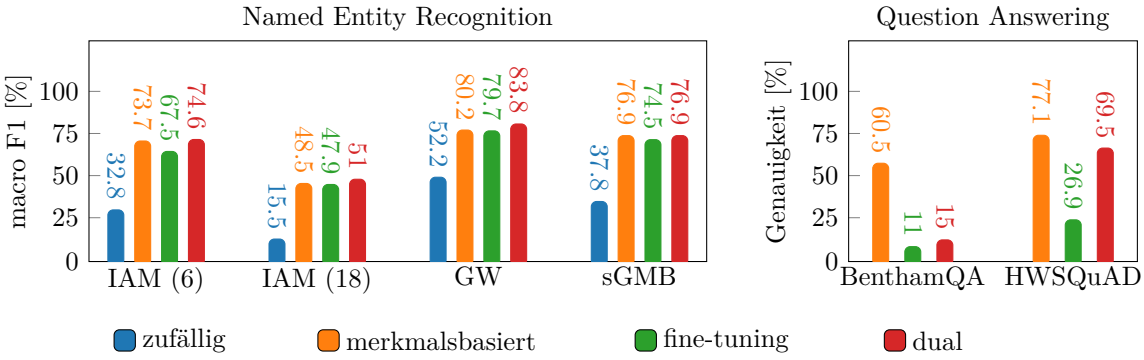


Abbildung 6.13: Vergleich der vorgestellten Ansätze zur Integration des optimierten semantischen Worteinbettungsmodells in das HTR-freie Verfahren. Die Leistung der Ansätze wird auf den Evaluationsbenchmarks für die NER und das QA angegeben.

Die Ergebnisse der Evaluationen sind in Abbildung 6.13 dargestellt. Die Leistungen der Ansätze variieren stark in Abhängigkeit von der semantischen Aufgabenstellung. Die drei vorgestellten Integrationsansätze erzielen bei allen NER-Benchmarks im Vergleich zu den zufällig initialisierten Modellparameter deutlich bessere Ergebnisse. Die Resultate der zufällig initialisierten Modelle unterscheiden sich auf den Benchmarks mit Leistungen zwischen 15.5% und 52.2% stark voneinander und sind ein Indikator für die unterschiedliche Komplexität der Benchmarks. Der duale Ansatz erreicht auf allen NER-Benchmarks die besten Ergebnisse. Die Abweichungen zwischen den Ergebnissen der Ansätze sind auf allen NER-Benchmarks weitgehend identisch. Das fine-tuning-Modell erzielt dabei die geringste Leistung der drei Integrationsansätze. Insgesamt unterscheiden sich die Ergebnisse der drei Ansätze auf den Benchmarks jedoch nur marginal voneinander, wobei der duale Ansatz am besten und der fine-tuning-Ansatz am schlechtesten abschneidet. Die maximalen Abweichungen der drei Ansätze auf den NER-Benchmarks liegen zwischen 2.4% und 7.1%.

Für die QA-Benchmarks wird keine Baseline-Methode mit zufälliger Initialisierung angegeben, da diese aufgrund des maschinenlesbaren Formats der zu beantwortenden Frage nicht ohne weiteres anwendbar ist. Bei beiden QA-Benchmarks erzielt der fine-tuning-Ansatz deutlich schlechtere Ergebnisse als der merkmalsbasierte Ansatz. Dies ist vermutlich auf die komplexere Architektur und Problemstellung im Vergleich zur NER-Aufgabe zurückzuführen. Die Kombination aus der hohen Komplexität und den anpassbaren Parametern des Worteinbettungsmodells kann zu einer Überanpassung an die Trainingsdaten und damit zu einer geringen Generalisierungsfähigkeit des HTR-freien Modells führen. Darüber hinaus beantwortet der duale Ansatz auf dem BenthamQA-Datensatz lediglich 15% der Fragen korrekt. Die geringe Leistung ist auf die Verwendung von ausschließlich synthetisch generierten Wortbildern während des QA-Trainings zurückzuführen, da diese nicht repräsentativ für die historischen Wortbilder aus dem Testdatensatz sind.

Insgesamt erzielt der duale Ansatz auf den NER-Benchmarks durchgängig die besten Ergebnisse, während auf den QA-Benchmarks der merkmalsbasierte Ansatz die höchsten Leistungen erreicht. In Anbetracht der nur geringfügig schlechteren Werte für den merkmalsbasierten Ansatz bei den NER-Benchmarks und dem im Vergleich zu den Ende-zu-Ende-Ansätzen deutlich geringeren Ressourcenbedarf beim Training empfiehlt sich in der Praxis der Einsatz des merkmalsbasierten Ansatzes.

Tabelle 6.1: Vergleich von Ansätze zur semantischen Schlüsselwortsuche unter Verwendung der Genauigkeit [%] für die WA-Aufgabe und der mAP [%] für die QbE- und QbS-basierte Schlüsselwortsuche.

		GW			*IAM-DB*		
Ansatz	semantisch	QbE	QbS	WA	QbE	QbS	WA
PHOCResNet [190]		97.8	98.0	—	85.5	94.1	—
HWNetv3 [94]		99.5	99.8	—	93.2	97.5	—
Triplet-CNN [233]	✓	96.9	69.8	—	81.6	75.7	—
Sem-MSE [98]	✓	97.6	94.4	—	84.6	69.7	63.2
Sem-Rank [98]	✓	97.8	93.7	—	83.3	71.3	65.6
Combined [98]	✓	99.4	98.8	—	90.6	94.3	61.5
HTR-frei	✓	98.1	98.9	100.0	85.7	86.2	89.1
HTR-basiert	✓	97.0	96.6	100.0	77.3	84.0	73.7

		HWSQuAD			*BenthamQA*			*sGMB*		
Ansatz	semantisch	QbE	QbS	WA	QbE	QbS	WA	QbE	QbS	WA
HTR-frei	✓	99.5	96.9	84.2	72.8	74.6	49.7	94.2	91.5	82.3
HTR-basiert	✓	98.9	98.9	83.8	49.1	56.5	35.1	89.1	91.3	84.3

6.6 ANWENDUNGSSPEZIFISCHE EVALUATION

In den vorhergehenden Abschnitten wurden wichtige Hyperparameter für das HTR-freie Verfahren zur semantischen Analyse von handschriftlichen Dokumentenbildern ermittelt. Im Folgenden wird die Leistungsfähigkeit der optimierten HTR-freien und HTR-basierten Modelle anhand mehrerer Benchmark-Datensätze evaluiert und mit Ansätzen aus der Literatur verglichen. Die Ergebnisse der Modelle und deren Vergleich sind für die semantische Schlüsselwortsuche im Abschnitt 6.6.1, für die NER im Abschnitt 6.6.2 und für das QA im Abschnitt 6.6.3 gegeben. Zusätzlich werden für die semantischen Aufgaben qualitative Analysen durchgeführt.

6.6.1 *Semantische Schlüsselwortsuche*

In der Literatur existieren derzeit nur wenige Veröffentlichungen im Bereich der semantischen Schlüsselwortsuche. Die wenigen publizierten Ansätze wurden ausschließlich auf der IAM-DB und dem GW-Datensatz evaluiert. Tabelle 6.1 zeigt die Ergebnisse der semantischen [98, 233] und der relevantesten syntaktischen [94, 190] Modelle zur Schlüsselwortsuche aus der Literatur, sowie die Resultate der HTR-freien und HTR-basierten Modelle aus dieser Arbeit. Dabei kann der GW-Datensatz als de-facto gelöst betrachtet werden, da sowohl die Ergebnisse bezüglich der semantischen als auch der syntaktischen Metriken nahezu fehlerfrei sind. Insbesondere auf dem IAM-Datensatz wird der Unterschied

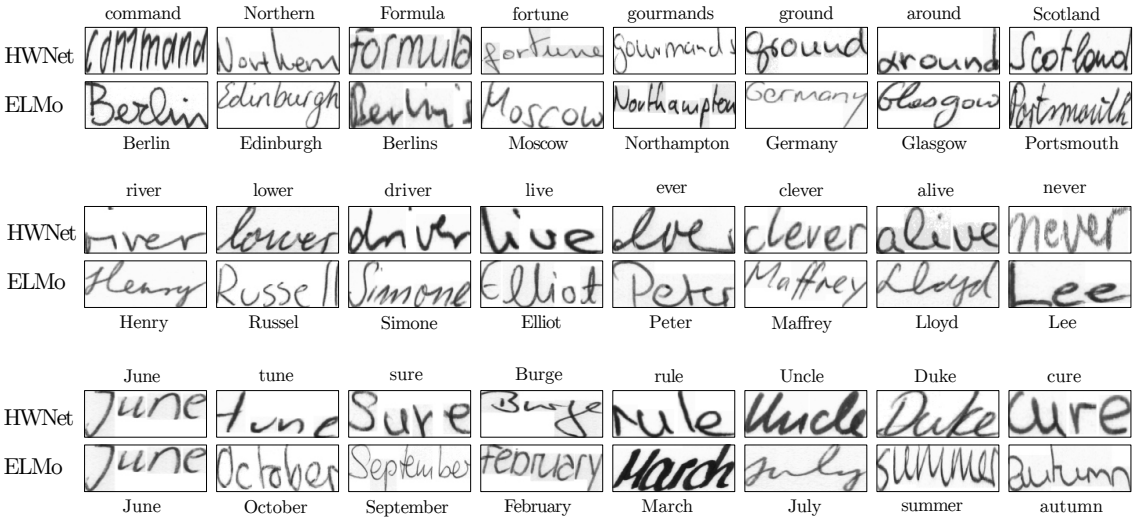


Abbildung 6.14: Qualitativer Vergleich zwischen den Top-8 Retrieval-Ergebnissen der syntaktischen (HWNet) und semantischen (ELMo) Wortbildeinbettungsmodelle anhand der QbS-Anfragewörter „Dortmund“, „Oliver“ und „June“. Die Retrieval-Listen sind mit dem HTR-freien Ansatz auf dem Testdatensatz der IAM-DB erstellt worden.

zwischen dem HTR-freien und dem HTR-basierten Modell ersichtlich. Hierbei sinken die Leistungen bei einer expliziten Texterkennung für das QbE-Retrieval und die WA-Aufgabe wesentlich. Im Vergleich zu den Ansätzen aus der Literatur können mit dem optimierten Wortbildeinbettungsverfahren speziell die QbS- und WA-Leistungen auf der IAM-DB wesentlich verbessert werden. Auch wenn die Kombination von semantischen und syntaktischen Repräsentationen in [98] zu besseren QbS-Resultaten führt, weist diese eine geringere semantische Qualität bzgl. der WA auf und bietet im Allgemeinen einen schlechteren Kompromiss zwischen den beiden Metriken. Auf den synthetischen HWSQuAD- und sGMB-Datensätzen können keine eindeutigen Vorteile für einen der beiden Ansätze aus dieser Arbeit identifiziert werden. Bei dem herausfordernden BenthamQA-Datensatz hingegen zeigen sich deutliche Vorteile für den HTR-freien Ansatz bei allen Metriken.

Eine qualitative Analyse der Retrieval-Ergebnisse des HTR-freien Ansatzes ist in Abbildung 6.14 visualisiert. Dazu werden die Top-8 Wortbilder aus den Retrieval-Listen für verschiedene QbS-Anfragen sowohl mit syntaktischen als auch mit semantischen Wortbildeinbettungsmodellen auf dem IAM-Benchmark dargestellt. Das erste Anfragewort „Dortmund“ ist in diesem Beispiel nicht Bestandteil der Testmenge und führt bei der syntaktischen Wortbildeinbettung zu einer Trefferliste, die für die Exploration des Datensatzes keinen echten Mehrwert bietet. Die semantische Rückgabeliste liefert hingegen sinnvolle Ergebnisse, selbst wenn der Suchbegriff nicht in den Dokumenten vorkommt. In diesem Beispiel bestehen die ersten acht Treffer aus Städte- und Ländernamen, wobei das erste Wortbild die deutsche Hauptstadt zeigt. Die Anfrage nach dem Wort „Oliver“ ergibt ein ähnliches Verhalten, wobei in den semantischen Retrieval-Listen überwiegend männliche Vornamen zu finden sind. Die syntaktische Wortbildeinbettung mit dem Anfragewort „June“ findet zwar alle syntaktisch relevanten Wortbilder des Datensatzes, jedoch sind die nachfolgenden Wortbilder für die Exploration des Datensatzes weniger hilfreich. In der semantischen Trefferliste werden weitere Monatsnamen und Jahreszeiten aufgeführt. Zusammenfassend zeigen die qualitativen Ergebnisse, dass die Verwendung von syntaktischen Wortbildeinbettungen bei der Suche nach Wortvorkommen mit gleicher Transkription sehr

Berlin	:	Germany	|	Moscow	:		Berlin	:	Germany	|	London	:	
England	:	English	|	Italy	:		bright	:	brighter	|	simple	:	
going	:	went	|	seeing	:		old	:	oldest	|	good	:	
boy	:	girl	|	sons	:		run	:	running	|	walk	:	
slow	:	slowly	|	serious	:		Germany	:	German	|	Austria	:	

Abbildung 6.15: Qualitative Beispiele für Wortanalogien auf dem IAM-Benchmark für die vorhergesagten ELMo-Repräsentationen. Die korrekt aufgelösten Analogien sind in grün und die falsche gelösten in rot dargestellt. Bei einer fehlerhaften Vorhersage steht die richtige Antwort der Analogie unter dem jeweiligen Wortbild. Die Analogien sind im Format $a : b \mid c : d$ angegeben, wobei die folgende Interpretation gilt: a verhält sich zu b wie c zu d.

effektiv ist. Für die meisten realen Anwendungsfälle sind jedoch in der Regel die semantischen Suchergebnisse interessanter, da diese zusätzliche Wortbilder in der Dokumentensammlung identifizieren, die in einem semantischen Zusammenhang mit der Anfrage stehen.

Die qualitativen Ergebnisse für die Wortanalogie auf dem IAM-Benchmark sind in Abbildung 6.15 dargestellt. Die Beispiele basieren auf den ELMo-Repräsentationen des IAM-Testdatensatzes, die vom HTR-freien Modell vorhergesagt wurden. Die Analogien sind im Format $a : b \mid c : d$ angegeben, wobei a, b und c durch maschinenlesbare Wörter vorgegeben sind. Für die Analogie gilt folgende Interpretation: a verhält sich zu b wie c zu d. Zur Lösung der Analogie muss ein handschriftliches Wortbild d aus der Testmenge ermittelt werden. Die Lösung der Analogien erfordert sowohl grammatische Kenntnisse als auch ein Verständnis der semantischen Beziehungen zwischen Wörtern. Die Qualität der semantischen Wortbildeinbettung lässt sich anhand der korrekten Auflösung komplexer Analogien in diesen Beispielen demonstrieren. Bei den falsch gelösten Analogien fällt auf, dass die Antworten zwar aus der richtigen semantischen Kategorie stammen, aber nicht das richtige Wort liefern. Dies deutet darauf hin, dass die fehlerhafte Auflösung der Analogien in der Regel nicht auf die semantische Qualität der textuellen Wortrepräsentation zurückzuführen ist, sondern auf die fehlerhafte Abbildung der Wortbilder in den semantischen Einbettungsraum.

6.6.2 *Named Entity Recognition*

Die NER-Ansätze aus der Literatur unterscheiden sich hinsichtlich der Segmentierung der Eingabedaten auf Wort- [22, 166, 206], Zeilen- [220] und Seitenebene [39]. Aufgrund fehlender Ergebnisse von wortsegmentierten Modellen auf den meisten NER-Benchmarks dieser Arbeit wurden die Modelle aus [166] und [206] vom Autor dieser Arbeit bestmöglich reimplementiert und auf den Benchmarks evaluiert. In Tabelle 6.2 sind die Ergebnisse der in dieser Arbeit vorgestellten Ansätze und der Modelle aus der Literatur aufgeführt. Der HTR-freie und der HTR-basierte Ansatz erzielen vergleichbare Ergebnisse für alle Benchmarks. Die macro F1-Werte des HTR-basierten Ansatzes sind für beide Versionen der

Tabelle 6.2: Vergleich zwischen den in dieser Arbeit vorgestellten Modellen und den in der Literatur veröffentlichten NER-Ansätzen. Die Ergebnisse sind in macro F1 [%] angegeben. Für die Ansätze sind die Anforderungen hinsichtlich der Segmentierung auf Wort- (W), Zeilen- (Z) und Seitenebene (S) spezifiziert. Zusätzlich wird eine Baseline des HTR-basierten Ansatzes evaluiert, der auf den Gold-Standard Textannotationen der eingegebenen Wortbilder anstelle von deren Transkriptionen basiert.

	Segmentierung			*Benchmarks*			
Ansatz	W	Z	S	IAM (6)	IAM (18)	GW	sGMB
Toledo et al. [206]*	✓			37.4	18.0	45.3	38.8
Carbonell et al. [22]	✓			—	—	—	53.5
Rowtula et al. [166]*	✓			54.6	30.3	66.6	60.1
Line-level CN [220]		✓		57.2	46.5	68.8	—
Dessurt [39]			✓	71.1	48.5	—	—
HTR-basiert	✓			76.4	53.6	81.3	75.8
HTR-frei	✓			74.6	51.0	83.8	76.9
HTR-basiert (Ann.)	✓			87.5	63.5	89.6	80.2

**Der Ansatz wurde vom Autor dieser Arbeit reimplementiert.*

IAM-DB marginal höher als die des HTR-freien Verfahrens, wobei letzteres wiederum zu höheren Werten bezüglich der GW- und sGMB-Benchmarks führt. Die geringere Leistung des HTR-basierten Ansatzes auf den sGMB- und GW-Benchmarks ist etwas überraschend, da die Zeichenfehlerraten bei der Texterkennung mit circa 3% auf diesen Datensätzen verhältnismäßig niedrig sind. Im Vergleich zu wortsegmentierten Ansätzen aus der Literatur bieten die in dieser Arbeit vorgestellten Verfahren eine Verdoppelung der Leistung gegenüber dem schlechtesten Modell sowie eine absolute Steigerung der F1-Werte um 20% bis 30% gegenüber dem besten Modell. Die in [166, 206] vorgestellten Ansätze verwenden eine identische NER-Architektur wie das HTR-freie Modell dieser Arbeit, integrieren jedoch kein externes semantisches Wissen. Stattdessen wird das Modell in einem Ende-zu-Ende-Verfahren mit zufällig initialisierten Gewichten trainiert. Die deutlich niedrigeren macro F1-Werte auf allen Benchmarks im Vergleich zum HTR-freien Ansatz dieser Arbeit unterstreichen erneut die Vorteile des eingeführten cross-modalen Ansatzes zur Wissensdestillation. Der Vergleich von Ansätzen mit unterschiedlicher Segmentierung ist nicht ohne weiteres möglich, obwohl die zeilen- und seitenbasierten Ansätze eine optimierte Zuordnung der vorhergesagten Annotationen zu den Gold-Standard-Annotationen durchführen. Bei den Ende-zu-Ende-Verfahren führt die Verwendung von Zeilenbildern anstelle von Wortbildern nur zu marginalen Verbesserungen beim GW- und IAM (6)-Benchmark. Beim IAM (18)-Benchmark ist der macro F1-Wert hingegen um 16 Prozentpunkte höher im Vergleich zu [166]. Das Dessurt-Modell [39] erzielt im Vergleich zu den anderen Ansätzen aus der Literatur hohe F1-Werte auf den Benchmarks, was vermutlich auf die implizite Kodierung von semantischem Wissen durch das selbstüberwachte Vortraining zurückzuführen ist. Jedoch erzielen sowohl das HTR-freie als auch das HTR-basierte Modell dieser Arbeit um circa 5% höhere F1-Werte auf den IAM-Benchmarks. Die Ergebnisse deuten darauf hin, dass die Integration semantischer Informationen mit dem cross-modalen Destillationsan-

Abbildung 6.16: Qualitative Analyse der NER-Modelle aus dieser Arbeit auf dem IAM-Benchmark. Die Transkriptionen des HTR-Modells befinden sich über den handschriftlichen Wortbildern, wobei Texterkennungsfehler rot markiert sind. Die NEs werden für die Wortbilder im Format *a/b/c* angegeben, wobei *a* der Gold-Standard Annotation, *b* der Vorhersage des HTR-freien Modells und *c* der Vorhersage des HTR-basierten Modells entspricht.

satz Vorteile gegenüber dem Lernen dieser Informationen auf der Basis von Transformern und großen Korpora von handschriftlichen Dokumentenbildern aufweist. Wie bereits in [39] erwähnt, ist der Unterschied zwischen dem HTR-basierten und dem Dessurt-Modell wahrscheinlich auf die Verwendung des leistungsfähigeren Sprachmodells RoBERTa zurückzuführen, da die Texterkennung ähnliche Fehlerraten aufweist. Die Verwendung von textuellen Annotationen im Gegensatz zu vorhergesagten Transkriptionen als Eingabe für das textuelle NER-Modell dieser Arbeit führt insbesondere bei den IAM-Benchmarks zu einem absoluten Leistungsgewinn von über 12%. Dies verdeutlicht den starken Einfluss von Texterkennungsfehlern auf die Leistungsfähigkeit von nachfolgenden NER-Modellen.

