T. Grust et al. (Hrsg.): Datenbanksysteme für Business, Technologie und Web (BTW 2019),
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2019 337

Einsatz kognitiver Verfahren am Deutschen Patent- und Markenamt

Mark Reinke[1] André Kischkel[2] Volker Jahns[3] Uwe Crenze[4] Olga Beltcheva[5]

Abstract: Die Begutachtung von Patentanträgen ist ein aufwändiger Prüfprozess, dessen Ziel es ist, eine Entgegenhaltung zu finden. Dabei stellt die Verschleierung des Patentanspruchs durch gezielte Umschreibungen eine große Herausforderung dar. In enger Zusammenarbeit zwischen dem Deutschen Patent- und Markenamt und der interface projects GmbH entstand ein modernes Recherche- und Klassifikationssystem auf Basis von durch neuronale Netze gelernten Distributed Word Embeddings. Der Beitrag stellt verschiedene Verfahren zum Lernen von Word Embeddings vor und bewertet diese hinsichtlich ihrer Eignung für die Prüfung von Patentanmeldungen.

Keywords: DPMA, Patentprüfung, Recherche, Klassifikation, kognitive Verfahren, Word Embeddings

1 Einleitung

In den letzten fünf Jahren konnten erhebliche Fortschritte beim Einsatz von maschinellen Lernverfahren, basierend auf künstlichen neuronalen Netzen, verzeichnet werden. Im Bereich der Text- und Multimedia-Analyse wird in diesem Zusammenhang typischerweise von kognitiven Verfahren gesprochen. Hierzu zählen insbesondere Verfahren zur Klassifikation, Verschlagwortung, Erkennung relevanter Entitäten und zur Unterstützung einer semantisch-assoziativen Suche (kognitive Suche). Insbesondere profitieren systematische Rechercheprozesse und die Erschließung unbekannter Dokumente von solchen Verfahren.

Wie in kaum einem anderen Bereich gehört das systematische Recherchieren zur Kernaufgabe der Prüfer in Patentämtern. Die stark ansteigende Anzahl von Patentanmeldungen auf nationaler und internationaler Ebene stellt eine große Herausforderung bei der Bearbeitung der Schutzrechtsverfahren dar. Gleiches gilt für Patentabteilungen in Unternehmen und Forschungseinrichtungen. Erschwerend kommt hinzu, dass bei der Erstellung der Patentschriften gezielt Umschreibungen verwendet werden, welche die Nähe der angemeldeten Innovation zu bereits erteilten Patenten verschleiern und eine Entgegenhaltung zum angemeldeten

[1] interface projects GmbH, Zwinglistraße 11/13, 01277 Dresden, mark.reinke@interface-projects.de
[2] interface projects GmbH, Zwinglistraße 11/13, 01277 Dresden, andre.kischkel@interface-projects.de
[3] Deutsches Patent- und Markenamt, Referat 2.3.1, Zweibrückenstraße 12, 80331 München, volker.jahns@dpma.de
[4] interface projects GmbH, Zwinglistraße 11/13, 01277 Dresden, uwe.crenze@interface-projects.de
[5] Deutsches Patent- und Markenamt, Referat 2.3.1, Zweibrückenstraße 12, 80331 München, olga.beltcheva@dpma.de

doi:10.18420/btw2019-20

Patentanspruch verhindern soll. Die durch kognitive Verfahren durchgeführte Analyse der Patentschriften unterstützt die Patentprüfer bei der Aufdeckung solcher Verschleierungen, indem besser nach inhaltlich ähnlichen Patentschriften für entsprechende Entgegenhaltungen recherchiert werden kann. Eine andere Herausforderung besteht in der gezielten Zuordnung eines hinsichtlich der fachlichen Aspekte der Patentanmeldung erfahrenen Prüfers. Hier unterstützen kognitive Verfahren bei der Klassifikation der eingehenden Patentanmeldungen.

In diesem Beitrag werden verschiedene kognitive Verfahren vorgestellt, die im Rahmen der Zusammenarbeit zwischen dem Deutschen Patent- und Markenamt (DPMA) und der Firma interface projects GmbH auf der Basis des Produktes *intergator*[6] entwickelt und am DPMA eingeführt wurden.

1.1 Die Evolution der Text-Indexierung: von Term-Vektoren zu Distributed Word Embeddings

Für ein effizientes Text-Information-Retrieval wurden in den letzten Jahrzehnten zahlreiche Verfahren entwickelt. Weit verbreitet sind insbesondere sogenannte Vector-Space-Modelle (VSM), die nahezu jede Text-Suchmaschine verwendet - oft unter Anwendung des `TF-IDF`-Maßes zur Bestimmung der Relevanz der einzelnen Terme. Die Dimension der Vektoren wird durch die Anzahl unterschiedlicher Terme in dem zu durchsuchenden Korpus bestimmt. Ein Vektor repräsentiert ein Dokument im Korpus. Die Ähnlichkeit von Dokumenten kann leicht durch den Kosinus zwischen den Dokument-Vektoren bestimmt werden. Die Nachteile von Term-VSM liegen in der hohen Dimensionalität des Vektorraums. Zudem sind die Vektoren sehr schwach besetzt. Verfahrensbedingt geben diese Vektoren auch keinerlei Information über die semantische Ähnlichkeit von Dokumenten. Erfolgt eine Suchanfrage mit einem Begriff, der in den gesuchten Dokumenten nicht enthalten ist, erhält man keine Treffer. Trotz einer breiten Palette von weiteren Methoden des Natural Language Processing (NLP) können viele linguistische und semantische Aufgabenstellungen meist nicht befriedigend gelöst werden. Hinterlegte Wörterbücher und sonstige Begriffssysteme, einschließlich aufwendig erstellter Ontologien, können von den Nutzern in der Regel weder bereitgestellt, noch aktuell gehalten werden.

Mit den Arbeiten von Mikolov bei Google und später bei Facebook wurden die grundlegenden VSM unter Anwendung künstlich neuronaler Netze auf eine neue Stufe gehoben. Das unter `Word2Vec` [Mi13] 2013 bekannt gewordene Verfahren für *Distributed Word Embeddings* verwendet neuronale Netze zum Lernen von Zusammenhängen zwischen Wörtern. Ein Vektor repräsentiert dabei ein Wort in einem Vektorraum mit wenigen hundert Dimensionen. Räumlich benachbarte Vektoren beschreiben Wörter mit einem ähnlichen Kontext. Daraus ergeben sich vielfältige Anwendungsgebiete, wie die automatische Ermittlung von sinnverwandten Suchbegriffen, die Suche in fremdsprachigen Dokumenten und die Klassifikation von Dokumenten.

