Extraktion von Metadaten aus wissenschaftlichen Artikeln mittels Transformern

Christian Peters, 08.02.2022

Grundlegendes Problem

- Jährlich mehrere Millionen neue wissenschaftliche Artikel (arXiv: >15 Paper pro Stunde)
- Relevante Artikel finden schwierig → Klassifizierung der Artikel notwendig
- Klassifizierung muss automatisiert werden

Klassifizierung von Artikeln

- Klassifizierung mit Hilfe von Metadaten
- Zwei Arten von Metadaten:
  - Bibliographisch (Autoren, Jahr, Titel, ...)
  - Inhaltlich
    - Im Artikel vorkommende Fachbegriffe
    - Verwendete Techniken
    - Untersuchungsort

Verwendeter Datensatz

- ~8.500 Artikel der Bodenwissenschaften
- Verschiedene Magazine, verschiedene Zeiträume
- Englische PDF-Dateien

Ansätze der Klassifizierung

- Umwandlung von PDF in TEI-XML
- Klassifizierung nach folgenden Datengruppen:
  - Bodenarten
  - Nutzpflanzen
  - Bodentexturen
  - Koordinaten

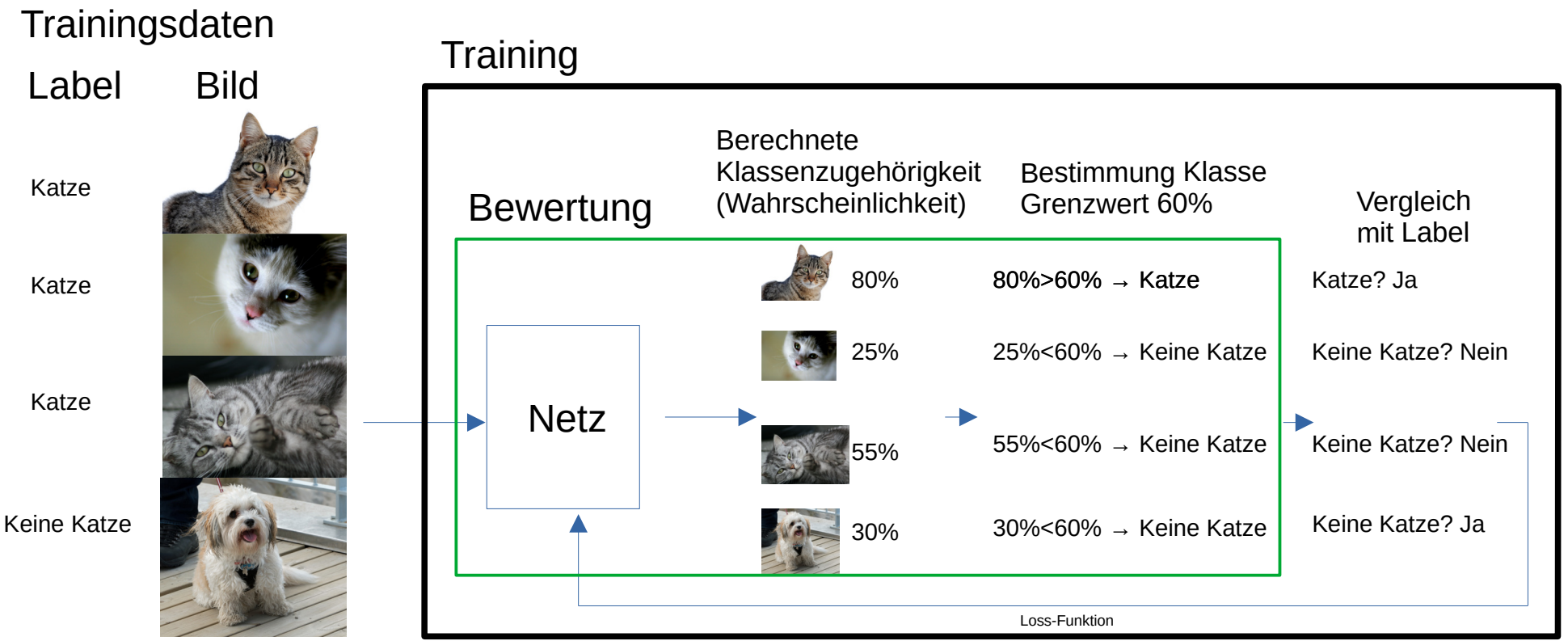
Volltextsuche

- Nicht für alle Datengruppen einfach möglich
  - Bodenarten, Bodentexturen, Pflanzen
- Probleme:
  - Wortlisten nötig (Vollständigkeit)
  - Rechtschreibfehler
  - Fehler in der Konvertierung