In Abbildung 6.16 sind qualitative Ergebnisse der vorgestellten HTR-freien und HTR-basierten Ansätze auf dem IAM-Benchmark mit 6 Kategorien visualisiert. Beide Modelle erreichen eine hohe Qualität bei der Identifikation und Klassifikation von NEs auf den beispielhaften handschriftlichen Dokumentenbildern. Der HTR-basierte Ansatz erkennt und klassifiziert viele Entitäten trotz Texterkennungsfehlern korrekt. Allerdings führen bereits geringfügige Fehler bei der Texterkennung zu einer falschen Klassifikation der Wortbilder, z.B. wird das Wortbild mit der Annotation „right“ fälschlicherweise als „night“ transkribiert und anschließend als Zeitangabe klassifiziert. Das größte Problem für den HTR-basierten Ansatz stellen Wortbilder dar, bei denen der erste Buchstabe fälschlicherweise als Kleinbuchstabe transkribiert wurde. In diesen Fällen ist es oft nicht möglich, diese Wortbilder als Entität zu identifizieren. Mit dem HTR-freien Modell können die meisten dieser Wortbilder hingegen korrekt identifiziert werden. Allerdings ist die Identifikation von Entitäten, die aus mehr als einem Wortbild bestehen, für das HTR-freie

Tabelle 6.3: QA-Ergebnisse auf den HWSQuAD- und BenthamQA-Datensätzen. Die Ergebnisse werden unabhängig voneinander für das Retrieval (R), die Antwortextraktion (Q) und dem gesamten Prozess (R+Q) angegeben. Zusätzlich wird eine Baseline des HTR-basierten Ansatzes evaluiert, der auf den Gold-Standard Textannotationen der gegebenen Wortbilder anstelle von deren Transkriptionen basiert.

	HWSQuAD			*BenthamQA*		
Ansatz	R	Q	R+Q	R	Q	R+Q
Mathew et al. (HTR-frei) [133]	46.5	—	15.9	55.5	—	17.5
Mathew et al. (HTR-basiert) [133]	86.1	—	59.3	32.0	—	2.5
BIDAF-Line (PHOC) [213]	86.2	68.1	45.0	92.5	50.5	37.5
HTR-basiert	89.7	95.1	73.5	85.5	50.0	41.5
HTR-frei	87.0	77.1	53.3	94.5	60.5	51.0
HTR-basiert (Ann.)	90.0	96.0	74.4	98.5	87.0	80.0

Modell oft problematisch. Hierbei wird beispielsweise nur das Wortbild von „Germany“ in „West Germany“ als Entität identifiziert. Die Fehler der Modelle sind in der Regel plausibel. So werden beispielsweise die Wortbilder mit den Annotationen „Nato headquarter“ von beiden Modellen nicht als Orte klassifiziert, sondern nur das Wortbild von „Nato“ als geopolitische Entität.

6.6.3 *Question Answering*

Analog zur semantischen Schlüsselwortsuche existieren in der Literatur nur wenige QA-Modelle für handschriftliche Dokumentenbilder, mit denen die Ansätze dieser Arbeit verglichen werden können. Es werden lediglich in [133] sowohl ein HTR-freies als auch ein HTR-basiertes QA-Modell für diese Problemstellung vorgestellt. Tabelle 6.3 zeigt einen Leistungsvergleich zwischen den in dieser Arbeit vorgestellten Ansätzen und verwandten Verfahren aus der Literatur. Die Ergebnisse demonstrieren die Anfälligkeit von HTR-basierten QA-Modellen für Eingaben mit hohen Fehlerraten bei der Texterkennung. Sowohl der HTR-basierte Ansatz dieser Arbeit als auch das in [133] publizierte Verfahren erzielen trotz state-of-the-art QA-Modellen nur unzureichende Ergebnisse von 41.5% bzw. 2.5% auf dem BenthamQA-Datensatz. Die absolute Leistungssteigerung des HTR-basierten Ansatzes dieser Arbeit um fast 40% im Vergleich zu dem Verfahren aus [133] ist vermutlich auf die erhöhte Robustheit des HTR-Modells zurückzuführen. Dazu wird das Modell für den BenthamQA-Benchmark zusätzlich mit realen handschriftlichen Wortbildern aus der IAM-DB und dem GW-Datensatz trainiert und verringert damit die WER von 76.8% auf 43.8%. Bei einer Eingabe ohne Texterkennungsfehler können 80% der Fragen des BenthamQA-Benchmarks mit dem Ansatz dieser Arbeit richtig beantwortet werden. Die Texterkennungsfehler wirken sich vor allem auf die Leistung des QA-Modells aus, während das Retrieval mit 85.5% Top-5 Genauigkeit trotz der hohen CER relativ robuste Leistungen zeigt. Die Vorteile des HTR-basierten QA-Ansatzes werden bei Eingaben mit einer geringen Fehlerrate bei der Texterkennung deutlich. Dabei werden 95.1% der Fragen bei vorliegen des korrekten Dokumentenbildes im HWSQuAD-Benchmark richtig beantwortet.

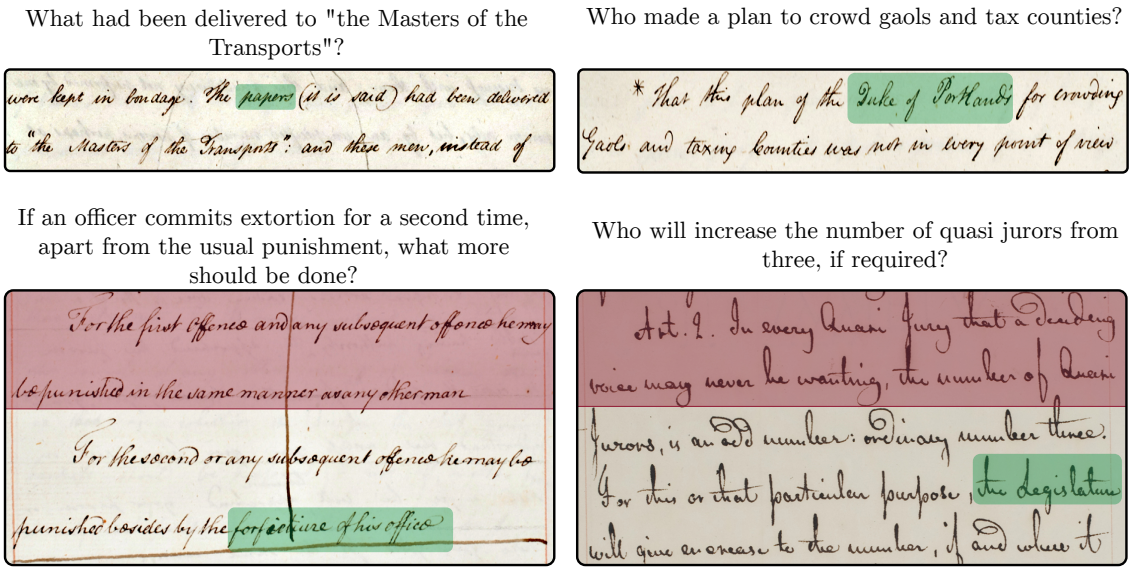


Abbildung 6.17: Qualitative Beispiele des HTR-freien QA-Modells auf dem BenthamQA-Benchmark. Die Gold-Standard Antworten sind in grün und die falsch vorhergesagten Bildbereiche in rot dargestellt.

Der HTR-freie Ansatz dieser Arbeit erzielt sowohl auf dem BenthamQA- als auch auf dem HWSQuAD-Benchmark mit 53.5% und 51% korrekt beantworteten Fragen robuste Ergebnisse. Im Vergleich zu dem HTR-freien Ansatz aus [133] werden deutliche Leistungsverbesserungen mit dem in dieser Arbeit vorgestellten Verfahren erzielt. Besonders der Retrieval-Ansatz zeigt eine hohe Robustheit bei der Identifikation der relevanten Dokumentenbilder mit 94.5% Top-5 Genauigkeit auf dem BenthamQA-Benchmark. In [213] wurde das adaptierte BIDAF-Modell vom Autor dieser Arbeit bereits veröffentlicht. Hierbei wird anstelle der semantischen ELMo-Repräsentation die syntaktische PHOC-Darstellung als Wortbildrepräsentation gewählt. Die Verwendung der semantischen anstelle der syntaktischen Wortbildrepräsentation weist bei beiden Benchmarks deutliche Vorteile auf. Die Leistung des HTR-freien Modells ist im Vergleich zum HTR-basierten Ansatz bei der Extraktion von Antworten auf dem HWSQuAD-Benchmark gering. Dies verdeutlicht die in der Literatur bereits demonstrierte Überlegenheit von Transformern gegenüber BLSTM-basierten QA-Modellen [42].

Ein fundamentales Problem der vorgestellten Ansätze ist die Kombination der Retrieval- und Extraktionsmodelle. Dabei erzielen die jeweiligen Ansätze teilweise Ergebnisse von über 90% beim Retrieval bzw. dem QA auf einem Einzeldokument und nach ihrer Kombination nur vergleichsweise geringe Ergebnisse um die 50% bis 70%. Für den HWSQuAD-Datensatz ist dies wahrscheinlich auf die ursprüngliche Entwicklungsidee zurückzuführen, wonach die Antwort aus einem einzelnen Dokument und nicht aus einer Dokumentensammlung extrahiert werden muss. Diese Annahme wird durch die Leistung des BenthamQA-Datensatzes unterstützt, der speziell für die Extraktion von Antworten aus Dokumentensammlungen entwickelt wurde und eine bessere Korrelation zwischen der Leistung bei den einzelnen Teilaufgaben und ihrer Kombination aufweist. Darüber hinaus ist die Wahl des Konfidenzmaßes wahrscheinlich problematisch, da die Konfidenz für jedes der k relevantesten Dokumente aus dem Retrieval-Schritt individuell berechnet wird.

Zusammenfassend zeigen die vorgestellten Ansätze eine deutliche Verbesserung der Leistung auf beiden Benchmarks im Vergleich zu den Ansätzen aus der Literatur. Darüber

hinaus legen die Ergebnisse den Einsatz von HTR-basierten Ansätzen für Datensätze mit niedriger CER und von HTR-freien Ansätzen für Datensätze mit hoher CER nahe.

Abbildung 6.17 zeigt vier qualitative Beispiele von Antworten des HTR-freien QA-Modells auf der Basis des BenthamQA-Benchmarks. Die qualitativen Ergebnisse demonstrieren, dass auch komplexe Fragen mit dem Verfahren korrekt beantwortet werden. Außerdem liegen die falsch vorhergesagten Bildbereiche verhältnismäßig nah an den Gold-Standard Annotationen im Dokument. Es ist jedoch offensichtlich, dass die vorhergesagten Dokumentenbereiche viele der Fragewörter enthalten. Daher ist die Frage berechtigt, ob das Modell den Kontext und die Frage semantisch versteht oder nur die Heuristik ausnutzt, dass Fragewörter häufig in der Nähe der Antwort vorkommen. Das letzte Beispiel widerspricht jedoch dieser Vermutung, da der vorhergesagte Ausschnitt zwei übereinstimmende Fragewörter enthält, während die Auswahl der zweiten und dritten Zeile im Dokumentenbild sechs übereinstimmende Fragewörter aufweist.

6.7 DISKUSSION

Die Ergebnisse des letzten Abschnitts zeigen, dass durch die Integration von externem semantischen Wissen HTR-freie Modelle bei den meisten semantischen Benchmarks ähnliche oder sogar bessere Ergebnisse erzielen können als HTR-basierte Ansätze. Diese Resultate bieten jedoch keine eindeutige Entscheidungshilfe, in welchem Szenario ein HTR-freies oder ein HTR-basiertes Verfahren zu bevorzugen ist. Im Folgenden wird unter anderem ein fairer Vergleich der beiden Ansätze durchgeführt und die Leistung der Verfahren in Abhängigkeit von der Zeichenfehlerrate ermittelt. Darüber hinaus wird die Robustheit sowie der Einfluss von Optimierungsschritten für HTR-freie Modelle analysiert und diskutiert.

6.7.1 *HTR-freie vs. HTR-basierte Wortbildeinbettung*

Ein direkter Vergleich zwischen dem HTR-freien und dem HTR-basierten Verfahren ist aufgrund der weitgehend unterschiedlichen semantischen Modelle nur bedingt möglich. Für eine faire Gegenüberstellung der Verfahren müssen möglichst gleiche Voraussetzungen vorliegen. Dazu wird in diesem Experiment der HTR-basierte Ansatz an die Gegebenheiten der HTR-freien Modelle angepasst. Die beiden Ansätze unterscheiden sich dann lediglich in der Bestimmung der ELMo-Repräsentationen für die Eingabebilder. Der HTR-freie Ansatz realisiert eine direkte Abbildung von Wortbildern mit einem neuronalen Modell. Das HTR-basierte Verfahren verwendet einen sequentiellen Ansatz. Dabei wird das gegebene Wortbild zunächst mit dem HTR-Modell aus Abschnitt 4.1.1 in ein maschinenlesbares Wort transformiert und anschließend die semantische Repräsentation mit dem ELMo-Modell bestimmt. Die jeweiligen HTR- und CNN-Modelle werden anhand derselben Daten trainiert. Zusätzlich wird ein Baseline-Ansatz mit einer fehlerfreien Vorhersage der semantischen ELMo-Einbettungen analysiert. Dieser Ansatz verwendet die Gold-Standard Textannotation des Wortbildes zur Bestimmung der Wortbildeinbettungen. Der Vergleich zwischen den Ansätzen erfolgt anhand der vorgestellten NER- und QA-Benchmarks. Für das QA wird nicht die vollständige Pipeline bewertet, sondern lediglich die Extraktion der Antwort auf Basis eines einzelnen Dokuments, da diese Bewertung repräsentativer für die semantische Qualität der Ansätze ist.

Die Ergebnisse in Tabelle 6.4 zeigen für alle getesteten NER-Benchmarks nur geringe Unterschiede zwischen der direkten und der sequentiellen Wortbildeinbettung. Die Leis-

Tabelle 6.4: Vergleich von direkten und sequentiellen Ansätzen zur Einbettung von Wortbildern auf mehreren semantischen Benchmarks. Die Ergebnisse werden für die NER-Ansätze in macro F1 [%] und für die QA-Modelle mit der Genauigkeit [%] angegeben.

	Benchmark	Annotation	Sequentiell	Direkt
NER	IAM (6)	85.7	72.1	74.6
	IAM (18)	63.4	46.9	51.0
	GW	84.5	81.9	83.8
	sGMB	81.1	75.2	76.9
QA	HWSQuAD	78.1	74.5	77.1
	BenthamQA	70.0	41.5	60.5

tung des HTR-freien Ansatzes ist jedoch auf allen Benchmarks höher. Bei den GW- und sGMB-Datensätzen unterscheidet sich die Leistung des direkten und des sequentiellen Ansatzes nur marginal, während bei den IAM-Benchmarks ein wesentlicher Unterschied zu beobachten ist. Dies ist vermutlich auf den bereits erwähnten Einfluss von HTR-Fehlern auf die Leistungsfähigkeit von NER-Modellen zurückzuführen. In diesem Zusammenhang weisen die HTR-Modelle für die GW- und sGMB-Datensätze eine CER von etwa 3% und für die IAM-DB von 7% auf. Bei identischen Voraussetzungen hinsichtlich der QA-Modelle kann der HTR-freie Ansatz den HTR-basierten Ansatz auf beiden QA-Benchmarks übertreffen. Insbesondere auf dem BenthamQA-Datensatz beträgt die absolute Verbesserung der direkten gegenüber der sequentiellen Wortbildeinbettung etwa 20 Prozent. Die Leistung des HTR-freien Ansatzes ist mit der des Orakel-Ansatzes vergleichbar. Dies deutet auf eine robuste Abbildung der Wortbilder hin. Bei der semantischen Schlüsselwortsuche unterscheidet sich das HTR-freie Verfahren in allen getesteten Benchmarks nur marginal vom HTR-basierten Ansatz oder erzielt sogar bessere Ergebnisse (siehe Tabelle 6.1). Die vergleichsweise geringe Leistung der HTR-basierten Modelle ist vermutlich darauf zurückzuführen, dass bereits kleine Fehler in der Texterkennung zu sehr unterschiedlichen Worteinbettungen führen können. Im Gegensatz zu den NER- und QA-Ansätzen lassen sich diese Fehler nicht von einem nachfolgenden Modell intern korrigieren.

Zusammenfassend verdeutlichen die Ergebnisse dieses Experiments die Vorteile von HTR-freien gegenüber HTR-basierten Ansätzen bei der Vorhersage semantischer Wortbildrepräsentationen. Es sei darauf hingewiesen, dass auf Grundlage der begrenzten Anzahl untersuchter Benchmarks die Schlussfolgerungen nicht zwangsläufig auf alle Datensätze übertragen werden können und weitere empirische Untersuchungen in diesem Bereich zur Bestätigung erforderlich sind.

6.7.2 *Analyse zur Robustheit von HTR-freien Modellen*

Das Hauptargument für die Verwendung HTR-freier Modelle ist ihre Robustheit gegenüber der hohen Variabilität von Handschriften. Diesbezüglich haben die bisherigen Ergebnisse bereits gezeigt, dass solche Modelle gegenüber HTR-basierten Verfahren insbesondere bei Benchmarks mit einer hohen Anzahl von Transkriptionsfehlern Vorteile aufweisen. Bei geringeren Fehlerraten sind die HTR-basierten Modelle jedoch im Allgemeinen leistungsfähiger. In diesem Experiment wird analysiert, ab welcher Fehlerrate bei der Texterkennung

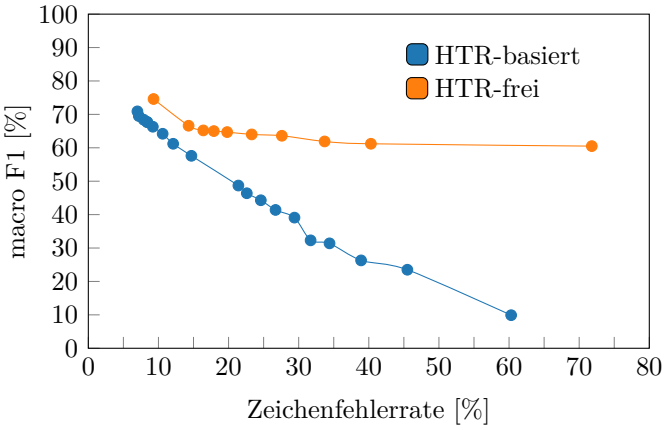


Abbildung 6.18: Bewertung der Robustheit von HTR-basierten und HTR-freien Ansätzen gegenüber HTR-Fehlern. Die Ergebnisse werden für beide Ansätze auf dem IAM (6)-Benchmark in Abhängigkeit von der CER dargestellt. Die HTR- und Worteinbettungsmodelle basieren auf unterschiedlichen Datenmengen der IAM-DB und weisen daher unterschiedliche CERs auf.

die Verwendung eines HTR-freien Ansatzes vorteilhaft ist. Dazu werden die Auswirkungen von Texterkennungsfehlern auf die Leistungsfähigkeit der beiden Ansätze anhand eines semantischen Benchmarks untersucht.