[6] `https://www.intergator.de`

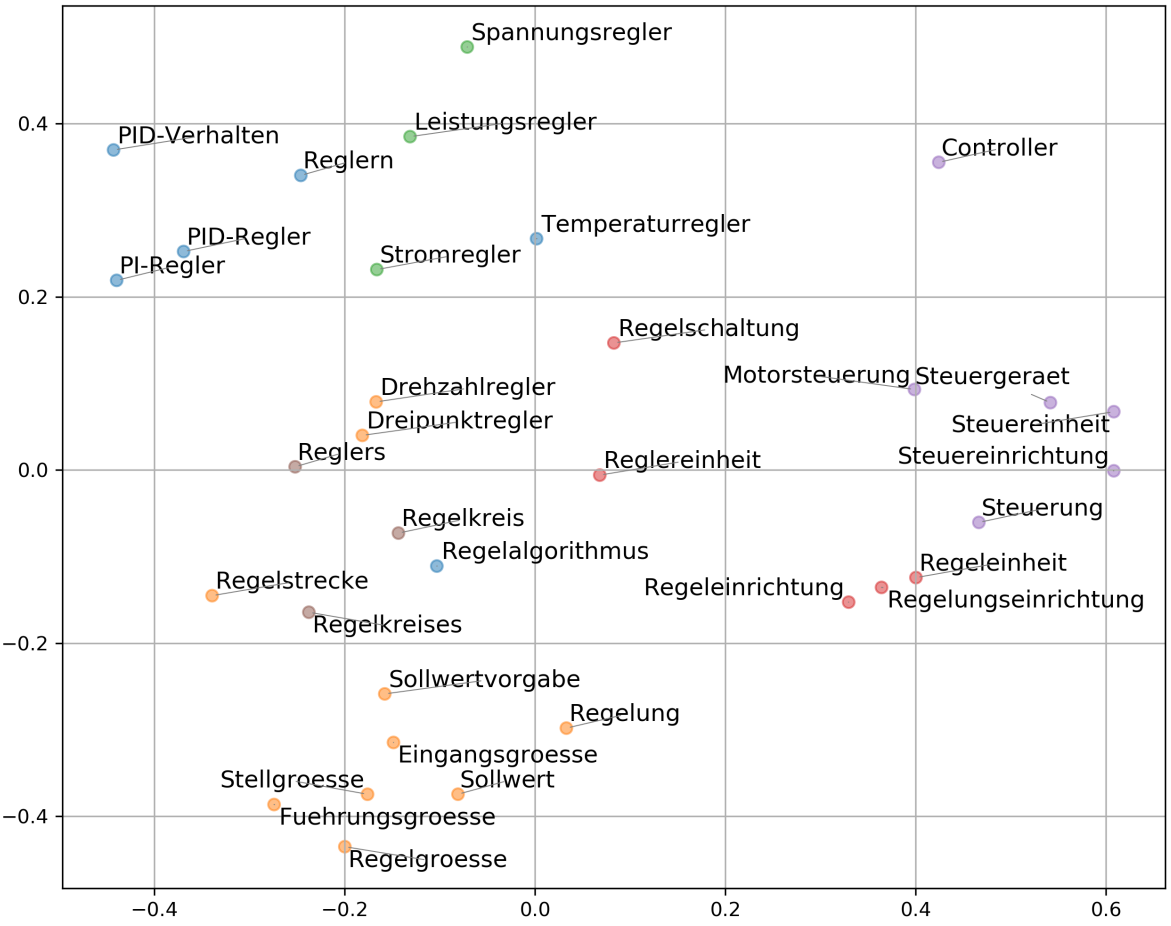

Abb. 1: Verwandte Begriffe zu *Regler*

Die Grafik Abb. 1 visualisiert den semantischen Kontext zum Begriff *Regler* in Form einer 2D-Projektion eines aus ca. 300.000 Patenten trainierten 300-dimensionalen Word-Embedding-Vektorraums.

Die Besonderheit dieser Word-Embedding-Vektoren liegt darin, dass ein Wort über die gelernten Kontexte, in denen es auftritt, beschrieben wird. Darüber hinaus sind Word-Embedding-Verfahren robust gegenüber unbekannten Begriffen in Dokumenten, die nicht in das Modell-Training einbezogen waren. Auf diese Weise können auch falsch geschriebene Wörter durch ihren Kontext erkannt und richtig eingeordnet werden. Word Embeddings liefern somit eine optimale Strategie für eine semantisch-assoziative (unscharfe) Suche. Letztendlich bildet die Gesamtheit der gelernten Word-Embedding-Vektoren ein Sprachmodell. Aus großen Korpora (z. B. Wikipedia, Zeitungs- und Nachrichtenarchiven u. ä.) lassen sich allgemeine Modelle vortrainieren und im Zusammenhang mit völlig anderen Textbeständen nutzen. Auch lassen sich Wort-Vektoren von einem sprachspezifischen Modell in ein anderes transformieren, um Suchen in fremdsprachigen Dokumentenbeständen zu ermöglichen.

Interessanterweise werden beim Erstellen der Word-Embedding-Vektoren auch linguistische Beziehungen zwischen Begriffen gelernt. Über Vektoroperationen lassen sich somit gezielt semantische Relationen ermitteln und für Frage-Antwort-Systeme oder Empfehlungssysteme einsetzen. Ein typisches Beispiel sind die schon in der ersten Veröffentlichung von Mikolov zu `Word2Vec` beschriebenen Wort-Analogien. Beispielsweise führt die Operation <König> - <Mann> + <Frau> zum Vektor <Königin>. Gleichermaßen lässt sich aus zwei Ländernamen und einer der Hauptstädte die Hauptstadt des zweiten Landes bestimmen.

Das maschinelle Lernen von Word Embeddings ist grundsätzlich unüberwacht. Das heißt, es müssen keine manuell kategorisierten Dokumente bereitgestellt werden. Über spezielle Erweiterungen von `Word2Vec` (z. B. bei `fastText` [Jo16]) lassen sich auch klassenspezifische Modelle trainieren, die im Anschluss von einem entsprechenden Klassifikator verwendet werden.