Künstliche neuronale Netze

- Umgehen Probleme der Volltextsuche
- Hier genutzt: BERT (Bidirectional Encoder Representations from Transformers)
- Kann nach Training mit Trainingsdaten beliebige Inhalte klassifizieren

Grundlegende Funktionsweise



Neuronale Netze und Texte

- Wörter werden in Token zerlegt, die vom Netz verarbeitet werden können
  - „science“ → science
  - „geoscience“ → geo, ##sc, ##ience
- Token stammen aus Wörterbuch des Netzes
- Ganzer Textabschnitt ist ein Datum

Transformer-Besonderheiten

- Transformer nutzen „Attention“
- Aufmerksamkeit, die ein Token einem anderen widmet (wichtig für Klassifizierung)
- Höhere Attention bedeutet wichtigeres Wort
- Zusätzlich: Class-Token [CLS]
  - Label für die gesamte Sequenz

Attention - Beispiel

- Klassifikationsproblem: Beschreibender Satz?
- Beispielsatz: „Der Baum ist grün.“

		Empfängertoken				
		CLS	Der	Baum	ist	grün.
Sender-token	CLS	0,1	0,3	0,4	0,3	0,4
	Der	0,2	0,3	0,1	0,3	0,3
	Baum	0,3	0,3	0,5	0,3	0,5
	ist	0,6	0,2	0,3	0,3	0,5
	grün.	0,6	0,1	0,5	0,6	0,5

Attention - Zusammenfassung

- Hoher Attentionwert zum Class-Token bedeutet „Sendertoken ist wichtig für die Klassifizierung“
- Benutzung aller Wörter und ihrer Position in der Sequenz für die Klassifizierung der Sequenz
- Attention wird bei der Extraktion mittels Sequenzklassifikation genutzt

Sequenzklassifikation

- Klassifikation eines Paragraphen eines Artikels
  - „Enthält dieser Paragraph eine Bodenart?“
- Jeweils eigenes Netz für jede Datenklasse
- Trainingsdaten sind Paare (Paragraph, Label)
- Trainingsdatenerzeugung ist ein Problem

Problem: Labelfindung

- Henne-Ei-Problem: Vollständige, korrekte Label benötigen Lösung des Problems, das mit dem Netz gelöst werden soll
- Abhilfe: Volltextsuche mit regulären Ausdrücken nach bekannten Worten
- Problem: Koordinaten

Beispielkoordinaten

Konvertierter Originaltext	Zu extrahierende Koordinate
35805 H S, 147820 H E	35° 05' S, 147° 20' E
13 • 15 N, 2 • 32 E	13° 15' N, 2° 32' E
43°43¢17¢¢N-4°10¢25¢¢E	43° 43' 17", 4° 10' 25" E
2847V -3815VN and 10824V -10851VE	2° 47' - 3° 15' N, 10° 24' - 10° 51' E

Labeln von Koordinaten

- Strenge reguläre Ausdrücke finden nur einzelne Koordinaten
  - Große Menge von regulären Ausdrücken nötig
- Freiere reguläre Ausdrücke finden sehr viele falsch-positive Ergebnisse (>70%)
- Lösung: Strenge verwenden

Extraktion aus Sequenzen

- Trainingsdatenerzeugung für die Sequenzklassifikation durchführbar
- Korrekte Label führen allerdings nicht direkt zur Extraktion
- Drei Varianten der Extraktion wurden getestet

Halbierungsmethode

- Wird ein Paragraph als „Enthaltend“ bewertet:
  - Teile den Paragraph in der Mitte
  - Wende das Netz auf so entstandene Teilparagraphen an
  - Wiederhole, bis kein Teilparagraph mehr als „Enthaltend“ gelabelt wird
  - Extrahiert wird der letzte „Enthaltend“-Teilparagraph

Attention-Methode

- Wird ein Paragraph als „Enthaltend“ bewertet:
  - Extrahiere Token, die die höchste Attention zum Class-Token haben
  - Hoffnung: Für Klassifikation relevante Token sind für Datenklasse relevante Token

Attention-Halbierungsmethode

- Wird ein Paragraph als „Enthaltend“ bewertet:
  - Zerlege den Paragraphen so in zwei Hälften, dass beide in Summe die gleichen Attentionwerte aufweisen, dann wiederhole mit neuen Teilparagraphen
  - Hoffnung: Extraktion von kurzen Sequenzen mit hoher Attention, die relevant sind

Ergebnisse der Sequenzen

	Halbierungsmethode		Attentionmethode		Attention-Halbierungsmethode	
	Extrahiert	Nicht extrahiert	Extrahiert	Nicht extrahiert	Extrahiert	Nicht extrahiert
Bodenarten	5.841	139	5.433	547	4.381	1.599
Koordinaten	282	221	0	503	281	222