Auch wenn im HTR-freien Verfahren keine explizite Texterkennung durchgeführt wird, kann auf Basis der vektoriellen Wortbildeinbettungen eine lexikonbasierte Texterkennung realisiert werden. Dazu wird die vorhergesagte Elmo-Repräsentation eines Wortbildes mit allen Wortrepräsentationen aus einem vorgegebenen Wortlexikon verglichen und das Wort mit der höchsten Kosinusähnlichkeit als Transkription des Wortbildes verwendet. Das Lexikon besteht aus allen Trainings-, Validierungs- und Testwörtern des gegebenen Benchmark-Datensatzes. Zur Bestimmung der semantischen Leistung werden die klassischen HTR-freien und HTR-basierten Modelle verwendet. Beim HTR-freien Modell wird lediglich der merkmalsbasierte Ansatz durch den dualen Ansatz für das Training ersetzt, da dieser eine Ende-zu-Ende-Optimierung des Modells ermöglicht. Zur Generierung von HTR- und Wortbildeinbettungsmodellen mit unterschiedlichen Texterkennungsleistungen werden die Modelle auf einer variablen Anzahl von Trainingsdaten trainiert.

Die Leistung des HTR-freien und des HTR-basierten Modells wird anhand des IAM (6)-Benchmarks bewertet. Die Ergebnisse beider Ansätze sind in Abbildung 6.18 dargestellt. Bei beiden Ansätzen wirken sich Texterkennungsfehler negativ auf die Leistungsfähigkeit der Modelle aus. Insbesondere verdeutlichen die Ergebnisse den grundsätzlichen Nachteil sequentieller Ansätze, bei denen sich HTR-Fehler fortpflanzen und die Modellleistung in etwa umgekehrt proportional zur CER abnimmt. Der macro F1-Wert des HTR-freien Modells sinkt von etwa 75% auf 65%, wenn die Zeichenfehlerrate des Worteinbettungsmodells von 9% auf 15% steigt. Danach bleibt die Leistung nahezu konstant, selbst bei der Verwendung eines initialen Worteinbettungsmodells mit einer Zeichenfehlerrate von 71.8%. Dies ist weitgehend auf die Ende-zu-Ende-Architektur und die verwendete Optimierungsstrategie zurückzuführen.

Zusammenfassend zeigen die Ergebnisse dieser Analyse, dass bereits ab einer CER von etwa 15% das HTR-freie Modell bessere Ergebnisse erreicht als der HTR-basierte Ansatz. Obwohl es sich bei dem IAM-Benchmark um einen praxisnahen Datensatz für die

semantische Analyse von handschriftlichen Wortbildern handelt, kann die generelle Überlegenheit des HTR-freien Modells aufgrund der Verwendung von nur einem Benchmark in diesem Experiment nicht abschließend nachgewiesen werden. Unter Berücksichtigung der anwendungsspezifischen Auswertung des vorherigen Abschnitts können dennoch begründete Schlussfolgerungen für den Einsatz der Verfahren abgeleitet werden. Die in dieser Arbeit verwendeten Benchmarks weisen Zeichenfehlerraten von 0.5% bis 7% auf, wobei BenthamQA mit 19% einen Extremfall darstellt. Bei der semantischen Schlüsselwortsuche und der NER lassen sich bei den Benchmarks mit einer CER von bis zu 7% keine bedeutenden Unterschiede zwischen den beiden Ansätzen feststellen. Deutlich werden die Unterschiede jedoch bei der QA-Aufgabe. Hier ergeben sich wesentliche Vorteile für den HTR-basierten Ansatz auf dem HWSQuAD-Benchmark bei der Verwendung eines HTR-Modells mit einer CER von unter einem Prozent. Beim BenthamQA-Benchmark erzielt das HTR-basierte Modell bei Verwendung eines Texterkennungsmodells mit einer CER von circa 20% jedoch deutlich schlechtere Ergebnisse als der HTR-freie Ansatz. Aus den Resultaten ergibt sich die Empfehlung, dass HTR-freie Modelle ab einer CER von circa 15% verwendet werden sollten, während bei niedrigeren Fehlerraten HTR-basierte Verfahren in der Regel ähnliche oder bessere Ergebnisse liefern. Auch wenn die ermittelten Grenzwerte intuitiv sinnvoll erscheinen, kann die Allgemeingültigkeit dieser Empfehlung aufgrund der begrenzten Anzahl der untersuchten Benchmarks nicht zwangsläufig auf alle Datensätze übertragen werden.

6.7.3 *Einfluss von Wortbildeinbettungen auf die HTR-freie Modellleistung*

In dieser Arbeit werden mehrere Anpassungen des HTR-freien Ansatzes vorgestellt und die Parameter dieses Verfahrens anhand intrinsischer Metriken festgelegt. Obwohl in den meisten Publikationen im NLP-Bereich eine starke Korrelation zwischen intrinsischen und extrinsischen Metriken nachgewiesen wurde, kann diese nicht uneingeschränkt garantiert werden. Daher wird in diesem Experiment zum einen der Zusammenhang zwischen den verwendeten intrinsischen und extrinsischen Metriken überprüft und zum anderen eine Art Ablationsstudie (engl.: *Ablation Study*) für die in dieser Arbeit vorgestellten Adaptionsverfahren des HTR-freien Ansatzes durchgeführt. Insbesondere werden die Auswirkungen der vorgestellten Optimierungsschritte zur robusten semantischen Wortbildeinbettung untersucht. Diese umfassen die Hinzunahme von synthetischen Daten bei der Wissensdestillation und die geeignete Kombination von semantischen und syntaktischen Wortbildeinbettungen. Darüber hinaus wird die Wahl der intrinsisch festgelegten ELMo-Einbettung als semantische Wortbildrepräsentation für das HTR-freie Modell evaluiert. Die Auswahl der Wortbildeinbettung ist einer der grundlegendsten Parameter für den Ansatz dieser Arbeit. Zur Bewertung der Wahl wird in diesem Experiment der Einfluss ausgewählter Wortbildeinbettungen auf die Leistung des HTR-freien Ansatzes extrinsisch evaluiert. Ein grundlegender Vergleich wird durch die Verwendung einer zufälligen Initialisierung des Wortbildeinbettungsmodells erreicht, da dies das Standardverfahren für Ende-zu-Ende-Ansätze darstellt. Um den Einfluss der semantischen Qualität der ELMo-Einbettung zu bestimmen, wird mit der HWNet-Repräsentation eine syntaktische Wortbildeinbettung evaluiert. Dabei werden im HWNet Wortbilder mit ähnlichen textuellen Annotationen auf ähnliche Vektorrepräsentationen abgebildet. Aufgrund der häufigen Verwendung des FastText-Verfahrens als semantische Wortbildrepräsentation in verwandten Arbeiten stellt diese Darstellung einen interessanten Vergleich zur ELMo-Repräsentation dar. Zur Realisierung des Experiments wird das Wortbildeinbettungsmodell des HTR-freien Ansatzes an die Vorhersage der Re-

Tabelle 6.5: Evaluierung des Einflusses verschiedener Wortbildrepräsentationen auf die Leistung des HTR-freien Ansatzes. Neben den Wortbildrepräsentationen wird auch die Hinzunahme von synthetisch generierten Wortbildern in das Training evaluiert. Die Ergebnisse werden für die vier NER-Benchmarks in macro F1 [%] angegeben.

Wortbildrepräsentation	+ Synth	IAM (6)	IAM (18)	GW	sGMB
Zufällig		32.8	15.5	52.2	37.8
HWNet		60.3	36.5	72.8	71.2
FastText		53.6	32.3	68.2	66.9
ELMo		63.9	40.2	72.1	70.2
ELMo	✓	74.6	51.0	83.8	76.9
ELMo + HWNet	✓	72.5	47.3	83.6	73.5

präsentationen angepasst. Für alle Repräsentationen außer der zufälligen Initialisierung wird das Einbettungsmodell mit der cross-modalen Wissensdestillation auf die Vorhersage der gegebenen Einbettungsverfahren vortrainiert und anschließend das komplette HTR-freie Modell auf mehreren NER-Benchmarks angepasst und evaluiert.

Die Ergebnisse der Experimente sind in Tabelle 6.5 dargestellt. Insgesamt zeigt sich für alle Benchmarks der gleiche Trend hinsichtlich der Wortbildeinbettungen, wobei sich die Leistungen der Modelle auf den Benchmarks erheblich unterscheiden. Die Ergebnisse verdeutlichen insbesondere die Vorteile der Verwendung von vortrainierten Repräsentationen gegenüber einer zufälligen Initialisierung des Wortbildeinbettungsmodells. Hierbei wird die Leistung auf allen Benchmarks bei geeigneter Wahl der Wortbildrepräsentation gegenüber dem klassischen Ende-zu-Ende-Verfahren nahezu verdoppelt. Die Ergebnisse zeigen zudem, dass es nicht ausreicht, semantische Informationen zu kodieren, sondern die Vorhersagequalität der Repräsentationen mindestens ebenso wichtig ist. Beispielsweise erzielt die syntaktische HWNet-Repräsentation in allen Benchmarks bessere Ergebnisse als die semantische FastText-Einbettung, obwohl die HWNet-Repräsentation im Vergleich zu FastText praktisch keine semantischen Beziehungen kodiert. Die ELMo-Repräsentation erzielt auf allen Benchmarks die besten Ergebnisse. Dies ist vermutlich auf die hohe semantische Qualität sowie die hohe Vorhersagefähigkeit auf handgeschriebenen Wortbildern zurückzuführen. Insbesondere das Vortraining des Wortbildeinbettungsmodells mit synthetischen Wortbildern führt zu einer wesentlichen Leistungssteigerung. Entgegen den Erwartungen führt die Kombination von semantischen und syntaktischen Einbettungen in allen Benchmarks zu einem marginalen Leistungsverlust.

Basierend auf den Ergebnissen kann die Wahl der intrinsisch bestimmten Parameter des cross-modalen Destillationsverfahrens als geeignet betrachtet werden. Insgesamt weisen die Ergebnisse eine hohe Korrelation zwischen den extrinsischen und intrinsischen Evaluationen auf. Dabei korreliert die Leistung der Wortbildeinbettungsverfahren auf den WA- und QbS-Benchmarks mit der Leistung der entsprechenden HTR-freien Modelle auf den NER-Benchmarks. Die Leistungsfähigkeit der NER-Ansätze kann jedoch nicht anhand einer einzelnen intrinsischen Metrik bestimmt werden, sondern erfordert eine gemeinsame Betrachtung der QbS- und WA-Werte.

7 FAZIT

Dieses Kapitel bietet eine Zusammenfassung der Arbeit und gibt einen Ausblick auf weitere Forschungsarbeiten im Bereich der semantischen Analyse von handschriftlichen Dokumentenbildern. Dazu werden im Abschnitt 7.1 zunächst die zentralen methodischen Ansätze, Ergebnisse und Erkenntnisse dieser Arbeit zusammengefasst und die erzielten Beiträge im Forschungskontext eingeordnet. Im Abschnitt 7.2 werden anschließend die Limitierungen der vorgestellten Ansätze diskutiert und ein Ausblick auf zukünftige Forschungsarbeiten in diesem Bereich gegeben.

7.1 ZUSAMMENFASSUNG

Die semantische Analyse von handschriftlichen Dokumentenbildern ist aufgrund der Kombination von visuellen und textuellen Eigenschaften sowie der hohen Variabilität von Handschriften eine anspruchsvolle Anwendung. In diesem Zusammenhang stellt sich zunächst die Forschungsfrage, ob die sequentielle Kombination eines Handschrifterkenners und eines textuellen NLP-Systems einen geeigneten Lösungsansatz für diese Aufgabenstellung bietet. Dazu wird in dieser Arbeit ein HTR-basierter Ansatz entwickelt und die Anfälligkeit der NLP-Modelle gegenüber Texterkennungsfehlern evaluiert. Die Ergebnisse dieser Evaluierung zeigen einen erheblichen negativen Einfluss von Texterkennungsfehlern auf die Leistungsfähigkeit semantischer Ansätze. Folglich ist eine einfache Kombination eines HTR- und eines NLP-Modells aufgrund der Fehlerfortpflanzung suboptimal. Um dieses Problem zu vermeiden, wird in dieser Arbeit ein HTR-freies Ende-zu-Ende-Modell vorgestellt. Dies ist ein robusteres Verfahren, welches eine explizite Texterkennung vermeidet. Obwohl dieses Modell das Problem der Fehlerfortpflanzung theoretisch lösen kann, erzielt es auf den meisten semantischen Benchmarks deutlich schlechtere Ergebnisse als der HTR-basierte Ansatz.

Das Fehlen von vortrainierten semantischen Worteinbettungen wird in dieser Arbeit als ein Hauptproblem von HTR-freien Ansätzen identifiziert. Ein zentraler Forschungsbeitrag dieser Arbeit ist die Integration von externem semantischen Wissen in HTR-freie Modelle und die Beantwortung der Forschungsfrage, ob durch diese Integration die bestehende Diskrepanz zu HTR-basierten Ansätzen behoben werden kann. Dazu wird in dieser Arbeit ein Verfahren zur cross-modalen Wissensdestillation vorgestellt, das semantisches Wissen aus textuell vortrainierten Worteinbettungsmodellen effizient in HTR-freie Modelle integriert. Dieses Verfahren basiert auf einer robusten Abbildung von handschriftlichen Wortbildern in einen textuell vortrainierten semantischen Worteinbettungsraum mit einem neuronalen Faltungsnetzwerk. Ein wesentlicher Parameter dieses Destillationsansatzes ist die Wahl eines geeigneten textuellen semantischen Worteinbettungsverfahrens als Lehrermodell. Die in diesem Zusammenhang durchgeführten Experimente zeigen, dass die semantische Qualität der Worteinbettungen aus dem Lehrermodell alleine nicht ausreicht, sondern deren Vorhersagefähigkeit auf Basis von Wortbildern mindestens ebenso wichtig ist. Auf den Benchmarks dieser Arbeit bietet die ELMo-Einbettung den besten Kompromiss zwischen der semantischen Qualität der Repräsentationen und ihrer Eignung für die Wortbildvorhersage. Eine Limitierung dieses Destillationsansatzes ergibt sich für

Wortbilder, deren textuelle Annotationen nicht im Training des Destillationsverfahrens berücksichtigt wurden. Um dieses Problem zu reduzieren, werden in dieser Arbeit Optimierungsverfahren zur robusteren Abbildung von Wortbildern in textuelle semantische Vektorräume vorgestellt. Ein Ansatz ist die annotationsfreie Destillationsstrategie, welche synthetisch erzeugte Wortbilder verwendet, um die Menge der nicht im Training vorkommenden Wörter zu minimieren. Dabei führt die Hinzunahme von synthetisch generierten Wortbildern beim Destillationsprozess zu einer deutlichen Leistungssteigerung auf allen in dieser Arbeit betrachteten Benchmarks. Ein weiterer vorgestellter Lösungsansatz ist die geeignete Kombination von semantischen und syntaktischen Worteinbettungen, sodass im Falle einer fehlerhaften semantischen Vorhersage zumindest die orthographische Information des im Wortbild enthaltenen Textes repräsentiert wird. Entgegen den Erwartungen führt diese Kombination jedoch auf allen Benchmarks zu marginalen Leistungsverlusten. Insgesamt wird durch die Optimierungsschritte eine deutlich robustere Einbettung erreicht, jedoch bleibt die Abbildung handschriftlicher Wortbilder in einen textuell vortrainierten semantischen Worteinbettungsraum eine anspruchsvolle Aufgabenstellung und stellt einen zentralen Schwachpunkt dieses Destillationsansatzes dar.

Die Leistungsfähigkeit der optimierten HTR-freien und HTR-basierten Modelle wird anhand semantischer Benchmarks für die NER, das QA und die semantische Schlüsselwortsuche evaluiert. Insgesamt unterscheiden sich die Leistungen der beiden Ansätze auf den meisten Benchmarks nur marginal. Hierbei erreicht der HTR-basierte Ansatz im Allgemeinen bessere Resultate auf Benchmarks mit einer niedrigen Anzahl an Texterkennungsfehlern, während der HTR-freie Ansatz auf Benchmarks mit einer hohen Fehlerrate im Allgemeinen besser ist. Unter Verwendung der gleichen Voraussetzungen erreicht das HTR-freie im Vergleich zum HTR-basierten Verfahren auf den semantischen Benchmarks dieser Arbeit höhere Leistungen. Darüber hinaus weist der HTR-freie Ansatz eine verbesserte Robustheit gegenüber Texterkennungsfehlern auf. Dabei nimmt die Leistungsfähigkeit des HTR-basierten Ansatzes etwa umgekehrt proportional zur CER ab, während die Leistung der HTR-freien Modelle zwar bis zu einer CER von etwa 10% sinkt, danach aber weitgehend konstant bleibt. Durch das Vortraining des HTR-freien Modells mit der cross-modalen Wissensdestillation wird die Leistung auf allen Benchmarks dieser Arbeit gegenüber dem klassischen Ende-zu-Ende-Verfahren aus der Literatur nahezu verdoppelt. Obwohl eine generelle Empfehlung zur Auswahl der Modelle anhand genauer Kriterien nicht möglich ist, zeigen die Ergebnisse, dass HTR-freie Modelle insbesondere bei Benchmarks ab einer Zeichenfehlerrate der Eingabedaten von 15% Vorteile gegenüber HTR-basierten Verfahren aufweisen. Bei niedrigeren Fehlerraten sind die HTR-basierten Modelle jedoch generell leistungsfähiger. Die Ergebnisse deuten zudem darauf hin, dass die Integration semantischer Informationen mit dem cross-modalen Destillationsansatz aktuell Vorteile gegenüber dem selbstüberwachten Lernen dieser Informationen auf der Basis von Transformern und großen Korpora von handschriftlichen Dokumentenbildern hat.

Zusammenfassend stellt der vorgestellte Ansatz zur cross-modalen Wissensdestillation eine interessante Alternative zu dem ressourcenintensiven selbstüberwachten Lernverfahren dar. Durch die Integration externer Informationen in das HTR-freie Modell wird der Abstand zu HTR-basierten Ansätzen bei den meisten semantischen Benchmarks dieser Arbeit deutlich verringert oder sogar übertroffen. Damit bietet das Verfahren im Gegensatz zu den klassischen HTR-freien Modellen aus der Literatur eine echte Alternative zu HTR-basierten Ansätzen.

7.2 AUSBLICK

Die in dieser Arbeit vorgestellten Ansätze erzielen im Allgemeinen eine akzeptable Qualität bei der semantischen Analyse von handschriftlichen Dokumentenbildern. Im Vergleich zu NLP-Modellen, die auf den textuellen Gold-Standard Annotationen der Dokumentenbilder basieren, besteht jedoch weiterhin ein erheblicher Optimierungsbedarf. Für die praktische Anwendung weisen die vorgestellten Ansätze zudem einige Einschränkungen auf und bieten zahlreiche Möglichkeiten für zukünftige Forschungsarbeiten.

Ein kritischer Aspekt, der im Rahmen dieser Arbeit nicht behandelt wird, ist die Frage, wie die vorgestellten Ansätze effektiv in Situationen eingesetzt werden können, in denen keine Wortsegmentierung für die Eingabedaten zur Verfügung steht. Dies ist insbesondere für den Einsatz in realen Anwendungen von großer Bedeutung, da die manuelle Erstellung von Segmentierungen auf Wortebene bei handschriftlichen Dokumentenbildern sehr aufwändig ist und die automatische Extraktion dieser Daten trotz großer Fortschritte häufig fehlerbehaftet ist. Eine zentrale zukünftige Forschungsfrage in diesem Bereich ist daher, wie sich Segmentierungsfehler auf die Leistungsfähigkeit der vorgestellten Modelle auswirken. Ein vielversprechender Lösungsansatz ist die konzeptionelle Erweiterung der vorgestellten Verfahren von der Wort- auf die Zeilen- oder sogar Dokumentenebene, wobei die Segmentierung der Wortbilder dann implizit erfolgt. Der Verzicht auf Gold-Standard segmentierte Wortbilder als Eingabe für die Modelle bedeutet dabei nicht notwendigerweise eine Verschlechterung der Systemleistung, sondern kann durch die Berücksichtigung von Kontextinformationen potenziell sogar zu einer Leistungssteigerung führen.