1.2 Lernen von Word Embeddings

Die Grundidee von Word Embeddings besteht darin, dass sich die Bedeutung eines Wortes aus den Wörtern erschließt, in deren syntaktischer Nähe es benutzt wird. Die Implementierung von Word-Embedding-Verfahren basiert hauptsächlich auf diesem linguistischen Phänomen. Einen umfassenden Überblick über das Lernen von Word Embeddings mit Hilfe neuronaler Netze wird in [Ko16] gegeben. Zum Erlernen solcher Kontext-Wort-Beziehungen werden zunächst Wort-Paare gebildet. Als Beispiel-Korpus dient folgendes Pangramm:

Zwölf Boxkämpfer jagen Viktor quer über den großen Sylter Deich.

Dieses Pangramm besteht aus 10 Wörtern, die unser Vokabular bilden. Der Kontext eines Wortes wird über ein Fenster aus Wörtern links und rechts um das Fokuswort gebildet. In unserem Beispiel gehen wir der Einfachheit halber von einem Wortfenster von 1 aus, d. h. der Kontext wird durch jeweils ein Wort links und eins rechts vom Fokuswort gebildet. So erhalten wir folgende (Kontext, Wort)-Paare:

([Zwölf, jagen], Boxkämpfer), ([Boxkämpfer, Viktor], jagen), ([jagen, quer], Viktor), ...

Im Folgenden wird das Lernen von Word Embeddings am Beispiel von `Word2Vec` erläutert. `Word2Vec` stellt zwei verschiedene Netzwerkarchitekturen bereit. Die erste Variante *Continuous Bag of Words* (`CBOW`) sagt zu einem Kontext das Fokuswort voraus. Die zweite Architekturvariante `Skip-Gram` funktioniert genau umgekehrt und sagt zu einem Wort den wahrscheinlichen Kontext voraus. Der Vorteil der `Skip-Gram`-Architektur liegt in einer höheren Robustheit bei der Erstellung von Wortvektoren für seltene Wörter. Dies wird allerdings mit einer gegenüber `CBOW` wesentlich höheren Berechnungszeit erkauft. Da im

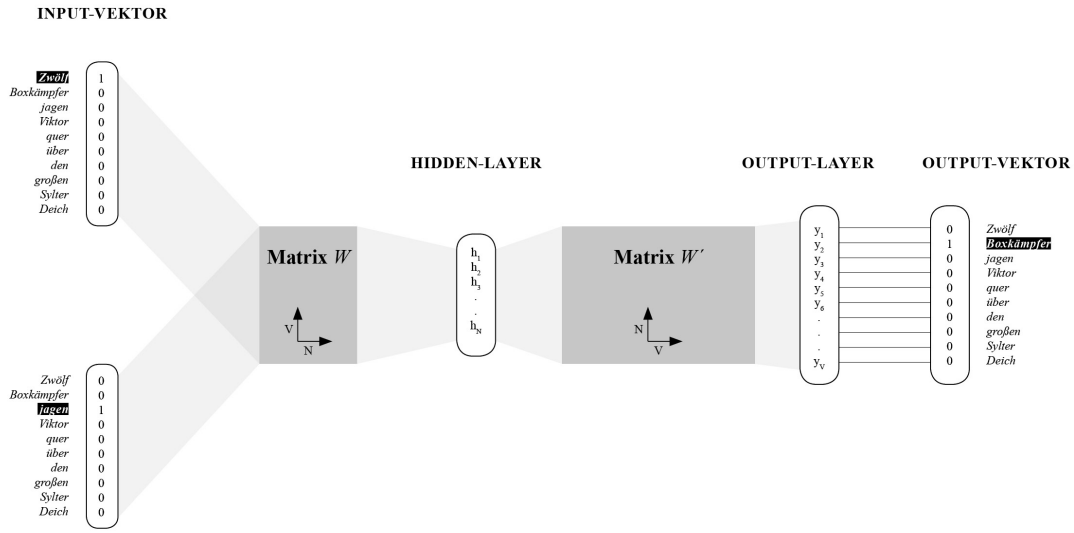

Abb. 2: Neuronales Netz mit `CBOW`-Architektur

vorliegenden Anwendungsfall zwischen `CBOW` und `Skip-Gram` kein signifikanter Unterschied in der Qualität der Ergebnisse festgestellt werden konnte, wurde aus Performance-Gründen `CBOW` verwendet.

Bei `Word2Vec` `CBOW` besitzen die neuronalen Netze je einen Input-Vektor für die Kontext-Wörter, einen Hidden-Layer mit N linearen Neuronen (Dimension des Modell-Vektorraums) und einem Output-Layer, der durch einen Softmax-Klassifikator ergänzt wird und das Fokuswort in einem Output-Vektor repräsentiert. Die Dimension der Input- und Output-Vektoren wird durch die Größe des Vokabulars (V) des zu untersuchenden Korpus bestimmt.

In unserem einfachen Beispiel besteht der Korpus aus einem einzigen Dokument mit einem Vokabular von lediglich 10 Wörtern. Normalerweise besteht ein Korpus aus vielen Dokumenten, die insgesamt ein Vokabular von tausenden von Wörtern umfassen. Die Input-Vektoren für die beiden Kontextwörter und der Output-Vektor für das Fokuswort werden durch sogenannte *One-Hot-Vectors* gebildet, die bis auf die Position des jeweiligen Kontext- bzw. Fokuswortes ausschließlich aus Nullen bestehen.

Für die abstrakte Repräsentation der aus den Kontextwörtern gelernten Bedeutung des Fokuswortes wird ein Vektor in einem N-dimensionalen Vektorraum erzeugt, wobei N typischerweise zwischen 300 und 500 liegt. Hierfür besitzt das neuronale Netzwerk einen Hidden-Layer, dessen N Neuronen der Spalten einer Gewichtsmatrix $W_{(VxN)}$ entsprechen. Die Matrix besteht aus V Reihen (für jedes Wort des Vokabulars).