Tokenklassifikation

- Anstatt einem Paragraphen mit einem Label zu beschreiben, wird jedem Token ein eigenes Label zugewiesen
- Beispielsatz: „Es wird Getreide angebaut.“

Token	Es	wird	Getreide	angebaut
Label	Keine	Keine	Nutzpflanze	Keine

Probleme bei Trainingsdaten

- Mehr Trainingsdaten sind besser
  - Überspezialisierung: Erkennungsrate von 100% auf Trainingsdaten, 0% auf Nicht-Trainingsdaten
- Menge an relevanten Wörtern ist beschränkt
- Schwierig, mehr Koordinaten zu finden
- Idee: Vorhandene Paragraphen bearbeiten

Trainingsdatenerweiterung

- Paragraphen mit Koordinaten vor/nach den Koordinaten abschneiden
- Koordinaten mit Koordinaten aus anderen Paragraphen tauschen
- Komplett zufällige Koordinaten erzeugen
- Negativbeispiele:
  - Koordinaten permutieren („13 • 15 N“ → „•1 N35 1“)
  - Teilkoordinaten löschen („13 • 15 N“ →“1 5“)

Auswertung - Erweiterungen

Modifikationen	F1-Werte je Modell		
	Standard-Modell	Ersetzte Koordinaten-Modell	Randomisierte Koordinaten-Modell
Keine	0,056	0,527	0,154
Kürzen	0,090	0,564	0,161
Permutieren	0,000	0,687	0,388
Löschen	0,039	0,718	0,501
Beste Variante	0,213	0,799	0,556
Modifikationen der besten Variante	Kürzen Permutieren Löschen	Permutieren Löschen	Kürzen Löschen

Weitere Modifikationen

- Modifizierte Loss-Funktion, die Fehler bei Koordinaten stärker bestraft
  - In Kombination mit anderen Optionen gut
- Einzelne Label für Koordinatenanteile (Grad - N/S, Min - N/S, ...)
  - Hohe Fehlerrate
- Gewichte, um Vorkommen anzugleichen

Modifizierter Tokenizer

- Anderes Wörterbuch, um Koordinaten besser zerlegen zu können
- Zahlen können nur noch einzeln stehen
- Keine Verbesserung gegenüber dem Standardtokenizer

Multi-Task-Learning

- Statt binärer Klassifikation Einteilung in mehrere Klassen
- Gleichzeitige Auswertung aller zu betrachtenden Datenklassen
- Vorteil: Kürzeres Training und schnellere Anwendung
- Nachteil: Geringere Genauigkeit

Vergleich der Ansätze

Datenklasse	Multi-Task-Learning	Single-Task-Learning
Bodenart	0,971	0,980
Nutzpflanzen	0,976	0,989
Texturen	0,994	0,994
Koordinaten	0,678	0,000 (0,799)

Falsch-positive Bewertungen

- Bei Erzeugung der Trainingsdaten nicht gefundene Daten einer Datenklasse können dennoch zur Datenklasse gehören
- Manuelle Überprüfung der Falsch-Positiven nötig

Manuelle Überprüfung

- Texturen:
  - Die meisten tatsächlichen Fehler stehen in unvollständige Paragraphen (Konvertierungsfehler, <100 Zeichen)
  - „Doch-Positiv-Rate“: 58%

Manuelle Überprüfung

- Nutzpflanzen:
  - „Fresno“ (Spanisch für „Esche“, im Kontext Stadt)
  - „Blackwood“ (Flussname)
  - „Erin“, „Brewin“ (Namen)
  - „Brussels Sprout“ (Rosenkohl)
  - „Doch-Positiv-Rate“: 36%

Manuelle Überprüfung

- Bodenarten
  - „meso-neoprotozoic“ (Zeitalter)
  - „ceptisols“ (Teil von „Inceptisol“, ohne „In“ im Text)
  - „Hapolxerolls“ (Eigentlich: Haploxerolls)
  - „Doch-Positiv-Rate“: 42%

Manuelle Überprüfung

- Koordinaten
  - Einzelne Zahlen aus Jahreszahlen
  - Das Komma aus „type S-S, 80 W“
  - „35Ћ11J“ kann nicht in Token umgewandelt werden („[UNK]“), aber Teil von Koordinate
  - „Doch-Positiv-Rate“: ~75%

Ergebnis

- BERT ist gut geeignet, Metadaten in wissenschaftlichen Artikeln zu finden
- Die Extraktion benötigt kaum manuelle Nachbearbeitung
- Das Prinzip sollte auch mit anderen Datenklassen funktionieren