Ungeachtet der in dieser Arbeit erzielten Fortschritte bleibt die Abbildung von handschriftlichen Wortbildern in einen textuell vortrainierten semantischen Worteinbettungsraum eine anspruchsvolle Aufgabe und ist die zentrale Schwachstelle des in dieser Arbeit vorgestellten Destillationsansatzes. Insbesondere ist die isolierte Betrachtung eines Wortbildes zur Vorhersage seiner semantischen Repräsentation problematisch. Eine Weiterentwicklung des cross-modalen Destillationsverfahrens, das den Kontext des Wortbildes zur Vorhersage seiner semantischen Repräsentation berücksichtigt, erscheint daher als vielversprechender Optimierungsschritt. Eine weitere potentielle Leistungsverbesserung des HTR-freien Modells ergibt sich aus dem Umstieg von statischen auf kontextbasierten Wortbildeinbettungen, da letztere das Problem der Worthomonymie lösen können und zu signifikanten Leistungsverbesserungen im NLP-Bereich führen. Eine weitere Ausbaustufe des HTR-freien Modells ist die Berücksichtigung bzw. Kodierung der Positionsinformationen von Wortbildern. Diese Informationen sind zwar für die in dieser Arbeit betrachteten Benchmarks von geringer Relevanz, es existieren jedoch zahlreiche reale Anwendungsfälle, in denen Strukturinformationen von grundlegender Bedeutung sind, wie z.B. die Informationsextraktion aus teilstrukturierten Dokumenten.

Insgesamt stellt die semantische Analyse von handschriftlichen Dokumentenbildern ein höchst relevantes und zugleich herausforderndes Forschungsthema mit vielen offenen Forschungsfragen dar. Die Forschung auf diesem Gebiet entwickelt sich gegenwärtig rasant und wird zukünftig sicherlich eines der wichtigsten Forschungsthemen im Bereich der Dokumentenanalyse darstellen. Neben dem in dieser Arbeit vorgestellten cross-modalen Destillationsverfahren existiert in der Literatur ein vielversprechender Ansatz zur HTR-freien Dokumentenbildanalyse, der bereits auf diversen Benchmarks zu state-of-the-art Ergebnissen führt. Dieser Ansatz basiert auf multimodalen Sprachmodellen, die mittels selbstüberwachtem Lernen auf großen Bild- und Textkorpora vortrainiert werden und eine textuelle Ausgabe bieten. Aufgrund der Textausgabe sind die Verfahren sehr flexibel und können

zahlreiche Anwendungen mit demselben Modell lösen. Allerdings weist dieses Verfahren aktuell eine Vielzahl von Herausforderungen auf und erfordert zudem einen hohen Ressourcenaufwand, der nur von wenigen Akteuren realisiert werden kann. Darüber hinaus gibt es bisher nur wenige Erkenntnisse und Ergebnisse zu multimodalen Sprachmodellen auf Datensätzen mit handgeschriebenen Dokumentbildern, sodass die Leistungsfähigkeit und Anwendbarkeit der Modelle in der Handschriftdomäne noch unter Beweis gestellt werden muss. Aufgrund des erheblichen Ressourcenbedarfs und den Herausforderungen multimodaler Sprachmodelle, wie z.B. Halluzinationen, wird es auch in Zukunft notwendig sein, erprobte und ressourceneffiziente Techniken für den Einsatz in sicherheitskritischen und lokalen Umgebungen zu entwickeln. In diesem Zusammenhang wird der multimodalen Wissensdestillation voraussichtlich eine Schlüsselrolle zukommen.

A VERÖFFENTLICHUNGEN DES AUTORS

In diesem Kapitel werden die Publikationen des Autors dieser Dissertation vorgestellt. Die wesentlichen Beiträge der vorliegenden Arbeit wurden bereits in Journals sowie auf renommierten Konferenzen und Workshops im Bereich der Dokumentenanalyse veröffentlicht. Die Veröffentlichungen wurden in einem *double-blind* (Konferenzen) bzw. *single-blind* (Journal und Workshops) Review-Prozess von mehreren Experten auf dem Gebiet der Dokumentenanalyse begutachtet und bewertet. Die Publikationen sind in der Reihenfolge ihres Erscheinungsdatums aufgelistet.

Literatureintrag: [214]

Oliver Tüselmann, Fabian Wolf und Gernot A. Fink. "Identifying and Tackling Key Challenges in Semantic Word Spotting" In: *Proc. Int. Conf. on Frontiers in Handwriting Recognition*. 2020, pp. 55-60

In dieser Arbeit werden die grundlegenden Herausforderungen und Probleme im Bereich der semantischen Schlüsselwortsuche vorgestellt und Ansätze zu deren Lösung präsentiert. Die Einbettung von Wortbildern in einen textuellen semantischen Worteinbettungsraum hat sich als der derzeit beste Ansatz zur Realisierung eines semantischen Retrievals herausgestellt. Dabei ist die Wahl des semantischen Worteinbettungsraums von grundlegender Bedeutung für die Qualität des Ansatzes. Im Rahmen dieser Arbeit wird gezeigt, dass die FastText-Repräsentation sowohl theoretische als auch praktische Eigenschaften für die semantische Wortbilddarstellung aufweist und damit die semantische Qualität des Retrievals im Vergleich zur ersten Publikation in diesem Bereich deutlich verbessern kann. Neben der Modifikation der Worteinbettung wird in dieser Arbeit eine optimierte zweistufige Architektur vorgestellt. Diese besteht aus einer Kombination des *TPP-PHOCNets* [191] und einem MLP. Dabei wird das eingegebene handschriftliche Wortbild zunächst in eine PHOC-Repräsentation überführt und anschließend mit dem MLP in eine FastText-Repräsentation umgewandelt. Eine wichtige Erkenntnis aus dieser Arbeit ist, dass sich die mAP nicht für die Bewertung von semantischen Retrieval-Listen eignet. Zur adäquaten Bewertung der semantischen Qualität von Retrieval-Listen wird daher eine Anpassung der Metrik präsentiert.

Literatureintrag: [215]

Oliver Tüselmann, Fabian Wolf und Gernot A. Fink. "Are End-to-End Systems Really Necessary for NER on Handwritten Document Images?" In: *Proc. Int. Conf. on Document Analysis and Recognition*. 2021, pp. 808-822

Im Bereich der semantischen Analyse von handschriftlichen Dokumentenbildern wird überwiegend davon ausgegangen, dass Ende-zu-Ende-Ansätze den sequentiellen Verfahren aus HTR- und NLP-Modellen vorzuziehen sind. Dies basiert auf der Annahme, dass die Handschrifterkennung zu fehlerhaft ist, um eine effektive Anwendung von textuellen NLP-Methoden zu ermöglichen. In dieser Arbeit wird ein zweistufiger NER-Ansatz vorgestellt

und mit den Ende-zu-Ende-Ansätzen aus der Literatur verglichen. Der Ansatz kombiniert einen aktuellen Handschrifterkenner [87] mit einem textuellen NER-Modell aus dem NLP-Bereich [4]. Aufgrund des Mangels an Datensätzen und Bewertungsprotokollen ist ein solcher Vergleich derzeit schwierig. Hierbei existieren im Forschungsgebiet der NER auf handschriftlichen Dokumentenbildern lediglich Datensätze, die entweder teilstrukturierte Dokumente enthalten oder ausschließlich privat zugänglich sind. Daher wurden die bekannten IAM- und George Washington-Datensätze manuell mit NE-Annotationen von dem Autor dieser Dissertation erweitert und zusammen mit einer optimierten Aufteilung der Daten und einem Evaluierungsprotokoll in dieser Arbeit veröffentlicht. Entgegen der allgemeinen Annahme, zeigen die Experimente, dass das zweistufige Modell im Vergleich zu den Ende-zu-Ende-Ansätzen auf allen getesteten NER-Benchmarks bessere Leistungen erzielt.

Literatureintrag: [211]

Oliver Tüselmann und Gernot A. Fink. "Named Entity Linking on Handwritten Document Images" In: *Proc. Int. Workshop on Document Analysis Systems*. 2022, pp. 199-213

Named Entity Linking (NEL) identifiziert Erwähnungen von NEs in einem Text und verknüpft diese mit Einträgen in einer vorgegebenen Wissensdatenbank. Für diese semantische Aufgabe wurden im Bereich der Dokumentenanalyse bisher nur Verfahren für maschinell gedruckte Dokumentenbilder entwickelt und evaluiert. Die Nichtberücksichtigung handschriftlicher Daten ist vermutlich auf den Mangel an NEL-Datensätzen für handschriftliche Dokumentenbilder zurückzuführen. Um diese Lücke zu schließen, werden in dieser Arbeit manuelle NEL-Annotationen für die bekannten IAM- und George Washington-Datensätze sowie ein synthetisch generierter handschriftlicher NEL-Datensatz auf Basis des AIDA-CoNLL-Benchmarks vorgestellt und veröffentlicht. Außerdem werden ein Evaluierungsprotokoll und ein HTR-basierter Ansatz zur Durchführung dieser Aufgabe vorgestellt. Der in dieser Arbeit verwendete Ansatz kombiniert einen aktuellen Handschrifterkenner [87] mit einem textuellen NEL-Modell [77]. Der vorgestellte HTR-basierte Ansatz erzielt vielversprechende Ergebnisse. Jedoch verdeutlichen die Resultate auch die Komplexität der NEL-Aufgabe auf Basis von handschriftlichen Dokumentenbildern.

Literatureintrag: [210]

Oliver Tüselmann, Kai Brandenbusch, Miao Chen und Gernot A. Fink. "A Weighted Combination of Semantic and Syntactic Word Image Representations" In: *Proc. Int. Conf. on Frontiers in Handwriting Recognition*. 2022, pp.285-299

Die Vorhersage semantischer Wortbildrepräsentationen mit neuronalen Faltungsnetzwerken ist insbesondere für solche Wortbilder problematisch, für die im Training des Modells keine Wortbilder mit gleicher Textannotation vorkamen. In [98] wird dieses Problem durch die Verwendung einer Kombination von semantischen und syntaktischen Wortbildrepräsentationen reduziert. In dieser Arbeit wird zunächst gezeigt, dass eine einfache Konkatenation der syntaktischen und semantischen Repräsentationen aufgrund ihrer stark unterschiedlichen Eigenschaften (z.B. Dimensionen und Dynamikbereiche) suboptimal ist. Anschließend wird ein Verfahren vorgestellt, das eine geeignete Kombination der beiden Repräsentationen ermöglicht. Dazu werden die Statistiken der beiden Darstellungen vor

deren Konkatenation durch Normalisierungs- und Standardisierungstechniken angeglichen. Zudem werden zwei Verfahren zur gewichteten Kombination der semantischen und syntaktischen Repräsentation vorgestellt und anhand der semantischen Schlüsselwortsuche sowohl qualitativ als auch quantitativ evaluiert. Die gewichtete Kombination der Einbettungen ermöglicht es Nutzern, sich mehr auf semantische oder syntaktische Aspekte bei der Wortsuche zu fokussieren und so neue Einblicke in Dokumentensammlungen zu erhalten.

Literatureintrag: [213]

Oliver Tüselmann, Friedrich Müller, Fabian Wolf und Gernot A. Fink. "Recognition-free Question Answering on Handwritten Document Collections" In: *Proc. Int. Conf. on Frontiers in Handwriting Recognition.* 2022, pp. 259-273

In dieser Arbeit wird ein HTR-freier Ansatz für das QA auf handschriftlichen Dokumentensammlungen vorgestellt. Aus Effizienzgründen basiert das Verfahren auf einer Kombination aus einem Retrieval- und einem QA-Modell. Dabei werden zunächst für eine gegebene Frage die Dokumentenbilder der Sammlung durch ein selbstkonstruiertes HTR-freies Retrieval-Modell auf wenige relevante Dokumente reduziert. Anschließend werden die relevanten Dokumentenbilder separat durch ein HTR-freies QA-Modell verarbeitet und für jedes dieser Dokumente eine Antwort mit einem zugehörigen Konfidenzwert ausgegeben. Aufgrund der hohen Variabilität von Handschriften ist das Ausgabeformat des Modells ein Bildausschnitt auf Zeilenebene und ermöglicht damit ein robusteres Ausgabeformat als klassische NLP-Verfahren. Der Hauptbeitrag dieser Arbeit ist die Entwicklung eines HTR-freien QA-Modells, das auf einer angepassten Variante des BIDAF-Modells aus dem NLP-Bereich basiert. Insgesamt erzielt das vorgestellte Verfahren im Vergleich zu HTR-freien Modellen aus der Literatur deutlich bessere Ergebnisse auf den anspruchsvollen BenthamQA- und HWSQuAD-Benchmarks.

Literatureintrag: [212]

Oliver Tüselmann und Gernot A. Fink. "Exploring Semantic Word Representations for Recognition-free NLP on Handwritten Document Images" In: *Proc. Int. Conf. on Document Analysis and Recognition.* 2023, pp. 85-100

Im Kontext der semantischen Analyse von handschriftlichen Dokumentenbildern haben sich HTR-freie Ansätze als robust erwiesen, liefern aber oft schlechtere Ergebnisse als HTR-basierte Methoden. Ein Hauptgrund dafür ist vermutlich, dass HTR-freie Ansätze nicht auf vortrainierte semantische Worteinbettungen zurückgreifen, die sich als eine der leistungsfähigsten Techniken im textuellen Bereich erwiesen haben. Um diese Einschränkung zu überwinden, werden in dieser Arbeit textuell vortrainierte semantische Worteinbettungsverfahren aus dem NLP-Bereich auf ihre Eignung zur semantischen Repräsentation für handschriftliche Wortbilder evaluiert. Hierbei wird eine HTR-freie Vorhersage der Wortrepräsentationen auf Basis handschriftlicher Wortbilder mit einem neuronalen Faltungsnetzwerk realisiert. Die Extraktion statischer Wortrepräsentationen aus kontextabhängigen semantischen Worteinbettungsverfahren (z.B. BERT [42]) erzielt auf allen getesteten Benchmarks state-of-the-art Ergebnisse. Zudem wird die Relevanz von semantischem Vorwissen für die Leistungsfähigkeit von HTR-freien Modellen zur semantischen Analyse von handschriftlichen Dokumentenbildern aufgezeigt. Der vorgestellte HTR-freie Ansatz mit externen semantischen Informationen erzielt im Vergleich zu HTR-freien und

HTR-basierten Ansätzen aus der Literatur bessere Ergebnisse auf allen getesteten NER-Benchmarks.

Literatureintrag: [216]

Oliver Tüselmann und Gernot A. Fink. “Neural Models for Semantic Analysis of Handwritten Document Images” In: *Int. Journal on Document Analysis and Recognition*. 2024

Diese Arbeit fasst die zuvor veröffentlichten Methoden und Ergebnisse zusammen und kann als kompakte Zusammenfassung dieser Dissertation betrachtet werden. Der Journalartikel stellt ein HTR-freies und ein HTR-basiertes System zur semantischen Analyse von handgeschriebenen Dokumentenbildern vor. Diese Systeme werden anhand mehrerer Benchmarks aus dem Bereich der wortsegmentierten handgeschriebenen Dokumentenbilder verglichen, darunter die semantische Schlüsselwortsuche, die NER und das QA. Darüber hinaus wird ein cross-modales Wissensdestillationsverfahren zur HTR-freien Integration von vortrainiertem semantischen Wissen aus der textuellen Domäne in die visuelle Domäne vorgestellt und evaluiert. Des Weiteren werden in einer Reihe von Experimenten Strategien zur Optimierung einer robusten semantischen Wortbildrepräsentation untersucht. Es wird gezeigt, dass die Integration von semantischem Wissen für HTR-freie Ansätze vorteilhaft ist, um für eine Vielzahl von Benchmarks Ergebnisse auf dem Stand der Technik zu erzielen.

B Weiterführende intrinsische Ergebnisse

Aus Gründen der Übersichtlichkeit werden die Auswertungen im Abschnitt 6.5 nur für den repräsentativen IAM-Benchmark durchgeführt. In diesem Abschnitt werden die Ergebnisse für die übrigen Datensätze dieser Arbeit vorgestellt. Dabei werden neben den Auswirkungen ausgewählter Worteinbettungsmethoden als Lehrermodell auch die Effekte für die Hinzunahme synthetisch generierter Wortbilder während des Destillationsprozesses präsentiert. Zusätzlich werden die Ergebnisse für die Kombination einer semantischen und syntaktischen Worteinbettung für die Datensätze dargestellt.

B.1 Analyse semantischer Worteinbettungsverfahren

Ein zentraler Parameter für die cross-modale Wissensdestillation ist die Wahl eines geeigneten Lehrermodells. Neben der semantischen Qualität der Worteinbettungen ist in dieser Arbeit insbesondere deren Eignung für die Vorhersage auf Basis handschriftlicher Wortbilder von Relevanz. Die semantische Qualität wird für die vorhergesagten Repräsentationen der Wortbilder mit der WA und die Abweichung von der Zielrepräsentation mit der Schlüsselwortsuche bewertet. Die folgenden Diagramme zeigen die Ergebnisse für die Datensätze George Washington, BenthamQA, HWSQuAD und sGMB. Insgesamt unterstützen die Resultate die Schlussfolgerungen aus der IAM-DB. Die ELMo-Einbettung bietet den besten Kompromiss zwischen der semantischen Qualität der Repräsentationen und ihrer Eignung für die Wortbildvorhersage auf allen Benchmarks. Detaillierte Informationen zu den Worteinbettungen und zur Durchführung der Experimente sind im Abschnitt 6.5.1 aufgeführt.

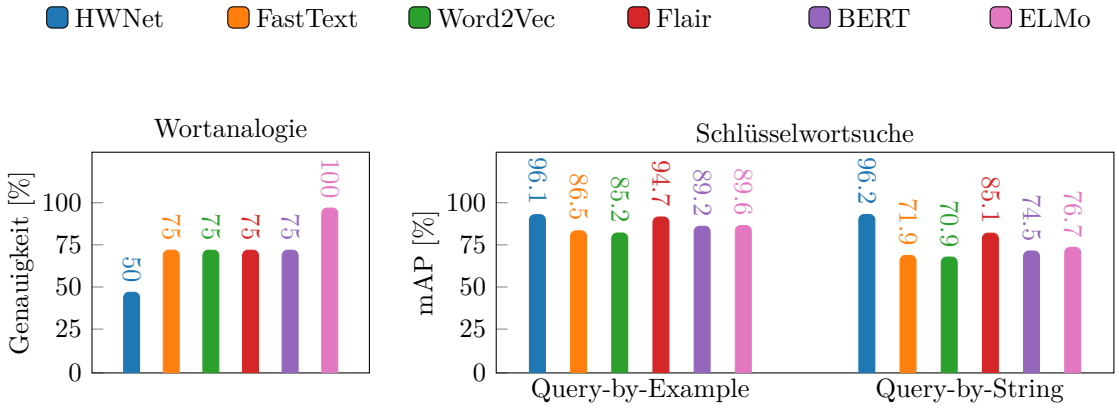


Abbildung B.1: Die Leistung der Worteinbettungsverfahren auf dem George Washington-Datensatz unter Verwendung der Wortanalogie und der QbE- bzw. QbS-basierten Schlüsselwortsuche.