Der Output-Layer besteht aus einer zweiten Gewichtsmatrix $W_{(NxV)}$ und einem Softmax-Regressionsklassifikator für die Rekonstruktion des Output-(Fokuswort)-Vektors. Jedes Output-Neuron besitzt einen Gewichtsvektor, der mit dem korrespondierenden Vektor im Hidden-Layer multipliziert wird. Das Ergebnis ist die Wahrscheinlichkeit, mit der das dem Output-Neuron zugeordnete Wort des Vokabulars ein Fokuswort zum gegebenen Kontext ist. Da ähnliche Wörter in ähnlichen Kontexten stehen, werden dementsprechend auch ähnliche Gewichtsvektoren erlernt.

1.3 Weiterentwicklung von Word Embeddings

Durch die Möglichkeit von Word Embeddings die Semantik und Syntaktik von Wörtern als Vektoren zu repräsentieren, eignen sie sich hervorragend als Grundbausteine für verschiedenste NLP-Anwendungen.

Durch Abwandlung und Erweiterung der ursprünglichen Architektur und Einbeziehung weiterer Techniken des maschinellen Lernens inklusive geschickter Optimierungen können Word Embeddings auch als Basis für Information-Retrieval-Tasks, wie Textklassifikation oder Textähnlichkeitsaufgaben, dienen. Facebook AI Research stellt mit `fastText` eine effiziente Erweiterung von `Word2Vec` zur Verfügung. Durch die Möglichkeit, Vektoren von *N-Grammen* auf Zeichen- oder Wortebene zu bilden, können auch unbekannte Wörter (Out-of-vocabulary) und Mehrwortbegriffe einbezogen werden. Weiterhin überträgt `fastText`, durch eine leichte Abwandlung der `CBOW`-Architektur, das Konzept der Embeddings auf Klassifikationsprobleme.

Im Folgenden wird die Anwendung und Evaluierung der hier dargestellten Verfahren anhand der Patentrecherche und Patentklassifikation am Deutschen Patent- und Markenamt vorgestellt.

2 Patentrecherche

Die Patentrecherche ist ein wesentlicher Bestandteil des Patenterteilungsverfahrens am DPMA. Im Rahmen der Patentprüfung ermittelt der Prüfer zu einer Patentanmeldung den relevanten Stand der Technik. Dies ist eine Liste relevanter Dokumente der Patent- und Nichtpatentliteratur, die sogenannten Entgegenhaltungen. Der Prüfer entscheidet anhand des Rechercheergebnisses, ob die Grundvoraussetzung für eine Patenterteilung, das Beruhen der angemeldeten Erfindung auf Neuheit und erfinderische Tätigkeit, gegeben ist.

Die Patentrecherche mit dem hauseigenen Deutschen Patentinformationssystem (DEPATIS) basiert bis dato auf der Suche nach Termen und bibliographischen Daten von Patentschriften. Die Suchtechnologie beruht auf einer Booleschen Suche. Die Suchbegriffe werden exakt oder trunkiert eingegeben und verwandte Begriffe müssen aus manuell gepflegten

Wortlisten übernommen werden. Die Suche nach bibliographischen Daten, wie zum Beispiel diversen Klassifikationssymbolen (IPC, CPC oder FI), Namen von Erfindern oder Anmeldern, Datumsbereichen wie Anmeldedatum, Publikationsdatum oder Prioritätsdaten, Familienmitgliedern, Entgegenhaltungen und zitierenden bzw. zitierten Schriften, liefert Dokumente, in denen exakt diese bibliographischen Daten vorkommen.

In der Praxis ist die Bildung von komplexen und langen Suchanfragen notwendig, weil die Kombination von Begriffen und Synonymen explizit angegeben werden muss. Die Suche soll eine Liste von Dokumenten liefern, die in einer angemessenen Zeit vom Prüfer gesichtet und bewertet werden kann. Das schnelle Anwachsen der Anzahl publizierter Patentschriften erschwert die Lage des Prüfers. Die Formulierung solcher Suchanfragen ist sehr zeitaufwändig, fehlerträchtig und erfordert langjährige Erfahrung auf dem Prüfgebiet. Da Anmeldungen oft allgemein formuliert werden, um den Wirkungsgrad des Patentes zu erhöhen (*Kleinfahrzeug* anstatt *Fahrrad*) und der Anmeldegegenstand durch die Verwendung abseits vom Stand der Technik liegender Begriffe, Umschreibungen (*im Kreis bewegen* anstatt *rotieren*) oder Wortneuschöpfungen (*Müllentsorgungsdrohne*) meist verschleiert ist, muss der Prüfer den Text inhaltlich erschließen und die Bedeutung des Textes intellektuell erfassen.

2.1 Kognitive Suche

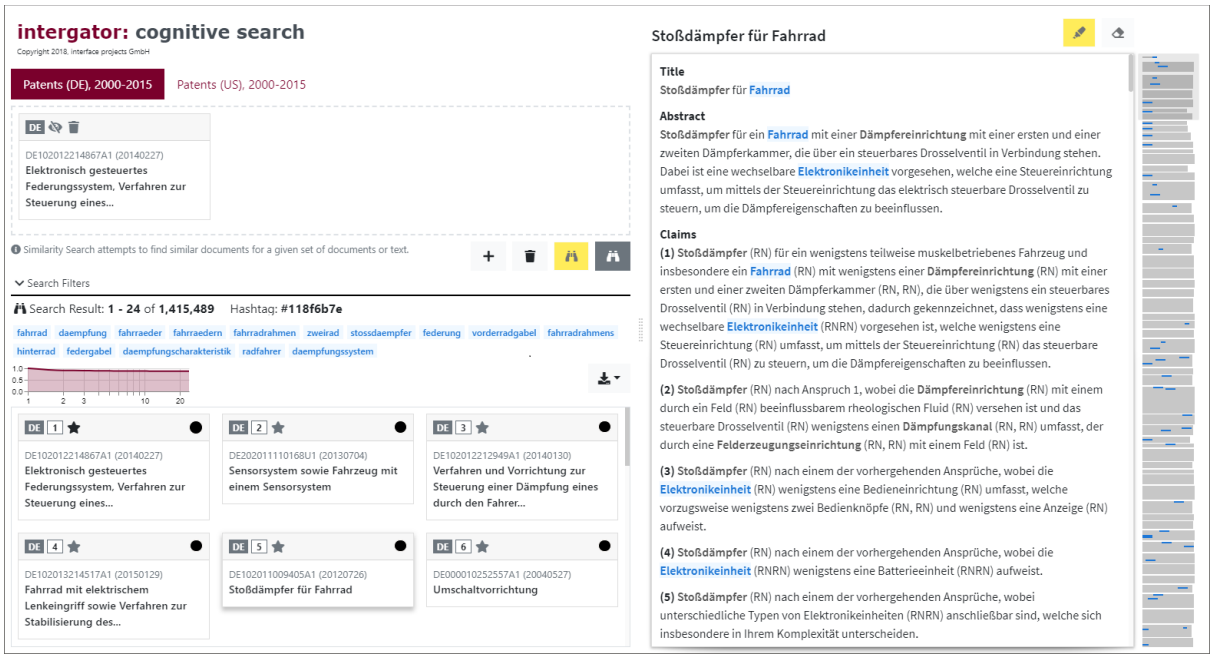

Abb. 3: Benutzeroberfläche der kognitiven Suche für die Patentrecherche

Die kognitive, auf semantischer Textähnlichkeit beruhende Suche reduziert den manuellen Aufwand und ermöglicht eine einfache, effiziente und effektive Recherche. Es werden folgende Funktionen bereitgestellt:

1. Die kognitive Suche ist in der Lage, inhaltlich ähnliche Dokumente zu einer Patentanmeldung ohne manuelle Eingabe zu finden. Es wird eine Auswahlliste von Dokumenten als nächstliegender Stand der Technik präsentiert. Eine automatisierte Patent-Vorsuche wird damit möglich.
2. Es ist sowohl möglich, relevante Abschnitte einer Anmeldung für die Suche auszuwählen, als auch in Kombination mit anderen relevanten Druckschriften zu suchen. So ist es beispielsweise möglich, ausgehend von einzelnen Ansprüchen der Anmeldung oder in Kombination mit dem in der Anmeldung zitiertem Stand der Technik zu suchen.
3. Bei der Suche werden Synonyme oder sinnverwandte Begriffe gefunden, die in weiteren Recherchen verwendet werden können. Die intellektuelle Ermittlung von Synonymen für die Formulierung der Suchanfrage bzw. Erstellung und Pflege von Synonym-Listen ist nicht mehr erforderlich.
4. Die Bewertung der Suchergebnisse wird durch Hervorheben von Begriffen, die eine semantische Nähe zur Eingabe aufweisen erleichtert.

Das Neuartige an dieser semantischen Suchtechnologie ist die Identifizierung von Patentliteratur mit ähnlicher Bedeutung. Ein ähnliches Dokument kann auch dann von der Suchmaschine als relevant angesehen werden, wenn die Begriffe der Sucheingabe nicht darin vorkommen. Diese Funktionalität ist für die Patentrecherche besonders wichtig, da für die Neuartigkeit eines Patentes die darin beschriebenen Ideen entscheidend sind und nicht die Begriffe, die verwendet werden.

Word Embeddings repräsentieren semantische und syntaktische Eigenschaften von Worten und sind damit eine geeignete Basistechnologie für die kognitive Suche. Die Idee der Word Embeddings ist auch auf größere Textabschnitte anwendbar. Diese sogenannten *Sentence Embeddings* bilden die Bedeutung von Phrasen oder Sätzen auf Vektoren ab. Damit lassen sich, wie bei Word Embeddings, Zusammenhänge als mathematische Operationen ausdrücken. Einige solcher Verfahren werden im Folgenden untersucht und es wird gezeigt, dass sich die Idee der Sentence Embeddings auch auf ganze Dokumente erweitern lässt (*Document Embeddings*). Eine Suchoperation kann dann als Vektorähnlichkeit zwischen Sucheingabe-Vektor und Dokumentvektoren im Suchraum ausgedrückt werden.

Die *Sentence Embedding*-Verfahren lassen sich in zwei Kategorien unterteilen: einfache, flache neuronale Netzwerke (`Word2Vec`, `SIF` [ALM17], `Sent2Vec` [PGJ17]) und komplexere, tiefe neuronale Netze (`Skip-Thoughts Vectors` [Ki15], `InferSent` [Co17]).

Wir beschränken uns hier auf flache neuronale Netze, da für das Training auf großen Datenmengen ein effizienter, ressourcenschonender Algorithmus benötigt wird und einfache Netzwerke bei ähnlichen Problemstellungen vergleichbar gute Ergebnisse erzielen konnten wie komplexere Architekturen (Arora et al. [ALM17], Pagliardini et al. [PGJ17], Hill et

al. [HCK16]). Die Entwicklung im Bereich *Deep Learning* ist jedoch rasant und eine zukünftige Betrachtung solcher Verfahren sinnvoll.

Für die kognitive Suche werden Embeddings verwendet, die mit dem `Sent2Vec`-Algorithmus erlernt wurden. Einzelne Wörter und ganze Dokumente werden in denselben Vektorraum projiziert und können durch einfache räumliche Entfernungsmetriken in Verbindung gesetzt werden. Mit diesem universellen Ansatz eröffnen sich neue Möglichkeiten. So realisieren wir damit, als Teil der kognitiven Suchanwendung, eine Reihe unterschiedlichster Information-Retrieval-Aufgaben:

Suche (Wörter ⇒ Dokumente): Hierbei macht man sich die Tatsache zu Nutze, dass der Zentroid der Wortvektoren der Sucheingabe räumlich nahe bei den semantisch ähnlichen Dokumenten liegt.

Dokumentähnlichkeit (Dokument ⇒ Dokumente): Semantisch ähnliche Dokumente liegen im Vektorraum nahe beieinander.

Automatische Synonyme (Wort ⇒ Wörter): Semantisch ähnliche Worte liegen im Vektorraum nahe beieinander.

Automatische Verschlagwortung (Dokument ⇒ Wörter): Schlagwörter eines Dokuments befinden sich in räumlicher Nähe zu den Dokumentvektoren.

Durch Matrix-Rotation werden die Vektorräume verschiedener Sprachmodelle zueinander ausgerichtet und lassen sich dadurch kombiniert verwenden [Co18]. Alle beschriebenen Aufgabenstellungen sind somit auch sprachübergreifend möglich.