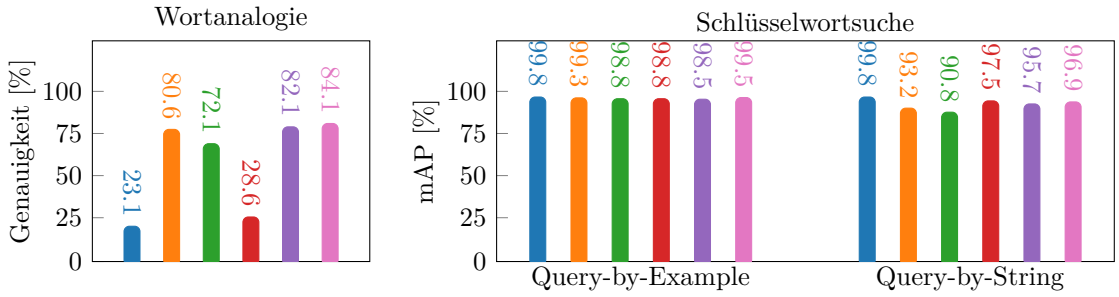


Abbildung B.2: Die Leistung der Worteinbettungsverfahren auf dem HWSQuAD-Datensatz unter Verwendung der Wortanalogie und der QbE- bzw. QbS-basierten Schlüsselwortsuche.

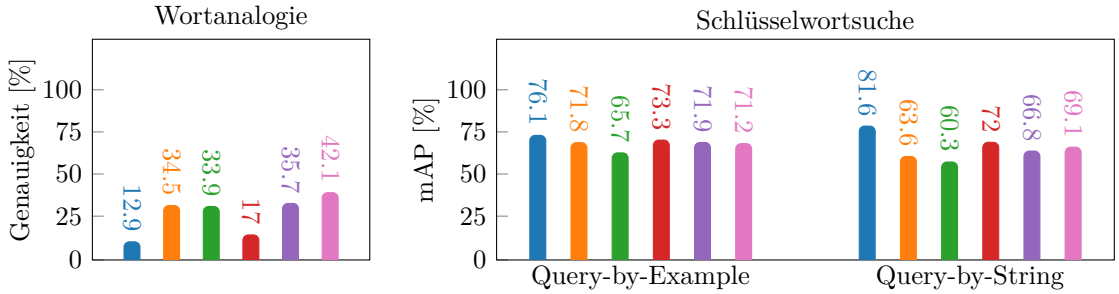


Abbildung B.3: Die Leistung der Worteinbettungsverfahren auf dem BenthamQA-Datensatz unter Verwendung der Wortanalogie und der QbE- bzw. QbS-basierten Schlüsselwortsuche.

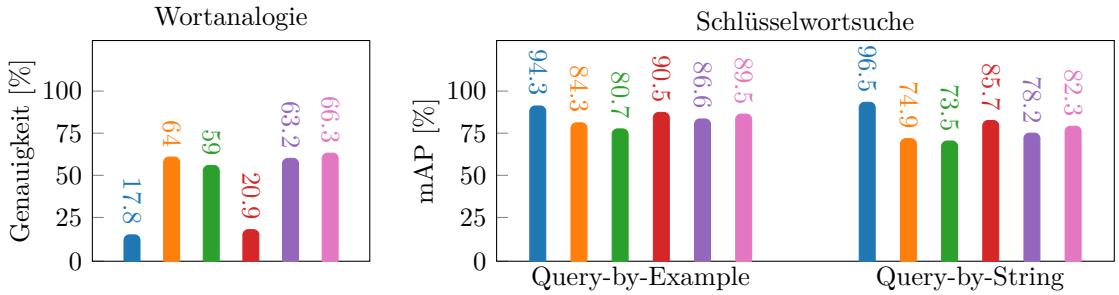


Abbildung B.4: Die Leistung der Worteinbettungsverfahren auf dem sGMB-Datensatz unter Verwendung der Wortanalogie und der QbE- bzw. QbS-basierten Schlüsselwortsuche.

B.2 EVALUATION DER ANNOTATIONSFREIEN WISSENSDESTILLATION

Die Diagramme in Abbildung B.5 zeigen die prozentuale Abdeckung der Anfragewörter aus der QbS-basierten Schlüsselwortsuche bzw. der Zielwörter aus der Wortanalogie mit den Wörtern der Trainingsmenge in Abhängigkeit von der Lexikongröße. Der Wert 0 auf der x-Achse repräsentiert in den Diagrammen die Anzahl der Wörter aus den Trainingsdaten des jeweiligen Benchmarks und jeder Wert $x > 0$ die Hinzunahme der $x \cdot 1000$ häufigsten englischen Wörter. Die Diagramme in Abbildung B.6 visualisieren den Einfluss von synthetisch generierten Wortbildern bei der Vorhersage der ELMo-Repräsentationen in Abhängigkeit von der Lexikongröße und dem jeweiligen Datensatz. Detaillierte Informationen zu der annotationsfreien Wissensdestillation und zur Durchführung der Experimente sind im Abschnitt 6.5.2 aufgeführt.

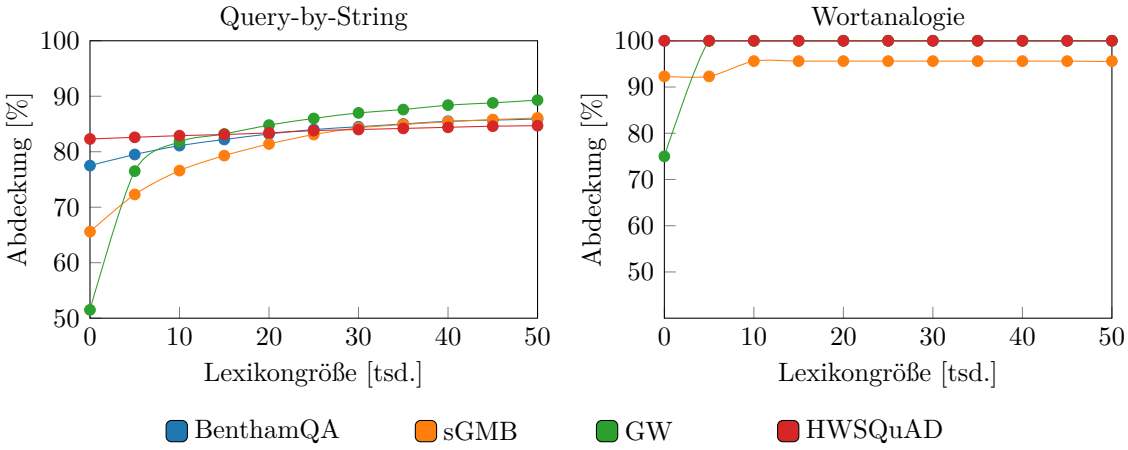


Abbildung B.5: Die prozentuale Abdeckung der QbS-Fragewörter bzw. der Zielwörter aus der Wortanalogie mit den Trainingsdaten in Abhängigkeit von der Lexikongröße.

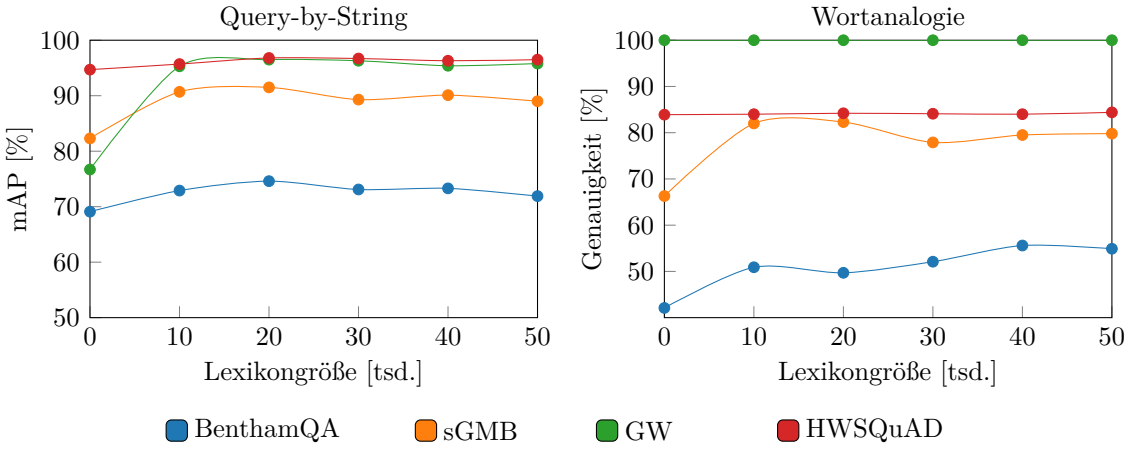


Abbildung B.6: Der Einfluss von synthetisch generierten Wortbildern bei der Vorhersage der ELMo-Repräsentationen in Abhängigkeit von der Lexikongröße.

B.3 EVALUATION VON KOMBINIERTEN WORTEINBETTUNGEN

In diesem Abschnitt werden die Auswirkungen der gewichteten Kombination von semantischen (ELMo) und syntaktischen (HWNet) Wortbildrepräsentationen auf die Datensätze George Washington, BenthamQA, HWSQuAD und sGMB gezeigt. Die Leistung wird anhand der Schlüsselwortsuche und der Wortanalogie bewertet. Die Diagramme in Abbildung B.7 zeigen die Auswirkungen des Gewichtungsfaktors bei der Kombination von ELMo- und HWNet-Repräsentationen bzgl. der QbS- und QbE-basierten Schlüsselwortsuche für die jeweiligen Benchmarks. Die Werte auf der x-Achse stellen den Gewichtungsfaktor dar, wobei der Wert 0 für eine rein syntaktische und der Wert 1 für eine rein semantische Wortbildrepräsentation steht. Abbildung B.8 visualisiert die Auswirkungen des Gewichtungsfaktors auf die Wortanalogie für die jeweiligen Benchmarks. Detaillierte Informationen zu der gewichteten Kombination und zur Durchführung der Experimente sind im Abschnitt 6.5.3 aufgeführt.

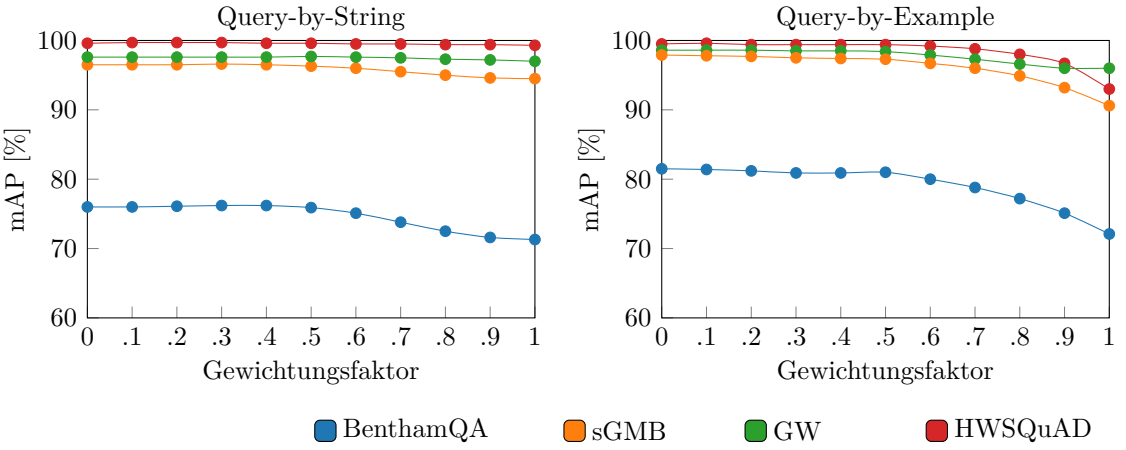


Abbildung B.7: Die Auswirkungen des Gewichtungsfaktors bei der Kombination von semantischen (ELMo) und syntaktischen (HWNet) Wortbildrepräsentationen auf die QbS- und QbE-basierte Schlüsselwortsuche für die jeweiligen Benchmarks.

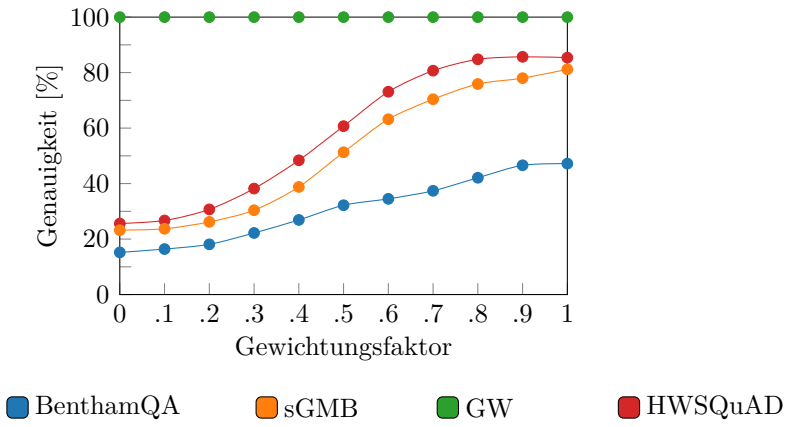


Abbildung B.8: Die Auswirkungen des Gewichtungsfaktors bei der Kombination von semantischen (ELMo) und syntaktischen (HWNet) Wortbildrepräsentationen auf die Wortanalogie für die jeweiligen Benchmarks.

[1] Chandranath Adak, Bidyut B. Chaudhuri und Michael Blumenstein. „Named Entity Recognition from Unstructured Handwritten Document Images“. In: *Int. Workshop on Document Analysis Systems*. Santorini, Greece, 2016, S. 375–380.

[2] Chandranath Adak, Bidyut B. Chaudhuri, Chin-Teng Lin und Michael Blumenstein. „Detecting Named Entities in Unstructured Bengali Manuscript Images“. In: *Int. Conf. on Document Analysis and Recognition*. Sydney, Australia, 2019, S. 196–201.

[3] Tomasz Adamek, Noel E. O'Connor und Alan F. Smeaton. „Word Matching Using Single Closed Contours for Indexing Handwritten Historical Documents“. In: *Int. Journal on Document Analysis and Recognition* 9.2-4 (2007), S. 153–165.

[4] Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter und Roland Vollgraf. „FLAIR: An Easy-to-Use Framework for State-of-the-Art NLP“. In: *Annual Conf. of the North American Chapter of the Association for Computational Linguistics*. Minneapolis, MN, USA, 2019, S. 54–59.

[5] Alan Akbik, Duncan Blythe und Roland Vollgraf. „Contextual String Embeddings for Sequence Labeling“. In: *Proc. Int. Conf. on Computational Linguistics*. Santa Fe, NM, USA, 2018, S. 1638–1649.

[6] Wissam AlKendi, Franck Gechter, Laurent Heyberger und Christophe Guyeux. „Advancements and Challenges in Handwritten Text Recognition: A Comprehensive Survey“. In: *Journal of Imaging* 10.1 (2024), S. 18.

[7] Jon Almazán, Albert Gordo, Alicia Fornés und Ernest Valveny. „Efficient Exemplar Word Spotting“. In: *British Machine Vision Conf.* Surrey, UK, 2012, S. 1–11.

[8] Jon Almazán, Albert Gordo, Alicia Fornés und Ernest Valveny. „Word Spotting and Recognition with Embedded Attributes“. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36.12 (2014), S. 2552–2566.

[9] Srikar Appalaraju, Bhavan Jasani, Bhargava Urala Kota, Yusheng Xie und R. Manmatha. „DocFormer: End-to-End Transformer for Document Understanding“. In: *Int. Conf. on Computer Vision*. Montreal, QC, Canada, 2021, S. 973–983.

[10] Srikar Appalaraju, Peng Tang, Qi Dong, Nishant Sankaran, Yichu Zhou und R. Manmatha. „DocFormerv2: Local Features for Document Understanding“. In: *Proc. AAAI Conf. on Artificial Intelligence*. Vancouver, BC, Canada, 2023, S. 709–718.

[11] Dzmitry Bahdanau, Kyunghyun Cho und Yoshua Bengio. „Neural Machine Translation by Jointly Learning to Align and Translate“. In: *Int. Conf. on Learning Representations*. San Diego, CA, USA, 2015.

[12] Vassileios Balntas, Edward Johns, Lilian Tang und Krystian Mikolajczyk. „PN-Net: Conjoined Triple Deep Network for Learning Local Image Descriptors“. In: *arXiv* (2016). arXiv: `abs/1601.05030`.

[13] Hangbo Bao, Li Dong, Songhao Piao und Furu Wei. „BEiT: BERT Pre-Training of Image Transformers“. In: *Int. Conf. on Learning Representations*. Virtuelles Event, 2022.

[14] Razieh Baradaran, Razieh Ghiasi und Hossein Amirkhani. „A Survey on Machine Reading Comprehension Systems“. In: *Natural Language Engineering* 28.6 (2022), S. 683–732.

[15] Richard Bellman. „Dynamic Programming“. In: *Science* 153 (1966), S. 34–37.

[16] Yoshua Bengio, Patrice Y. Simard und Paolo Frasconi. „Learning Long-Term Dependencies with Gradient Descent is Difficult“. In: *IEEE Transactions on Neural Networks* 5.2 (1994), S. 157–166.

[17] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. 2006. ISBN: 0387310738.

[18] Piotr Bojanowski, Edouard Grave, Armand Joulin und Tomás Mikolov. „Enriching Word Vectors with Subword Information“. In: *Transactions of the Association for Computational Linguistics* 5 (2017), S. 135–146.

[19] Emanuela Boros, Verónica Romero, Martin Maarand, Katerina Zenklová, Jitka Krecková, Enrique Vidal, Dominique Stutzmann und Christopher Kermorvant. „A Comparison of Sequential and Combined Approaches for Named Entity Recognition in a Corpus of Handwritten Medieval Charters“. In: *Proc. Int. Conf. on Frontiers in Handwriting Recognition*. Dortmund, Germany, 2020, S. 79–84.

[20] Johan Bos, Valerio Basile, Kilian Evang, Noortje Venhuizen und Johannes Bjerva. „The Groningen Meaning Bank“. In: *Handbook of Linguistic Annotation*. 2017, S. 463–496.

[21] Tom B. Brown *u. a.* „Language Models are Few-Shot Learners“. In: *Conf. on Neural Information Processing Systems*. Virtuelles Event, 2020, S. 1877–1901.

[22] Manuel Carbonell, Alicia Fornés, Mauricio Villegas und Josep Lladós. „A Neural Model for Text Localization, Transcription and Named Entity Recognition in Full Pages“. In: *Pattern Recognition, Letters* 136 (2020), S. 219–227.

[23] Manuel Carbonell, Pau Riba, Mauricio Villegas, Alicia Fornés und Josep Lladós. „Named Entity Recognition and Relation Extraction with Graph Neural Networks in Semi Structured Documents“. In: *Int. Conf. on Pattern Recognition*. Milan, Italy, 2020, S. 9622–9627.

[24] Manuel Carbonell, Mauricio Villegas, Alicia Fornés und Josep Lladós. „Joint Recognition of Handwritten Text and Named Entities with a Neural End-to-End Model“. In: *Int. Workshop on Document Analysis Systems*. Vienna, Austria, 2018, S. 399–404.

[25] Kartik Chaudhary und Raghav Bali. „Easter2.0: Improving Convolutional Models for Handwritten Text Recognition“. In: *arXiv* (2022). arXiv: `abs/2205.14879`.

[26] Ciprian Chelba, Tomás Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn und Tony Robinson. „One Billion Word Benchmark for Measuring Progress in Statistical Language Modeling“. In: *Annual Conf. of the Int. Speech Communication Association*. Singapore, 2014, S. 2635–2639.

[27] Danqi Chen, Adam Fisch, Jason Weston und Antoine Bordes. „Reading Wikipedia to Answer Open-Domain Questions“. In: *Annual Meeting of the Association for Computational Linguistics*. Vancouver, BC, Canada, 2017, S. 1870–1879.

[28] Jianpeng Cheng, Li Dong und Mirella Lapata. „Long Short-Term Memory-Networks for Machine Reading“. In: *Proc. Conf. on Empirical Methods in Natural Language Processing*. Austin, TX, USA, 2016, S. 551–561.

[29] Guillaume Chiron, Antoine Doucet, Mickaël Coustaty, Muriel Visani und Jean-Philippe Moreux. „Impact of OCR Errors on the Use of Digital Libraries: Towards a Better Access to Information“. In: *Joint Conf. on Digital Libraries*. Toronto, ON, Canada, 2017, S. 249–252.

[30] Laura Chiticariu, Yunyao Li und Frederick R. Reiss. „Rule-Based Information Extraction is Dead! Long Live Rule-Based Information Extraction Systems!“ In: *Proc. Conf. on Empirical Methods in Natural Language Processing*. Seattle, WA, USA, 2013, S. 827–832.