2.2 Evaluation

Es gibt einige Veröffentlichungen, in denen *Sentence Embedding*-Verfahren für Aufgaben wie Sentimentanalyse, Textklassifikation oder Satzähnlichkeit evaluiert wurden. Soweit uns bekannt ist, wurden Sentence Embeddings noch nicht auf ihre Eignung für Suchanwendungen getestet. Es wurden in diesem Projekt zwei Tests entwickelt - der Titel-Test, um die Suche und der Entgegenhaltungstest, um die Dokumentähnlichkeit zu evaluieren. Folgende Verfahren werden miteinander verglichen:

`T-VSM TF-IDF`: Als Baseline dient das *Term Vector Space Model* (`T-VSM`) mit `TF-IDF`-Maß. Es wird die *More Like This Query* von Elasticsearch[7] verwendet, um die relevantesten Begriffe eines Textes zu ermitteln und ODER-verknüpft nach diesen zu suchen.

[7] `https://www.elastic.co/guide/en/elasticsearch/reference/current/query-dsl-mlt-query.html#_how_it_works`

Word2Vec: Die einfachste Methode Sentence Embeddings zu erhalten, ist die Mittelung aller mit Word2Vec gelernten Word Embeddings eines Textes. Eine Suche nach ähnlichen Dokumenten lässt sich als Suche nach den ähnlichsten Dokument-Vektoren zum Durchschnittsvektor der Sucheingabe ausdrücken.

SIF: Bei *Smooth Inverse Frequency* (SIF) erfolgt die Durchschnittsbildung mit gewichteten Wortvektoren, bei der relevante Wörter bevorzugt werden. Zudem werden die ersten Prinzipalkomponenten der Dokument-Vektoren entfernt (*Common Component Removal*), um semantisch unbedeutende Aspekte aus den Embeddings zu entfernen.

Sent2Vec: Sent2Vec ist eine Erweiterung von fastText. Ähnlich wie bei CBOW werden Wortvektoren trainiert, indem Wörter auf Grund ihres Kontextes vorhergesagt werden. Word2Vec verwendet ein Kontextfenster mit parametrisierbarer Maximallänge, Sent2Vec den gesamten Satz unter Erhalt aller Wort-N-Gramme. Die trainierten Wortvektoren eignen sich besonders gut für die Durchschnittsbildung langer Eingabetexte.

2.2.1 Titel-Test

In Ermangelung eines Goldstandards wurde ein automatisiertes Testverfahren konzipiert, mit dem eine „unscharfe“ Stichwortsuche simuliert wird. Beim *Titel-Test* werden die Patentschriften in Titel und Volltext (Zusammenfassung, Ansprüche, Beschreibung) separiert. Es wird nach dem Titel gesucht und evaluiert, ob das System das zugehörige Dokument als relevantesten Suchtreffer zurückliefert. Um zu testen, ob ein Text aufgrund seiner semantischen Ähnlichkeit zur Suchanfrage gefunden werden kann, werden in einem Vorverarbeitungsschritt alle Begriffe der Titel aus dem Volltext entfernt. Für diesen Test werden 10.000 deutsche Patentschriften als Datenbasis verwendet. Tab. 1 zeigt für jedes Verfahren den erreichten Recall in Prozent. Bei *Recall @1* wird getestet, ob das erste gefundene Dokument das gesuchte ist. Bei *Recall @10* muss das gesuchte Dokument unter den ersten 10 Treffern sein.

Verfahren	Recall @1	Recall @10
T-VSM TF-IDF	-	-
Word2Vec	10,8	29,6
SIF	28,9	61,0
Sent2Vec	**44,1**	**76,6**

Tab. 1: Die Ergebnisse des Titel-Tests

Sent2Vec ist den anderen Verfahren deutlich überlegen. Der Test (Tab. 1) zeigt, dass eine kognitive Suche, die nicht vom Vorkommen der Suchwörter in den Dokumenten abhängt, sondern auf inhaltlicher Ähnlichkeit basiert, möglich ist. Eine Suche mit Term-VSM kann keine korrekten Ergebnisse liefern, da die Eingabewörter aus den Dokumenten entfernt wurden.

2.2.2 Entgegenhaltungstest

In diesem Test wird evaluiert, wie gut ein System zu einer vorgegebenen Patentschrift ähnliche Schriften finden kann. Dafür wird die Tatsache genutzt, dass zitierte Literatur in den bibliografischen Daten der Patente hinterlegt wird. Dieser Test spiegelt den realen Anwendungsfall der Patentrecherche wieder. Für die Evaluation werden ca. 1,5 Millionen deutsche Patent- und Offenlegungsschriften der Jahrgänge 2000 - 2015 verwendet. Als Eingabe werden 2.630 Schriften betrachtet, für die alle mindestens eine zitierte Schrift im Datenbestand enthalten ist.

Verfahren	Recall @10	Recall @100	Recall @1000	Trainingszeit	Laufzeit
`T-VSM TF-IDF`	10,4	21,1	31,2	-	1h 52m
`Word2Vec`	0,6	12,3	20,6	**18h**	**1m 39s**
`SIF`	16,6	41,0	**69,8**	**18h**	3m 10s
`Sent2Vec`	**17,6**	**43, 3**	69,4	30h	4m 40s

Tab. 2: Ergebnisse des Entgegenhaltungstests

Die Eignung von `Sent2Vec` für die Patentrecherche wird durch diesen Test (siehe Tab. 2) bestätigt. Wie beim Titel-Test bringt eine einfache Mittelung von `Word2Vec`-Vektoren keine guten Ergebnisse und schneidet schlechter ab als die Baseline. Mit `SIF` und `Sent2Vec` können hochwertige Dokumentvektoren erzeugt werden, die die semantischen Eigenschaften von Dokumenten abbilden.

2.3 Fallbeispiel

Am Beispiel der Patentschrift *„Elektronisch gesteuertes Federungssystem, Verfahren zur Steuerung eines Federungssystems und Computerprogramm“* (DE102012214867) soll veranschaulicht werden, wie mittels kognitiver Suche das entgegengehaltene Patentdokument *„Stoßdämpfer für Fahrrad“* (DE102011009405) trotz unterschiedlicher Terminologie gefunden werden konnte. Während mit einer klassischen Stichwortsuche die Entgegenhaltung nicht ermittelt werden kann, ist die Ähnlichkeit der Dokument-Vektoren hoch (unter den Top-3 Treffern im Entgegenhaltungstest). In der qualitativen Betrachtung wird ersichtlich, dass sich, obwohl keine große textuelle Übereinstimmung der Patente besteht, die Texte dennoch inhaltlich ähnlich sind, wie im Folgenden anhand der Textauszüge zu sehen ist.

Auszug aus DE102012214867:

> Die Erfindung betrifft ein **elektronisch gesteuertes Federungssystem** für ein **Fahrrad** (1), enthaltend zumindest einem **Federelement** (3, 4), welches zwischen einem ersten Teil (10) des **Fahrrades** und einem zweiten Teil (14, 15) des **Fahrrades** (1) angeordnet ist, welche beweglich miteinander **verbunden** sind, wobei zumindest eine Kenngröße des **Federelementes** veränderbar ist, und zumindest einen Aktor (431), welcher auf das **Federelement** (3, 4) einwirkt, um die zumindest eine Kenngröße zu verändern, und ein **Elektronikmodul** (6), mit welchem ein Ansteuersignal für den zumindest einen Aktor (431) erzeugbar ist, wobei weiterhin ein **Steuerelement** (2, 63, 66) vorhanden ist, mit welchem das vom **Elektronikmodul** (6) erzeugte Ansteuersignal beeinflussbar ist, wobei das **Steuerelement** (2, 63) mit dem **Elektronikmodul** (6) über ein **Funksignal** (64) verbindbar ist und/oder der Aktor (431) mit dem **Elektronikmodul** (6) über ein Funksignal (64) verbindbar ist. Weiterhin betrifft die Erfindung ein entsprechendes Verfahren zur **Steuerung** eines **Federungssystems** für ein **Fahrrad** und ein Computerprogramm zu dessen Durchführung.

Auszug aus DE102011009405:

> **Stoßdämpfer** für ein **Fahrrad** mit einer **Dämpfereinrichtung** mit einer ersten und einer zweiten **Dämpferkammer**, die über ein steuerbares Drosselventil in Verbindung stehen. Dabei ist eine wechselbare **Elektronikeinheit** vorgesehen, welche eine **Steuereinrichtung** umfasst, um mittels der **Steuereinrichtung** das elektrisch steuerbare Drosselventil zu steuern, um die **Dämpfereigenschaften** zu beeinflussen.

Dieses Beispiel ist typisch für die Vielfältigkeit der im Patentwesen verwendeten Fachsprache. Die folgende Tabelle zeigt die Verwendung unterschiedlicher, aber sinnverwandter Begriffe in den zwei Patenten:

DE102012214867	DE102011009405
Elektronikmodul, elektronisch	Elektronikeinheit, elektrisch
Steuerelement, Steuerung, gesteuert	Steuereinrichtung, steuerbar
Federungssystem, Federelement	Stoßdämpfer, Dämpfereinrichtung, Dämpferkammer

Tab. 3: Unterschiede im Vokabular zweier ähnlicher Patentschriften

3 Patentklassifikation

Die Patentklassifikation dient dem einheitlichen Klassifizieren von Patentdokumenten nach technischen Gebieten. Durch die ordnungsgemäße Einordnung von Patentdokumenten in eine Klassifikationsstruktur wird dem Patentprüfer der Zugriff zu der darin enthaltenen technischen Information erleichtert. Es existieren verschiedene Klassifikationsschemata, wie zum Beispiel die internationale Patentklassifikation (IPC), die Cooperative Patent Classification (CPC), die Klassifikation des Japanischen Patentamts (FI) und weitere. Alle Klassifikationsschemata besitzen eine hierarchische Baumstruktur. Im Prüfungsverfahren am DPMA werden Patentdokumente einheitlich nach der IPC klassifiziert, aber es werden auch andere Schemata berücksichtigt.

Die Klassifikation dient dem DPMA vor allem zur Unterstützung der Recherche, um Patentdokumente wieder aufzufinden, damit für die technischen Offenbarungen in Patentanmeldungen die Patentfähigkeit festgestellt werden kann. Eine korrekte und möglichst feingranulare Klassifizierung der Patentliteratur ist für die Arbeit des Patentprüfers somit entscheidend. Mit dem ständig wachsenden Stand der Technik wird auch die Patentklassifikation fortlaufend erweitert.

Die international harmonisierte Klassifikation IPC unterteilt sich von der obersten zur untersten Hierarchiestufe in Sektionen, Hauptklassen, Unterklassen, Hauptgruppen und Untergruppen. Ein Klassifikationseintrag besteht aus einem IPC-Symbol und einer textuellen Beschreibung [DP18]. Die IPC mit dem Revisionsstand 2018.01 enthält über 74.000 vollständige IPC Symbole: 642 Symbole bis zur Unterklasse und über 8.000 Symbole bis zur Hauptgruppe.

```
Sektion A: Täglicher Lebensbedarf
Sektion B: Arbeitsverfahren; Transportieren
Sektion C: Chemie; Hüttenwesen
Sektion D: Textilien; Papier
Sektion E: Bauwesen; Erdbohren; Bergbau
Sektion F: Maschinenbau; Beleuchtung; Heizung; Waffen; Sprengen
Sektion G: Physik
Sektion H: Elektrotechnik
```

Abb. 4: Die acht Fachgebiete (Sektionen)

Die Klassifikation von Patentanmeldungen in die IPC kann durch ein automatisiertes Klassifikationssystem erfolgen [BG11]. Der technische Gegenstand des Textes muss durch diesen Klassifikator möglichst genau erfasst und dann zum richtigen Knoten in die hierarchische Struktur der IPC klassifiziert werden.

Die möglichen Einsatzgebiete im DPMA werden im Wesentlichen durch folgende Anwendungsfälle beschrieben:

Elektronische Vorklassifikation: Die Verteilung der Patent- und Gebrauchsmusteranmeldungen im DPMA erfolgt anhand der vergebenen IPC-Klassifikation. Diese Vorklassifikation ist der Schritt im Patentverfahren, bei dem bereits möglichst genau die IPC-Stelle und somit das genaue Themengebiet der Anmeldung ermittelt werden soll. Zudem können weitere Nebenklassen der IPC falls notwendig vergeben werden. Die manuelle Vorklassifikation ist intellektuell sehr aufwändig. Der Inhalt des Anmeldungstextes muss von der klassifizierenden Person erfasst und sehr gut verstanden werden. Weiterhin muss eine große Vertrautheit mit der IPC-Klassifikation gegeben sein, um einen möglichst genauen Klassifikationsvorschlag zu liefern.