[31] Billy Chiu, Anna Korhonen und Sampo Pyysalo. „Intrinsic Evaluation of Word Vectors Fails to Predict Extrinsic Performance“. In: *Workshop on Evaluating Vector-Space Representations for NLP*. Berlin, Germany, 2016, S. 1–6.

[32] Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau und Yoshua Bengio. „On the Properties of Neural Machine Translation: Encoder-Decoder Approaches“. In: *Workshop on Syntax, Semantics and Structure in Statistical Translation*. Doha, Qatar, 2014, S. 103–111.

[33] Junyoung Chung, Çaglar Gülçehre, KyungHyun Cho und Yoshua Bengio. „Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling“. In: *arXiv* (2014). arXiv: `abs/1412.3555`.

[34] Corinna Cortes und Vladimir Vapnik. „Support-Vector Networks“. In: *Machine learning* 20.3 (1995), S. 273–297.

[35] Gabriella Csurka, Christopher Dance, Lixin Fan und Cédric Bray. „Visual Categorization with Bags of Keypoints“. In: *European Conf. on Computer Vision*. Prague, Czech Republic, 2004, S. 1–22.

[36] Lei Cui, Yiheng Xu, Tengchao Lv und Furu Wei. „Document AI: Benchmarks, Models and Applications“. In: *arXiv* (2021). arXiv: `abs/2111.08609`.

[37] Navneet Dalal und Bill Triggs. „Histograms of Oriented Gradients for Human Detection“. In: *Conf. on Computer Vision and Pattern Recognition*. San Diego, CA, USA, 2005, S. 886–893.

[38] Brian L. Davis, Bryan S. Morse, Brian L. Price, Chris Tensmeyer und Curtis Wigington. „Visual FUDGE: Form Understanding via Dynamic Graph Editing“. In: *Int. Conf. on Document Analysis and Recognition*. Lausanne, Switzerland, 2021, S. 416–431.

[39] Brian L. Davis, Bryan S. Morse, Brian L. Price, Chris Tensmeyer, Curtis Wigington und Vlad I. Morariu. „End-to-End Document Recognition and Understanding with Dessurt“. In: *European Conf. on Computer Vision*. Tel Aviv, Israel, 2022, S. 280–296.

[40] Claudio De Stefano, Francesco Fontanella, Donato Impedovo, Giuseppe Pirlo und Alessandra Scotto di Freca. „Handwriting Analysis to Support Neurodegenerative Diseases Diagnosis: A Review“. In: *Pattern Recognition, Letters* 121 (2019), S. 37–45.

[41] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li und Li Fei-Fei. „ImageNet: A Large-Scale Hierarchical Image Database“. In: *Conf. on Computer Vision and Pattern Recognition*. Miami, FL, USA, 2009, S. 248–255.

[42] Jacob Devlin, Ming-Wei Chang, Kenton Lee und Kristina Toutanova. „BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding“. In: *Annual Conf. of the North American Chapter of the Association for Computational Linguistics*. Minneapolis, MN, USA, 2019, S. 4171–4186.

[43] Marwa Dhiaf, Sana Khamekhem Jemni und Yousri Kessentini. „DocNER: A Deep Learning System for Named Entity Recognition in Handwritten Document Images“. In: *Int. Conf. on Neural Information Processing*. Bali, Indonesia, 2021, S. 239–246.

[44] Mohamed Dhouib, Ghassen Bettaieb und Aymen Shabou. „DocParser: End-to-end OCR-free Information Extraction from Visually Rich Documents“. In: *Int. Conf. on Document Analysis and Recognition*. San José, CA, USA, 2023, S. 155–172.

[45] David S. Doermann und Karl Tombre. *Handbook of Document Image Processing and Recognition*. 2014. ISBN: 978-0-85729-858-4.

[46] Yerai Doval, Jesús Vilares und Carlos Gómez-Rodríguez. „Towards Robust Word Embeddings for Noisy Texts“. In: *Applied Sciences* 10.19 (2020), S. 6893.

[47] Richard O. Duda, Peter E. Hart und David G. Stork. *Pattern Classification*. Second. New York, NY, USA, 2000. ISBN: 0471056693.

[48] Maud Ehrmann, Ahmed Hamdi, Elvys Linhares Pontes, Matteo Romanello und Antoine Doucet. „Named Entity Recognition and Classification in Historical Documents: A Survey“. In: *ACM Computing Surveys* 56 (2023), S. 1–47.

[49] Hicham El Boukkouri, Olivier Ferret, Thomas Lavergne, Hiroshi Noji, Pierre Zweigenbaum und Jun'ichi Tsujii. „CharacterBERT: Reconciling ELMo and BERT for Word-Level Open-Vocabulary Representations from Characters“. In: *Proc. Int. Conf. on Computational Linguistics*. Barcelona, Spain, 2020, S. 6903–6915.

[50] Kawin Ethayarajh. „How Contextual are Contextualized Word Representations? Comparing the Geometry of BERT, ELMo, and GPT-2 Embeddings“. In: *Proc. Conf. on Empirical Methods in Natural Language Processing*. Hong Kong, 2019, S. 55–65.

[51] Andreas Fischer, Andreas Keller, Volkmar Frinken und Horst Bunke. „Lexicon-Free Handwritten Word Spotting Using Character HMMs“. In: *Pattern Recognition, Letters* 33.7 (2012), S. 934–942.

[52] Andreas Fischer, Kaspar Riesen und Horst Bunke. „Graph Similarity Features for HMM-Based Handwriting Recognition in Historical Documents“. In: *Proc. Int. Conf. on Frontiers in Handwriting Recognition*. Kolkata, India, 2010, S. 253–258.

[53] Andreas Fischer, Ching Y. Suen, Volkmar Frinken, Kaspar Riesen und Horst Bunke. „Approximation of Graph Edit Distance Based on Hausdorff Matching“. In: *Pattern Recognition* 48.2 (2015), S. 331–343.

[54] Alicia Fornés, Verónica Romero, Arnau Baro, Juan Ignacio Toledo, Joan-Andreu Sánchez, Enrique Vidal und Josep Lladós. „ICDAR2017 Competition on Information Extraction in Historical Handwritten Records“. In: *Int. Conf. on Document Analysis and Recognition*. Kyoto, Japan, 2017, S. 1389–1394.

[55] Volkmar Frinken, Andreas Fischer, R. Manmatha und Horst Bunke. „A Novel Word Spotting Method Based on Recurrent Neural Networks“. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34.2 (2012), S. 211–224.

[56] Jörg Frochte. *Maschinelles Lernen: Grundlagen und Algorithmen in Python*. 2018. ISBN: 978-3446452916.

[57] Kunihiko Fukushima. „A Self-Organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position“. In: *Biological Cybernetics* 36 (1980), S. 193–202.

[58] Lukasz Garncarek, Rafal Powalski, Tomasz Stanislawek, Bartosz Topolski, Piotr Halama, Michal Turski und Filip Gralinski. „LAMBERT: Layout-Aware Language Modeling for Information Extraction“. In: *Int. Conf. on Document Analysis and Recognition*. Lausanne, Switzerland, 2021, S. 532–547.

[59] Andrea Gemelli, Sanket Biswas, Enrico Civitelli, Josep Lladós und Simone Marinai. „Doc2Graph: A Task Agnostic Document Understanding Framework Based on Graph Neural Networks“. In: *European Conf. on Computer Vision*. Tel Aviv, Israel, 2022, S. 329–344.

[60] Wulfram Gerstner und Werner M. Kistler. *Spiking Neuron Models: Single Neurons, Populations, Plasticity*. Cambridge University Press, 2002. ISBN: 978-0-51181570-6.

[61] Angelos P. Giotis, Giorgos Sfikas, Basilis Gatos und Christophoros Nikou. „A Survey of Document Image Word Spotting Techniques“. In: *Pattern Recognition* 68 (2017), S. 310–332.

[62] Xavier Glorot und Yoshua Bengio. „Understanding the Difficulty of Training Deep Feedforward Neural Networks“. In: *Int. Conf. on Artificial Intelligence and Statistics*. Sardinia, Italy, 2010, S. 249–256.

[63] Ian Goodfellow, Yoshua Bengio und Aaron Courville. *Deep Learning*. `http://www.deeplearningbook.org`. 2016.

[64] Jianping Gou, Baosheng Yu, Stephen J Maybank und Dacheng Tao. „Knowledge Distillation: A Survey“. In: *Int. Journal of Computer Vision* 129 (2021), S. 1789–1819.

[65] Ahmed Hamdi, Axel Jean-Caurant, Nicolas Sidere, Mickaël Coustaty und Antoine Doucet. „An Analysis of the Performance of Named Entity Recognition over OCRed Documents“. In: *Joint Conf. on Digital Libraries*. Champaign, IL, USA, 2019, S. 333–334.

[66] Ahmed Hamdi, Axel Jean-Caurant, Nicolas Sidère, Mickaël Coustaty und Antoine Doucet. „Assessing and Minimizing the Impact of OCR Quality on Named Entity Recognition“. In: *Int. Conf. on Theory and Practice of Digital Libraries*. Lyon, France, 2020, S. 87–101.

[67] Kaiming He, Xiangyu Zhang, Shaoqing Ren und Jian Sun. „Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition“. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37.9 (2015), S. 1904–1916.

[68] Kaiming He, Xiangyu Zhang, Shaoqing Ren und Jian Sun. „Deep Residual Learning for Image Recognition“. In: *Conf. on Computer Vision and Pattern Recognition*. Las Vegas, NV, USA, 2016, S. 770–778.

[69] Donald O. Hebb. *The Organization of Behavior: A Neuropsychological Theory*. 1949. ISBN: 0-8058-4300-0.

[70] Suzana Herculano-Houzel. „The Human Brain in Numbers: A Linearly Scaled-Up Primate Brain“. In: *Frontiers in Human Neuroscience* 3 (2009), S. 31.

[71] Sepp Hochreiter und Jürgen Schmidhuber. „Long Short-Term Memory“. In: *Neural Computation* 9.8 (1997), S. 1735–1780.

[72] Teakgyu Hong, Donghyun Kim, Mingi Ji, Wonseok Hwang, Daehyun Nam und Sungrae Park. „BROS: A Pre-trained Language Model Focusing on Text and Layout for Better Key Information Extraction from Documents“. In: *Proc. AAAI Conf. on Artificial Intelligence*. Virtuelles Event, 2022, S. 10767–10775.

[73] K. Hornik, M. Stinchcombe und H. White. „Multilayer Feedforward Networks Are Universal Approximators“. In: *Neural Networks* 2.5 (1989), S. 359–366.

[74] Chen Huang und Harish Srinivasan. „On the Discriminability of the Handwriting of Twins“. In: *Journal of Forensic Sciences* 53 (2008), S. 430–46.

[75] Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu und Furu Wei. „LayoutLMv3: Pre-training for Document AI with Unified Text and Image Masking“. In: *Int. Conf. on Multimedia*. Lisboa, Portugal, 2022, S. 4083–4091.

[76] David H. Hubel und Torsten N. Wiesel. „Receptive Fields and Functional Architecture of Monkey Striate Cortex“. In: *The Journal of Physiology* 195.1 (1968), S. 215–243.

[77] Johannes M. van Hulst, Faegheh Hasibi, Koen Dercksen, Krisztian Balog und Arjen P. de Vries. „REL: An Entity Linker Standing on the Shoulders of Giants“. In: *Proc. Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*. Xi'an, China, 2020, S. 2197–2200.

[78] Vinh-Nam Huynh, Ahmed Hamdi und Antoine Doucet. „When to Use OCR Post-Correction for Named Entity Recognition?“ In: *Int. Conf. on Asia-Pacific Digital Libraries*. Kyoto, Japan, 2020, S. 33–42.

[79] Emanuel Indermühle, Volkmar Frinken, Andreas Fischer und Horst Bunke. „Keyword Spotting in Online Handwritten Documents Containing Text and Non-text Using BLSTM Neural Networks“. In: *Int. Conf. on Document Analysis and Recognition*. Beijing, China, 2011, S. 73–77.

[80] Sergey Ioffe und Christian Szegedy. „Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift“. In: *Int. Conf. on Machine Learning*. Lille, France, 2015, S. 448–456.

[81] Anil K. Jain und Anoop M. Namboodiri. „Indexing and Retrieval of On-line Handwritten Documents“. In: *Int. Conf. on Document Analysis and Recognition*. Edinburgh, Scotland, 2003, S. 655–659.

[82] C. V. Jawahar, A. Balasubramanian, Million Meshesha und Anoop M. Namboodiri. „Retrieval of Online Handwriting by Synthesis and Matching“. In: *Pattern Recognition* 42.7 (2009), S. 1445–1457.

[83] Shaoxiong Ji, Shirui Pan, Erik Cambria, Pekka Marttinen und Philip S. Yu. „A Survey on Knowledge Graphs: Representation, Acquisition, and Applications“. In: *IEEE Transactions on Neural Networks and Learning Systems* 33.2 (2022), S. 494–514.

[84] Dan Jurafsky und James H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Second. 2009. ISBN: 9780135041963.

[85] Dan Jurafsky und James H. Martin. *Chapter B: Speech and Language Processing (3rd ed. draft)*. 2024. URL: `https://web.stanford.edu/~jurafsky/slp3/B.pdf` (besucht am 21. 05. 2024).

[86] Samira Ebrahimi Kahou, Vincent Michalski, Adam Atkinson, Ákos Kádár, Adam Trischler und Yoshua Bengio. „FigureQA: An Annotated Figure Dataset for Visual Reasoning“. In: *Int. Conf. on Learning Representations*. Vancouver, BC, Canada, 2018.

[87] Lei Kang, J. Ignacio Toledo, Pau Riba, Mauricio Villegas, Alicia Fornés und Marçal Rusiñol. „Convolve, Attend and Spell: An Attention-Based Sequence-to-Sequence Model for Handwritten Word Recognition“. In: *Proc. German Conf. on Pattern Recognition*. Stuttgart, Germany, 2018, S. 459–472.

[88] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick S. H. Lewis, Ledell Wu, Sergey Edunov, Danqi Chen und Wen-tau Yih. „Dense Passage Retrieval for Open-Domain Question Answering“. In: *Proc. Conf. on Empirical Methods in Natural Language Processing*. Virtuelles Event, 2020, S. 6769–6781.

[89] Siamak Khoubyari und Jonathan J. Hull. „Keyword Location in Noisy Document Images“. In: *Annual Symposium on Document Analysis and Information Retrieval*. Las Vegas, NV, USA, 1993, S. 217–231.

[90] Geewook Kim, Teakgyu Hong, Moonbin Yim, JeongYeon Nam, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han und Seunghyun Park. „OCR-Free Document Understanding Transformer“. In: *European Conf. on Computer Vision*. Tel Aviv, Israel, 2022, S. 498–517.

[91] Yoon Kim, Yacine Jernite, David A. Sontag und Alexander M. Rush. „Character-Aware Neural Language Models“. In: *Proc. AAAI Conf. on Artificial Intelligence*. Phoenix, AZ, USA, 2016, S. 2741–2749.

[92] Aleksander Kolcz, Joshua Alspector und Marijke F. Augusteijn. „A Line-Oriented Approach to Word Spotting in Handwritten Documents“. In: *Pattern Analysis and Applications* 3.2 (2000), S. 153–168.

[93] Sotiris B. Kotsiantis. „Supervised Machine Learning: A Review of Classification Techniques“. In: *Informatica (Slovenia)* 31.3 (2007), S. 249–268.

[94] Praveen Krishnan, Kartik Dutta und C. V. Jawahar. „HWNet v3: A Joint Embedding Framework for Recognition and Retrieval of Handwritten Text“. In: *Int. Journal on Document Analysis and Recognition* 26 (2023), S. 401–417.

[95] Praveen Krishnan und C. V. Jawahar. „Bringing Semantics in Word Image Retrieval“. In: *Int. Conf. on Document Analysis and Recognition*. Washington, DC, USA, 2013, S. 733–737.

[96] Praveen Krishnan und C. V. Jawahar. „Generating Synthetic Data for Text Recognition“. In: *arXiv* (2016). arXiv: `abs/1608.04224`.

[97] Praveen Krishnan und C. V. Jawahar. „HWNet v2: An Efficient Word Image Representation for Handwritten Documents“. In: *Int. Journal on Document Analysis and Recognition* 22 (2019), S. 387–405.

[98] Praveen Krishnan und C. V. Jawahar. „Bringing Semantics into Word Image Representation“. In: *Pattern Recognition* 108 (2020), S. 107542.

[99] Solomon Kullback und Richard A Leibler. „On Information and Sufficiency“. In: *The Annals of Mathematical Statistics* 22.1 (1951), S. 79–86.

[100] Ankit Kumar, Surbhi Bhatiya, Mohammad R Khosravi, Arwa Mashat und Parul Agarwal. „Semantic and Context Understanding for Sentiment Analysis in Hindi Handwritten Character Recognition Using a Multiresolution Technique“. In: *Transactions on Asian and Low-Resource Language Information Processing* 23.1 (2022), S. 1–22.

[101] Ankit Kumar, Piyush Makhija und Anuj Gupta. „Noisy Text Data: Achilles' Heel of BERT“. In: *Workshop on Noisy User-generated Text*. Virtuelles Event, 2020, S. 16–21.

[102] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami und Chris Dyer. „Neural Architectures for Named Entity Recognition“. In: *Annual Conf. of the North American Chapter of the Association for Computational Linguistics*. San Diego, CA, USA, 2016, S. 260–270.

[103] Jordy Van Landeghem, Rubèn Tito, Lukasz Borchmann, Michal Pietruszka, Dawid Jurkiewicz, Rafal Powalski, Pawel Józiak, Sanket Biswas, Mickaël Coustaty und Tomasz Stanislawek. „ICDAR 2023 Competition on Document UnderstanDing of Everything (DUDE)“. In: *Int. Conf. on Document Analysis and Recognition*. San José, CA, USA, 2023, S. 420–434.

[104] Jordy Van Landeghem *u. a.* „Document Understanding Dataset and Evaluation (DUDE)“. In: *Int. Conf. on Computer Vision*. Paris, France, 2023, S. 19471–19483.

[105] Svetlana Lazebnik, Cordelia Schmid und Jean Ponce. „Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories“. In: *Conf. on Computer Vision and Pattern Recognition*. New York, NY, USA, 2006, S. 2169–2178.

[106] Yann LeCun, Yoshua Bengio und Geoffrey E. Hinton. „Deep Learning“. In: *Nature* 521.7553 (2015), S. 436–444.

[107] Yann LeCun, Yoshua Bengio *u. a.* „Convolutional Networks for Images, Speech and Time Series“. In: *The Handbook of Brain Theory and Neural Networks* 3361.10 (1995), S. 255–258.

[108] Yann LeCun, Léon Bottou, Yoshua Bengio und Patrick Haffner. „Gradient-Based Learning Applied to Document Recognition“. In: *Proc. of the IEEE* 86.11 (1998), S. 2278–2324.

[109] Kenton Lee, Ming-Wei Chang und Kristina Toutanova. „Latent Retrieval for Weakly Supervised Open Domain Question Answering“. In: *Annual Meeting of the Association for Computational Linguistics*. Florence, Italy, 2019, S. 6086–6096.

[110] Kenton Lee, Mandar Joshi, Iulia Turc, Hexiang Hu, Fangyu Liu, Julian Eisenschlos, Urvashi Khandelwal, Peter Shaw, Ming-Wei Chang und Kristina Toutanova. „Pix2Struct: Screenshot Parsing as Pretraining for Visual Language Understanding“. In: *Int. Conf. on Machine Learning*. Honolulu, HI, USA, 2023, S. 18893–18912.

[111] Vladimir Levenshtein. „Binary Codes Capable of Correcting Deletions, Insertions and Reversals“. In: *Soviet Physics Doklady* 10 (1966), S. 707.

[112] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov und Luke Zettlemoyer. „BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension“. In: *Annual Meeting of the Association for Computational Linguistics*. Virtuelles Event, 2020, S. 7871–7880.

[113] Junlong Li, Yiheng Xu, Tengchao Lv, Lei Cui, Cha Zhang und Furu Wei. „DiT: Self-supervised Pre-training for Document Image Transformer“. In: *Int. Conf. on Multimedia*. Lisboa, Portugal, 2022, S. 3530–3539.

[114] Minghao Li, Tengchao Lv, Lei Cui, Yijuan Lu, Dinei A. F. Florêncio, Cha Zhang, Zhoujun Li und Furu Wei. „TrOCR: Transformer-Based Optical Character Recognition with Pre-trained Models“. In: *Proc. AAAI Conf. on Artificial Intelligence*. Washington, DC, USA, 2023, S. 13094–13102.

[115] Wenqi Li, Guotai Wang, Lucas Fidon, Sébastien Ourselin, M. Jorge Cardoso und Tom Vercauteren. „On the Compactness, Efficiency, and Representation of 3D Convolutional Networks: Brain Parcellation as a Pretext Task“. In: *Information Processing in Medical Imaging*. Boone, NC, USA, 2017, S. 348–360.

[116] Vladimir Lifschitz. „What Is Answer Set Programming?“ In: *Proc. AAAI Conf. on Artificial Intelligence*. Chicago, IL, USA, 2008, S. 1594–1597.

[117] Min Lin, Qiang Chen und Shuicheng Yan. „Network In Network“. In: *Int. Conf. on Learning Representations*. Banff, AB, Canada, 2014.

[118] Tianyang Lin, Yuxin Wang, Xiangyang Liu und Xipeng Qiu. „A Survey of Transformers“. In: *AI Open* 3 (2022), S. 111–132.

[119] Cheng-Lin Liu, Gernot A. Fink, Venu Govindaraju und Lianwen Jin. „Special Issue on Deep Learning for Document Analysis and Recognition“. In: *Int. Journal on Document Analysis and Recognition* 21.3 (2018), S. 159–160.

[120] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi und Graham Neubig. „Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing“. In: *ACM Computing Surveys* 55.9 (2023), S. 1–35.

[121] Shanshan Liu, Xin Zhang, Sheng Zhang, Hui Wang und Weiming Zhang. „Neural Machine Reading Comprehension: Methods and Trends“. In: *Applied Sciences* 9.18 (2019), S. 3698.

[122] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer und Veselin Stoyanov. „RoBERTa: A Robustly Optimized BERT Pretraining Approach“. In: *arXiv* (2019). arXiv: `abs/1907.11692`.

[123] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin und Baining Guo. „Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows“. In: *Int. Conf. on Computer Vision*. Montreal, QC, Canada, 2021, S. 9992–10002.

[124] Daniel Lopresti. „Optical Character Recognition Errors and Their Effects on Natural Language Processing“. In: *Proc. Workshop on Analytics for Noisy Unstructured Text Data*. Singapore, 2008, S. 9–16.

[125] David G. Lowe. „Object Recognition from Local Scale-Invariant Features“. In: *Int. Conf. on Computer Vision*. Corfu, Greece, 1999, S. 1150–1157.

[126] Wenjie Luo, Yujia Li, Raquel Urtasun und Richard S. Zemel. „Understanding the Effective Receptive Field in Deep Convolutional Neural Networks“. In: *Conf. on Neural Information Processing Systems*. Barcelona, Spain, 2016, S. 4898–4906.

[127] Mishaim Malik, Muhammad Kamran Malik, Khawar Mehmood und Imran Makhdoom. „Automatic Speech Recognition: A Survey“. In: *Multimedia Tools and Applications* 80.6 (2021), S. 9411–9457.

[128] R. Manmatha, Chengfeng Han und Edward M. Riseman. „Word Spotting: A New Approach to Indexing Handwriting“. In: *Conf. on Computer Vision and Pattern Recognition*. San Francisco, CA, USA, 1996, S. 631–637.

[129] R. Manmatha, Chengfeng Han, Edward M. Riseman und W. Bruce Croft. „Indexing Handwriting Using Word Matching“. In: *Int. Conf. on Digital Libraries*. Bethesda, MD, USA, 1996, S. 151–159.

[130] Christopher D. Manning, Prabhakar Raghavan und Hinrich Schütze. *Introduction to Information Retrieval*. 2008. ISBN: 978-0-521-86571-5.

[131] Urs-Viktor Marti und Horst Bunke. „The IAM-Database: An English Sentence Database for Offline Handwriting Recognition“. In: *Int. Journal on Document Analysis and Recognition* 5.1 (2002), S. 39–46.

[132] Minesh Mathew, Viraj Bagal, Rubèn Pérez Tito, Dimosthenis Karatzas, Ernest Valveny und C. V. Jawahar. „InfographicVQA“. In: *IEEE Winter Conf. on Applications of Computer Vision*. Waikoloa, HI, USA, 2022, S. 2582–2591.

[133] Minesh Mathew, Lluís Gómez, Dimosthenis Karatzas und C. V. Jawahar. „Asking Questions on Handwritten Document Collections“. In: *Int. Journal on Document Analysis and Recognition* 24 (2021), S. 235–249.

[134] Minesh Mathew, Dimosthenis Karatzas und C. V. Jawahar. „DocVQA: A Dataset for VQA on Document Images“. In: *IEEE Winter Conf. on Applications of Computer Vision*. Waikoloa, HI, USA, 2022, S. 2199–2208.

[135] Minesh Mathew, Rubèn Tito, Dimosthenis Karatzas, R. Manmatha und C. V. Jawahar. „Document Visual Question Answering Challenge 2020“. In: *arXiv* (2020). arXiv: `abs/2008.08899`.

[136] Warren S. McCulloch und Walter Pitts. „A Logical Calculus of the Ideas Immanent in Nervous Activity“. In: *The Bulletin of Mathematical Biophysics* 5.4 (1943), S. 115–133.

[137] Tomás Mikolov, Kai Chen, Greg Corrado und Jeffrey Dean. „Efficient Estimation of Word Representations in Vector Space“. In: *Int. Conf. on Learning Representations*. Scottsdale, AZ, USA, 2013.

[138] Shervin Minaee, Tomás Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain und Jianfeng Gao. „Large Language Models: A Survey“. In: *arXiv* (2024). arXiv: `abs/2402.06196`.

[139] Marvin Minsky und Seymour Papert. *Perceptrons: An Introduction to Computational Geometry*. Cambridge, MA, USA: MIT Press, 1969. ISBN: 9780262343930.

[140] Mandar Mitra und B. B. Chaudhuri. „Information Retrieval from Documents: A Survey“. In: *Information Retrieval Journal* 2.2 (2000), S. 141–163.

[141] Claire Bizon Monroc, Blanche Miret, Marie-Laurence Bonhomme und Christopher Kermorvant. „A Comprehensive Study of Open-Source Libraries for Named Entity Recognition on Handwritten Historical Documents“. In: *Int. Workshop on Document Analysis Systems*. La Rochelle, France, 2022, S. 429–444.

[142] Kevin P. Murphy. *Machine Learning - A Probabilistic Perspective*. MIT Press, 2012. ISBN: 0262018020.

[143] Jakub Náplava, Martin Popel, Milan Straka und Jana Straková. „Understanding Model Robustness to User-generated Noisy Texts“. In: *Workshop on Noisy User-generated Text*. Punta Cana, Dominican Republic, 2021, S. 340–350.

[144] Konstantina Nikolaidou, Mathias Seuret, Hamam Mokayed und Marcus Liwicki. „A Survey of Historical Document Image Datasets“. In: *Int. Journal on Document Analysis and Recognition* 25.4 (2022), S. 305–338.

[145] Timo Ojala, Matti Pietikäinen und David Harwood. „A Comparative Study of Texture Measures with Classification Based on Featured Distributions“. In: *Pattern Recognition* 29.1 (1996), S. 51–59.

[146] Jürgen Pafel und Ingo Reich. *Einführung in die Semantik: Grundlagen – Analysen – Theorien*. J.B. Metzler, 2016. ISBN: 9783476054258.

[147] Qiming Peng *u. a.* „ERNIE-Layout: Layout Knowledge Enhanced Pre-training for Visually-rich Document Understanding“. In: *Proc. Conf. on Empirical Methods in Natural Language Processing*. Abu Dhabi, United Arab Emirates, 2022, S. 3744–3756.

[148] Florent Perronnin, Jorge Sánchez und Thomas Mensink. „Improving the Fisher Kernel for Large-Scale Image Classification“. In: *European Conf. on Computer Vision*. Crete, Greece, 2010, S. 143–156.

[149] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee und Luke Zettlemoyer. „Deep Contextualized Word Representations“. In: *Annual Conf. of the North American Chapter of the Association for Computational Linguistics*. New Orleans, LA, USA, 2018, S. 2227–2237.

[150] Aleksandra Piktus, Necati Bora Edizel, Piotr Bojanowski, Edouard Grave, Rui Ferreira und Fabrizio Silvestri. „Misspelling Oblivious Word Embeddings“. In: *Annual Conf. of the North American Chapter of the Association for Computational Linguistics*. Minneapolis, MN, USA, 2019, S. 3226–3234.

[151] Réjean Plamondon und Sargur N. Srihari. „On-Line and Off-Line Handwriting Recognition: A Comprehensive Survey“. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22.1 (2000), 63–84.

[152] Rafal Powalski, Lukasz Borchmann, Dawid Jurkiewicz, Tomasz Dwojak, Michal Pietruszka und Gabriela Palka. „Going Full-TILT Boogie on Document Understanding with Text-Image-Layout Transformer“. In: *Int. Conf. on Document Analysis and Recognition*. Lausanne, Switzerland, 2021, S. 732–747.

[153] Animesh Prasad, Hervé Déjean, Jean-Luc Meunier, Max Weidemann, Johannes Michael und Gundram Leifert. „Bench-Marking Information Extraction in Semi-Structured Historical Handwritten Records“. In: *arXiv* (2018). arXiv: `abs/1807.06270`.

[154] Yuanyuan Qiu, Hongzheng Li, Shen Li, Yingdi Jiang, Renfen Hu und Lijiao Yang. „Revisiting Correlations Between Intrinsic and Extrinsic Evaluations of Word Embeddings“. In: *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data*. Changsha, China, 2018, S. 209–221.

[155] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever *u. a. Improving Language Understanding by Generative Pre-Training*. `https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf`. (besucht am 01.07.2024). 2018.

[156] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev und Percy Liang. „SQuAD: 100, 000+ Questions for Machine Comprehension of Text“. In: *Proc. Conf. on Empirical Methods in Natural Language Processing*. Austin, TX, USA, 2016, S. 2383–2392.

[157] Toni M. Rath und R. Manmatha. „Word Image Matching Using Dynamic Time Warping“. In: *Conf. on Computer Vision and Pattern Recognition*. Madison, WI, USA, 2003, S. 521–527.

[158] Toni M. Rath und R. Manmatha. „Word Spotting for Historical Documents“. In: *Int. Journal on Document Analysis and Recognition* 9.2-4 (2007), S. 139–152.

[159] Ehud Reiter und Robert Dale. „Building Applied Natural Language Generation Systems“. In: *Natural Language Engineering* 3.1 (1997), S. 57–87.

[160] Pau Riba, Josep Lladós und Alicia Fornés. „Handwritten Word Spotting by Inexact Matching of Grapheme Graphs“. In: *Int. Conf. on Document Analysis and Recognition*. Nancy, France, 2015, S. 781–785.

[161] Verónica Romero, Alicia Fornés, Nicolás Serrano, Joan-Andreu Sánchez, Alejandro H. Toselli, Volkmar Frinken, Enrique Vidal und Josep Lladós. „The ESPOSALLES Database: An Ancient Marriage License Corpus for Off-line Handwriting Recognition“. In: *Pattern Recognition* 46 (2013), S. 1658–1669.

[162] Frank Rosenblatt. „The Perceptron: A Probabilistic Model for Information Storage and Organization in The Brain“. In: *Psychological Review* 65.6 (1958), S. 65–386.

[163] Alejandro Héctor Toselli Rossi und Enrique Vidal. „Fast HMM-Filler Approach for Key Word Spotting in Handwritten Documents“. In: *Int. Conf. on Document Analysis and Recognition*. Washington, DC, USA, 2013, S. 501–505.

[164] Leonard Rothacker, Marçal Rusiñol und Gernot A. Fink. „Bag-of-Features HMMs for Segmentation-Free Word Spotting in Handwritten Documents“. In: *Int. Conf. on Document Analysis and Recognition*. Washington, DC, USA, 2013, S. 1305–1309.

[165] Ahmed Cheikh Rouhou, Marwa Dhiaf, Yousri Kessentini und Sinda Ben Salem. „Transformer-Based Approach for Joint Handwriting and Named Entity Recognition in Historical Document“. In: *Pattern Recognition, Letters* 155 (2022), S. 128–134.

[166] Vijay Rowtula, Praveen Krishnan und C. V. Jawahar. „PoS Tagging and Named Entity Recognition on Handwritten Documents“. In: *Int. Conf. on Natural Language Processing*. Patiala, India, 2018, 82–86.

[167] Vijay Rowtula, Subba Reddy Oota und C. V. Jawahar. „Towards Automated Evaluation of Handwritten Assessments“. In: *Int. Conf. on Document Analysis and Recognition*. Sydney, Australia, 2019, S. 426–433.

[168] David E Rumelhart, Geoffrey E Hinton und Ronald J Williams. „Learning Internal Representations by Error Propagation“. In: *Biometrika* 71 (1986), S. 599–607.

[169] David E. Rumelhart, Geoffrey E. Hinton und Ronald J. Williams. „Learning Representations by Back-Propagating Errors“. In: *Nature* 323 (1986), S. 533–536.

[170] Marçal Rusiñol, David Aldavert, Ricardo Toledo und Josep Lladós. „Browsing Heterogeneous Document Collections by a Segmentation-Free Word Spotting Method“. In: *Int. Conf. on Document Analysis and Recognition*. Beijing, China, 2011, S. 63–67.

[171] Marçal Rusiñol, David Aldavert, Ricardo Toledo und Josep Lladós. „Towards Query-by-Speech Handwritten Keyword Spotting“. In: *Int. Conf. on Document Analysis and Recognition*. Nancy, France, 2015, S. 501–505.

[172] Stuart J. Russell und Peter Norvig. *Artificial Intelligence - A Modern Approach*. Third. 2010. ISBN: 978-0-13-207148-2.

[173] Magnus Sahlgren. „The Distributional Hypothesis“. In: *Italian Journal of Linguistics* 20 (2008), S. 33–53.

[174] Hiroaki Sakoe und Seibi Chiba. „Dynamic Programming Algorithm Optimization for Spoken Word Recognition“. In: *Transactions on Acoustics, Speech, and Signal Processing* 26.1 (1978), S. 43–49.

[175] Arthur L. Samuel. „Some Studies in Machine Learning Using the Game of Checkers“. In: *IBM Journal of Research and Development* 3.3 (1959), S. 210–229.

[176] Mike Schuster und Kuldip K. Paliwal. „Bidirectional Recurrent Neural Networks“. In: *IEEE Transactions on Signal Processing* 45.11 (1997), S. 2673–2681.

[177] Stefan Schweter und Alan Akbik. „FLERT: Document-Level Features for Named Entity Recognition“. In: *arXiv* (2020). arXiv: `abs/2011.06993`.

[178] Rico Sennrich, Barry Haddow und Alexandra Birch. „Improving Neural Machine Translation Models with Monolingual Data“. In: *Annual Meeting of the Association for Computational Linguistics*. Berlin, Germany, 2016, S. 86–96.

[179] Min Joon Seo, Jinhyuk Lee, Tom Kwiatkowski, Ankur P. Parikh, Ali Farhadi und Hannaneh Hajishirzi. „Real-Time Open-Domain Question Answering with Dense-Sparse Phrase Index“. In: *Annual Meeting of the Association for Computational Linguistics*. Florence, Italy, 2019, S. 4430–4441.

[180] Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi und Hananneh Hajishirzi. „Bidirectional Attention Flow for Machine Comprehension“. In: *Int. Conf. on Learning Representations*. Toulon, France, 2017.

[181] Özge Sevgili, Artem Shelmanov, Mikhail Y. Arkhipov, Alexander Panchenko und Chris Biemann. „Neural Entity Linking: A Survey of Models Based on Deep Learning“. In: *Semantic Web* 13.3 (2022), S. 527–570.

[182] Erhan Sezerer und Selma Tekir. „A Survey On Neural Word Embeddings“. In: *arXiv* (2021). arXiv: abs/2110.01804.

[183] Karen Simonyan und Andrew Zisserman. „Very Deep Convolutional Networks for Large-Scale Image Recognition“. In: *Int. Conf. on Learning Representations*. San Diego, CA, USA, 2015.

[184] Sumeet S. Singh und Sergey Karayev. „Full Page Handwriting Recognition via Image to Sequence Extraction“. In: *Int. Conf. on Document Analysis and Recognition*. Lausanne, Switzerland, 2021, S. 55–69.

[185] Sargur N. Srihari, Jim Collins, Rohini K. Srihari, Harish Srinivasan, Shravya Shetty und Janina Brutt-Griffler. „Automatic Scoring of Short Handwritten Essays in Reading Comprehension Tests“. In: *Artificial Intelligence* 172.2-3 (2008), S. 300–324.

[186] Michael Stauffer, Andreas Fischer und Kaspar Riesen. „Graph-Based Keyword Spotting in Historical Documents Using Context-Aware Hausdorff Edit Distance“. In: *Int. Workshop on Document Analysis Systems*. Vienna, Austria, 2018, S. 49–54.

[187] Michael Stauffer, Andreas Fischer und Kaspar Riesen. „Keyword Spotting in Historical Handwritten Documents Based on Graph Matching“. In: *Pattern Recognition* 81 (2018), S. 240–253.

[188] Johansson. Stig, G. Leech und H. Goodluck. *Manual of Information to Accompany the Lancaster-Oslo-Bergen Corpus of British English, for Use with Digital Computers*. 1978. URL: http://korpus.uib.no/icame/manuals/LOB/INDEX.HTM (besucht am 01. 07. 2024).

[189] Daniel van Strien., Kaspar Beelen., Mariona Coll Ardanuy., Kasra Hosseini., Barbara McGillivray. und Giovanni Colavizza. „Assessing the Impact of OCR Quality on Downstream NLP Tasks“. In: *Proc. Int. Conf. on Agents and Artificial Intelligence*. Valletta, Malta, 2020, S. 484–496.

[190] Sebastian Sudholt. „Learning Attribute Representations with Deep Convolutional Neural Networks for Word Spotting“. Diss. Technische Universität Dortmund - Fakultät für Informatik, 2018.

[191] Sebastian Sudholt und Gernot A. Fink. „PHOCNet: A Deep Convolutional Neural Network for Word Spotting in Handwritten Documents“. In: *Proc. Int. Conf. on Frontiers in Handwriting Recognition*. Shenzhen, China, 2016, S. 277–282.

[192] Sebastian Sudholt und Gernot A. Fink. „Evaluating Word String Embeddings and Loss Functions for CNN-Based Word Spotting“. In: *Int. Conf. on Document Analysis and Recognition*. Kyoto, Japan, 2017, S. 493–498.

[193] Sebastian Sudholt und Gernot A. Fink. „Attribute CNNs for Word Spotting in Handwritten Documents“. In: *Int. Journal on Document Analysis and Recognition* 21.3 (2018), S. 199–218.

[194] Alaa Sulaiman, Khairuddin Omar und Mohammad Faidzul Nasrudin. „Degraded Historical Document Binarization: A Review on Issues, Challenges, Techniques, and Future Directions“. In: *Journal of Imaging* 5.4 (2019), S. 48.

[195] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E. Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke und Andrew Rabinovich. „Going Deeper with Convolutions“. In: *Conf. on Computer Vision and Pattern Recognition*. Boston, MA, USA, 2015, S. 1–9.

[196] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens und Zbigniew Wojna. „Rethinking the Inception Architecture for Computer Vision“. In: *Conf. on Computer Vision and Pattern Recognition*. Las Vegas, NV, USA, 2016, S. 2818–2826.

[197] Sachin S. Talathi und Aniket Vartak. „Improving Performance of Recurrent Neural Network with ReLU Nonlinearity“. In: *arXiv* (2015). arXiv: `abs/1511.03771`.

[198] Ryota Tanaka, Kyosuke Nishida, Kosuke Nishida, Taku Hasegawa, Itsumi Saito und Kuniko Saito. „SlideVQA: A Dataset for Document Visual Question Answering on Multiple Images“. In: *Proc. AAAI Conf. on Artificial Intelligence*. Washington, DC, USA, 2023, S. 13636–13645.

[199] Zineng Tang, Ziyi Yang, Guoxin Wang, Yuwei Fang, Yang Liu, Chenguang Zhu, Michael Zeng, Cha Zhang und Mohit Bansal. „Unifying Vision, Text, and Layout for Universal Document Processing“. In: *Conf. on Computer Vision and Pattern Recognition*. Vancouver, BC, Canada, 2023, S. 19254–19264.

[200] Solène Tarride, Mélodie Boillet und Christopher Kermorvant. „Key-Value Information Extraction from Full Handwritten Pages“. In: *Int. Conf. on Document Analysis and Recognition*. San José, CA, USA, 2023, S. 185–204.

[201] Solène Tarride, Aurélie Lemaitre, Bertrand Coüasnon und Sophie Tardivel. „A Comparative Study of Information Extraction Strategies Using an Attention-Based Neural Network“. In: *Int. Workshop on Document Analysis Systems*. La Rochelle, France, 2022, S. 644–658.

[202] Solène Tarride, Martin Maarand, Mélodie Boillet, James McGrath, Eugénie Capel, Hélène Vézina und Christopher Kermorvant. „Large-Scale Genealogical Information Extraction from Handwritten Quebec Parish Records“. In: *Int. Journal on Document Analysis and Recognition* 26 (2023), S. 255–272.

[203] Raytheon BBN Technologies. *OntoNotes Named Entity Guidelines Version 14.0*. 2004.

[204] Ian Tenney, Dipanjan Das und Ellie Pavlick. „BERT Rediscovers the Classical NLP Pipeline“. In: *Annual Meeting of the Association for Computational Linguistics*. Florence, Italy, 2019, S. 4593–4601.

[205] Rubèn Tito, Minesh Mathew, C. V. Jawahar, Ernest Valveny und Dimosthenis Karatzas. „ICDAR 2021 Competition on Document Visual Question Answering“. In: *Int. Conf. on Document Analysis and Recognition*. Lausanne, Switzerland, 2021, S. 635–649.

[206] Juan Ignacio Toledo, Manuel Carbonell, Alicia Fornés und Josep Lladós. „Information Extraction from Historical Handwritten Document Images with a Context-Aware Neural Model“. In: *Pattern Recognition* 86 (2019), S. 27–36.

[207] Juan Ignacio Toledo, Sebastian Sudholt, Alicia Fornés, Jordi Cucurull, Gernot A. Fink und Josep Lladós. „Handwritten Word Image Categorization with Convolutional Neural Networks and Spatial Pyramid Pooling“. In: *Structural, Syntactic, and Statistical Pattern Recognition*. Mérida, Mexico, 2016, S. 543–552.

[208] François Torregrossa, Robin Allesiardo, Vincent Claveau, Nihel Kooli und Guillaume Gravier. „A Survey on Training and Evaluation of Word Embeddings“. In: *Int. Journal of Data Science and Analytics* 11.2 (2021), S. 85–103.

[209] François Torregrossa, Vincent Claveau, Nihel Kooli, Guillaume Gravier und Robin Allesiardo. „On the Correlation of Word Embedding Evaluation Metrics“. In: *Proc. Int. Conf. on Language Resources and Evaluation*. Marseille, France, 2020, S. 4789–4797.

[210] Oliver Tüselmann, Kai Brandenbusch, Miao Chen und Gernot A. Fink. „A Weighted Combination of Semantic and Syntatic Word Image Representations“. In: *Proc. Int. Conf. on Frontiers in Handwriting Recognition*. Hyderabad, India, 2022, S. 285–299.

[211] Oliver Tüselmann und Gernot A. Fink. „Named Entity Linking on Handwritten Document Images“. In: *Int. Workshop on Document Analysis Systems*. La Rochelle, France, 2022, S. 199–213.

[212] Oliver Tüselmann und Gernot A. Fink. „Exploring Semantic Word Representations for Recognition-free NLP on Handwritten Document Images“. In: *Int. Conf. on Document Analysis and Recognition*. San José, CA, USA, 2023, S. 85–100.

[213] Oliver Tüselmann, Friedrich Müller, Fabian Wolf und Gernot A. Fink. „Recognition-free Question Answering on Handwritten Document Collections“. In: *Proc. Int. Conf. on Frontiers in Handwriting Recognition*. Hyderabad, India, 2022, S. 259–273.

[214] Oliver Tüselmann, Fabian Wolf und Gernot A. Fink. „Identifying and Tackling Key Challenges in Semantic Word Spotting“. In: *Proc. Int. Conf. on Frontiers in Handwriting Recognition*. Dortmund, Germany, 2020, S. 55–60.

[215] Oliver Tüselmann, Fabian Wolf und Gernot A. Fink. „Are End-to-End Systems Really Necessary for NER on Handwritten Document Images?“ In: *Int. Conf. on Document Analysis and Recognition*. Lausanne, Switzerland, 2021, S. 808–822.

[216] Oliver Tüselmann und Gernot A. Fink. „Neural Models for Semantic Analysis of Handwritten Document Images“. In: *Int. Journal on Document Analysis and Recognition* (2024).

[217] David Vallet, Miriam Fernández und Pablo Castells. „An Ontology-Based Information Retrieval Model“. In: *European Semantic Web Conf.* Crete, Greece, 2005, S. 455–470.

[218] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser und Illia Polosukhin. „Attention is All you Need“. In: *Conf. on Neural Information Processing Systems*. Long Beach, CA, USA, 2017, 6000–6010.

[219] David Villanova-Aparisi, Carlos D. Martínez-Hinarejos, Verónica Romero und Moisés Pastor-i-Gadea. „Consistent Nested Named Entity Recognition in Handwritten Documents via Lattice Rescoring“. In: *Int. Conf. on Document Analysis and Recognition*. San José, CA, USA, 2023, S. 255–268.

[220] David Villanova-Aparisi, Carlos D. Martínez-Hinarejos, Verónica Romero und Moisés Pastor-i-Gadea. „Evaluation of Different Tagging Schemes for Named Entity Recognition in Handwritten Documents“. In: *Int. Conf. on Document Analysis and Recognition*. San José, CA, USA, 2023, S. 3–16.

[221] Mauricio Villegas, Joan Puigcerver, Alejandro H. Toselli, Joan-Andreu Sánchez und Enrique Vidal. „Overview of the ImageCLEF 2016 Handwritten Scanned Document Retrieval Task“. In: *Working Notes of Conf. and Labs of the Evaluation Forum*. Évora, Portugal, 2016, S. 233–253.

[222] Bin Wang, Angela Wang, Fenxiao Chen, Yuncheng Wang und C.-C. Jay Kuo. „Evaluating Word Embedding Models: Methods and Experimental Results“. In: *Transactions on Signal and Information Processing* 8 (2019), S. 19–33.

[223] Peng Wang, Véronique Eglin, Christophe Garcia, Christine Largeron, Josep Lladós und Alicia Fornés. „A Novel Learning-Free Word Spotting Approach Based on Graph Representation“. In: *Int. Workshop on Document Analysis Systems*. Tours, France, 2014, S. 207–211.

[224] Shirui Wang, Wen'an Zhou und Chao Jiang. „A Survey of Word Embeddings Based on Deep Learning“. In: *Computing* 102 (2019), S. 717–740.

[225] Xinyu Wang, Yong Jiang, Nguyen Bach, Tao Wang, Zhongqiang Huang, Fei Huang und Kewei Tu. „Automated Concatenation of Embeddings for Structured Prediction“. In: *Int. Joint Conf. on Natural Language Processing*. Bangkok, Thailand, 2021, S. 2643–2660.

[226] Yizhong Wang, Kai Liu, Jing Liu, Wei He, Yajuan Lyu, Hua Wu, Sujian Li und Haifeng Wang. „Multi-Passage Machine Reading Comprehension with Cross-Passage Answer Verification“. In: *Annual Meeting of the Association for Computational Linguistics*. Melbourne, Australia, 2018, S. 1918–1927.

[227] Zhiguo Wang, Patrick Ng, Xiaofei Ma, Ramesh Nallapati und Bing Xiang. „Multi-Passage BERT: A Globally Normalized BERT Model for Open-Domain Question Answering“. In: *Proc. Conf. on Empirical Methods in Natural Language Processing*. Hong Kong, China, 2019, S. 5877–5881.

[228] Yan Wen, Cong Fan, Geng Chen, Xin Chen und Ming Chen. „A Survey on Named Entity Recognition“. In: *Int. Conf. on Communications, Signal Processing, and Systems*. Urumqi, China, 2019, S. 1803–1810.

[229] Lilian Weng. *Learning Word Embedding*. 2017. URL: `https://lilianweng.github.io/posts/2017-10-15-word-embedding/` (besucht am 01. 07. 2024).

[230] Paul J. Werbos. „Backpropagation Through Time: What It Does and How to Do It“. In: *Proc. of the IEEE* 78.10 (1990), S. 1550–1560.

[231] Christian Wieprecht, Leonard Rothacker und Gernot A. Fink. „Word Spotting in Historical Document Collections with Online-Handwritten Queries“. In: *Int. Workshop on Document Analysis Systems*. Santorini, Greece, 2016, S. 162–167.

[232] Max De Wilde und Simon Hengchen. „Semantic Enrichment of a Multilingual Archive with Linked Open Data“. In: *Digital Humanities Quarterly* 11.4 (2017).

[233] Tomas Wilkinson und Anders Brun. „Semantic and Verbatim Word Spotting Using Deep Neural Networks“. In: *Proc. Int. Conf. on Frontiers in Handwriting Recognition*. Shenzhen, China, 2016, S. 307–312.

[234] Tomas Wilkinson, Jonas Lindström und Anders Brun. „Neural Ctrl-F: Segmentation-Free Query-by-String Word Spotting in Handwritten Manuscript Collections“. In: *Int. Conf. on Computer Vision*. Venice, Italy, 2017, S. 4443–4452.

[235] Fabian Wolf, Kai Brandenbusch und Gernot A. Fink. „Improving Handwritten Word Synthesis for Annotation-free Word Spotting“. In: *Proc. Int. Conf. on Frontiers in Handwriting Recognition*. Dortmund, Germany, 2020, S. 61–66.

[236] Fabian Wolf und Gernot A. Fink. „Annotation-free Learning of Deep Representations for Word Spotting Using Synthetic Data and Self Labeling“. In: *Int. Workshop on Document Analysis Systems*. Wuhan, China, 2020, S. 293–308.

[237] Fabian Wolf und Gernot A. Fink. „Self-Training of Handwritten Word Recognition for Synthetic-to-Real Adaptation“. In: *Int. Conf. on Pattern Recognition*. Montreal, QC, Canada, 2022, S. 3885–3892.

[238] Yonghui Wu *u. a.* „Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation“. In: *arXiv* (2016). arXiv: `abs/1609.08144`.

[239] Peng Xu, Xiatian Zhu und David A. Clifton. „Multimodal Learning with Transformers: A Survey“. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45.10 (2023), S. 12113–12132.

[240] Yang Xu *u. a.* „LayoutLMv2: Multi-modal Pre-training for Visually-rich Document Understanding“. In: *Int. Joint Conf. on Natural Language Processing*. Bangkok, Thailand, 2021, S. 2579–2591.

[241] Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei und Ming Zhou. „LayoutLM: Pre-training of Text and Layout for Document Image Understanding“. In: *Int. Conf. on Knowledge Discovery and Data Mining*. Virtuelles Event, 2020, S. 1192–1200.

[242] Vikas Yadav und Steven Bethard. „A Survey on Recent Advances in Named Entity Recognition from Deep Learning Models“. In: *Proc. Int. Conf. on Computational Linguistics*. Santa Fe, NM, USA, 2018, S. 2145–2158.

[243] Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda und Yuji Matsumoto. „LUKE: Deep Contextualized Entity Representations with Entity-Aware Self-attention“. In: *Proc. Conf. on Empirical Methods in Natural Language Processing*. Virtuelles Event, 2020, S. 6442–6454.

[244] Hang Yan, Bocao Deng, Xiaonan Li und Xipeng Qiu. „TENER: Adapting Transformer Encoder for Named Entity Recognition“. In: *arXiv* (2019). arXiv: `abs/1911.04474`.

[245] Shuoheng Yang, Yuxin Wang und Xiaowen Chu. „A Survey of Deep Learning Techniques for Neural Machine Translation“. In: *arXiv* (2020). arXiv: `abs/2002.07526`.

[246] Shukang Yin, Chaoyou Fu, Sirui Zhao, Ke Li, Xing Sun, Tong Xu und Enhong Chen. „A Survey on Multimodal Large Language Models“. In: *arXiv* (2023). arXiv: `abs/2306.13549`.

[247] Yuechen Yu, Yulin Li, Chengquan Zhang, Xiaoqiang Zhang, Zengyuan Guo, Xiameng Qin, Kun Yao, Junyu Han, Errui Ding und Jingdong Wang. „StrucTexTv2: Masked Visual-Textual Prediction for Document Image Pre-training“. In: *Int. Conf. on Learning Representations*. Kigali, Ruanda, 2023.

[248] Matthew D. Zeiler und Rob Fergus. „Visualizing and Understanding Convolutional Networks“. In: *European Conf. on Computer Vision*. Cham, Germany, 2014, S. 818–833.

[249] Chengchang Zeng, Shaobo Li, Qin Li, Jie Hu und Jianjun Hu. „A Survey on Machine Reading Comprehension: Tasks, Evaluation Metrics, and Benchmark Datasets“. In: *Applied Sciences* 10.21 (2020), S. 7640.

[250] Aston Zhang, Zachary C. Lipton, Mu Li und Alexander J. Smola. *Dive into Deep Learning*. `https://D2L.ai`. 2023.

[251] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong *u. a.* „A Survey of Large Language Models“. In: *arXiv* (2023). arXiv: `abs/2303.18223`.

[252] Fengbin Zhu, Wenqiang Lei, Chao Wang, Jianming Zheng, Soujanya Poria und Tat-Seng Chua. „Retrieving and Reading: A Comprehensive Survey on Open-Domain Question Answering“. In: *arXiv* (2021). arXiv: `abs/2101.00774`.

[253] Yukun Zhu, Ryan Kiros, Richard S. Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba und Sanja Fidler. „Aligning Books and Movies: Towards Story-Like Visual Explanations by Watching Movies and Reading Books“. In: *Int. Conf. on Computer Vision*. Santiago, Chile, 2015, S. 19–27.

[254] Amin Zollanvari. „Decision Trees“. In: *Machine Learning with Python: Theory and Implementation*. 2023, S. 187–207. ISBN: 978-3-031-33342-2.

ABKÜRZUNGSVERZEICHNIS

AP	Average Precision
BERT	Bidirectional Encoder Representations from Transformers
BIDAF	Bidirectional Attention Flow
BLSTM	Bidirectional Long Short-Term Memory
BoF	Bag-of-Features
CBoW	Continuous Bag-of-Words
CER	Character Error Rate
CharCNN	Character-Level Convolutional Neural Network
CNN	Convolutional Neural Network
CRF	Conditional Random Field
CV	Computer Vision
Dessurt	Document end-to-end self-supervised understanding and recognition transformer
DIS	Double Inclusion Score
DL	Deep Learning
ELMo	Embeddings from Language Model
GAP	Global Average Pooling
GNN	Graph Neural Network
GRU	Gated Recurrent Unit
GW	George Washington
HMM	Hidden Markov Model
HTR	Handwritten Text Recognition
IoU	Intersection over Union
IR	Information Retrieval
LLM	Large Language Model
LSTM	Long Short-Term Memory
mAP	mean Average Precision
MLM	Masked Language Modelling
MLP	Multi-Layer Perceptron

MRC Machine Reading Comprehension
MSE Mean Squared Error

NE Named Entity
NEL Named Entity Linking
NER Named Entity Recognition
NLP Natural Language Processing
NSP Next Sentence Prediction

OCR Optical Character Recognition

PHOC Pyramidal Histogram of Characters
PoS Part-of-Speech

QA Question Answering
QbE Query-by-Example
QbS Query-by-String

ReLU Rectified Linear Unit
ResNet Residual Network
RNN Recurrent Neural Network
RoBERTa Robustly optimized BERT approach

sGMB synthetic Groningen Meaning Bank
SPP Spatial Pyramid Pooling
SQuAD Stanford Question Answering Dataset
SVM Support Vector Machine

tanh Tangens Hyperbolicus
TF-IDF Term Frequency-Inverse Document Frequency
TPP Temporal Pyramid Pooling

WA Word Analogy
WER Word Error Rate
WS Word Spotting

DANKSAGUNG

Mit der Fertigstellung dieser Dissertation geht für mich eine prägende und intensive Zeit zu Ende, die ich ohne die Unterstützung vieler Menschen nicht erfolgreich hätte bewältigen können. Ihnen allen gilt mein herzlichster Dank.

Mein besonderer Dank gilt zunächst meinem Betreuer, Prof. Dr.-Ing. Gernot A. Fink. Gernot, ich danke dir herzlich dafür, dass ich meine Arbeit unter deiner Leitung verfassen konnte und für die Freiheiten, die du mir dabei gewährt hast. Deine fachliche Expertise und dein stets offenes Ohr für meine Fragen und Anliegen waren von unschätzbarem Wert. Dein wertvolles Feedback hat diese Arbeit entscheidend geprägt und das Erreichen dieses Ziels möglich gemacht.

Ebenso möchte ich meinem Zweitgutachter, Prof. Dr. Andreas Fischer, herzlich danken, der sich sofort bereit erklärt hat, meine Dissertation zu begutachten und mir mit seiner fachlichen Expertise zur Seite gestanden hat. Dein wertvolles Feedback und deine Unterstützung in diesem Prozess waren mir eine große Hilfe.

Ein weiterer Dank gilt Prof. Dr. Erich Schubert für seine Bereitschaft, mich im Rahmen des strukturierten Promotionsprogramms als Mentor zu betreuen. Ebenso danke ich Prof. Dr. Mario Botsch und Prof. Dr. Stefan Harmeling für ihre Mitarbeit in der Prüfungskommission.

Mein Dank gilt auch allen Kolleginnen und Kollegen der Mustererkennungsgruppe, die mich während meiner Promotion mit Rat und Tat unterstützt haben. Namentlich danken möchte ich Fabian Wolf, Dr.-Ing. Philipp Oberdiek, Dr.-Ing. Fernando Moya, Kai Brandenbusch, Dominik Koßmann, Arthur Matei, Nilah Ravi Nair, Eugen Rusakov, Tim Raven und Tim Hallyburton. Trotz der vielen Arbeitstage im Home-Office werden mir die gemeinsamen Bürotage, Tagungsreisen und Teamevents immer in bester Erinnerung bleiben.

Ein herzlicher Dank geht zudem an meine Familie und Freunde, die mich während meiner gesamten Promotionszeit bedingungslos unterstützt haben. Besonders danke ich meinen Eltern, die mir immer den Glauben an mich selbst vermittelt und mir jede Möglichkeit eröffnet haben. Ich kann euch gar nicht genug dafür danken.

Ein besonderer Dank gilt meiner Lebensgefährtin Lara Stockel. Du hast mich durch all die Höhen und Tiefen dieser Zeit begleitet und sowohl die schönen Erfolge als auch die Herausforderungen mit mir geteilt. Dein Vertrauen in mich und deine positive Art haben mir immer wieder Kraft gegeben, auch wenn es mal schwierig wurde. Ohne deine Geduld und deine Unterstützung wäre dieser Weg für mich kaum denkbar gewesen. Ein großer Teil dieses Erfolges gehört deshalb auch dir.

Abschließend danke ich allen, die auf unterschiedlichste Weise zum Gelingen dieser Arbeit beigetragen haben – durch fachliche Anregungen, praktische Hilfe oder moralische Unterstützung.