Der Schritt der Vorklassifikation kann durch einen automatischen Klassifikator unterstützt werden. Dazu muss die Anmeldung in elektronisch verarbeitbarer Form vorliegen. Im Jahre 2011 wurden die bisher an die Papierform gebundenen Patentakten durch ein vollelektronisches Verfahren ersetzt. Damit werden sämtliche in Papierform eingehenden Patentanmeldungen vollständig digitalisiert. Weiterhin können Patentanmeldungen über die webbasierte Plattform DPMAdirekt in elektronischer Form eingereicht werden. Diese Grundlage ermöglicht es einem automatischen Klassifikator, die in elektronischer Form vorliegenden Texte zu analysieren und Klassifikationsvorschläge zu erstellen. Anhand des IPC-Vorschlags wird die neu angelegte elektronische Patentakte automatisch an einen Patentprüfer weitergeleitet, welcher den IPC-Vorschlag prüft.

Interaktive Klassifikation: Ein weiteres Einsatzgebiet eines elektronischen Klassifikators ist die interaktive Klassifikation. Der Klassifikator übernimmt eine Assistenzfunktion und unterstützt die Patentprüfer bei der Vergabe der IPC-Symbole. Hierbei muss der Klassifikator verschiedene Dokumente interaktiv verarbeiten und Klassifikationsvorschläge erstellen. Durch die direkte Interaktion des Prüfers mit dem Klassifikator sollte es auch möglich sein, die Vorschläge auf bestimmte Bereiche der IPC, beispielsweise auf bestimmte Unterklassen, einzuschränken und Rückmeldungen zur Qualität einzugeben.

Umklassifizierung: Die IPC befindet sich in ständiger Überarbeitung. Jährliche Revisionen machen es erforderlich, die bestehende Klassifizierung von Patentdokumenten zu korrigieren. Für diesen Zweck werden bei jeder IPC-Revision Konkordanzlisten herausgegeben, welche ein Umschreiben der Klassifikation erleichtern. Häufig kommt es jedoch vor, dass IPC-Klassen zu umfangreich geworden sind oder sich die bisherige Einteilung als zu grob herausgestellt hat. In diesem Fall werden IPC-Symbole in weitere Untergruppen untergliedert oder teilweise sogar auf unterschiedliche Klassen aufgeteilt. Solche Fälle können nicht mehr regelbasiert umklassifiziert werden, sondern erfordern eine manuelle Neuklassifizierung. Hierfür muss, ähnlich wie bei der Vorklassifikation, der Inhalt des Dokuments von einem Prüfer beurteilt werden, um anschließend eine geeignete neue Klasse vergeben zu können. Zudem kann bei der Patentprüfung auffallen, dass die Anmeldung mehrere technische Aspekte hat, die während der ersten Patentklassifikation übersehen wurden oder tatsächlich anders zu gewichten sind.

Prüfstoffpflege: Als Prüfstoff wird der Gesamtbestand des im DPMA archivierten Standes der Technik bezeichnet. Dazu zählen sowohl sämtliche nationale Patent- und Gebrauchsmusterschriften, als auch internationale Patentdokumente von anderen Patentämtern. Gepflegt und verwaltet werden diese Dokumente in elektronischer Form im Deutschen Patentinformationssystem (DEPATIS). Die Informationen werden der Öffentlichkeit über den Onlinedienst DEPATISnet zur Verfügung gestellt. Die Qualität des Prüfstoffs mit seinen Volltexten und bibliographischen Daten ist sowohl für das DPMA selbst, als auch für seine Anmelderschaft und Informationskunden von großer Bedeutung. Die Prüfer müssen im Rahmen ihrer Tätigkeit den Prüfstoff aus deutschen, europäischen, aber auch asiatischen und amerikanischen Patent- und Offenlegungsschriften sichten, eingruppieren und können gegebenenfalls eine Änderung der IPC-Hauptklasse vornehmen. Die beschriebene Prüfstoffpflege ist eine kontinuierliche Aufgabe, die von jedem Prüfer Sorgfalt und einen gewissen Zeitaufwand fordert, da er jedes zu verifizierende Dokument intellektuell durchdringen muss, um die korrekte Einordnung in die IPC vornehmen zu können. Hier kann ein automatischer Klassifikator eine unterstützende Funktion zur Validierung, Ergänzung oder Umklassifikation der bestehenden Klassifizierung bieten.

Weitere Anwendungsfälle: Einige ausländische Ämter klassifizieren nicht nach IPC. Hier ist eine Neuklassifikation nach IPC notwendig, ähnlich dem Anwendungsfall der interaktiven Klassifikation. Da die internationalen Schriften in unterschiedlichen Sprachen abgefasst sind, ist es erforderlich, dass der Klassifikator auch nach Fremdsprachen, wie Englisch oder Französisch, klassifizieren kann. Neben der Klassifikation nach dem IPC-Schema kann der Prüfer die Dokumente auch in dem von der IPC abgeleiteten erweiterten Klassifikationsschema des DPMA (DEKLA) einordnen, welches am DPMA für die Suche im Prüfstoff verwendet wird. Zusätzlich zur Vergabe eines IPC-Symbols für eine deutsche Patentanmeldung sollte deshalb als eine weitere Funktionalität eines automatischen Klassifikators auch ein DEKLA-Symbol vergeben werden können.

Die beschriebenen Anwendungsfälle erfordern einen Text-Klassifikator, der die IPC-Klasse einer gegebenen Patentanmeldung oder -schrift vorhersagen kann. Verglichen mit anderen Klassifkationsproblemen, gibt es hier besondere Herausforderungen:

1. Der Wortschatz der Patentliteratur ist auf Grund seiner technischen Natur sehr umfangreich und mehrdeutig.
2. Potentiell sind sehr viele Kategorien zu unterscheiden. Diese sind hierarchisch angeordnet, so dass sie thematisch überlappen und sich in tieferen Ebenen nur noch durch spezifische Teilaspekte unterscheiden.
3. Die unterschiedliche Relevanz von Themengebieten im Stand der Technik spiegelt sich in der IPC-Hierarchie wieder. Die Patentschriften sind daher ungleich über die

Kategorien verteilt. So gibt es Kategorien, denen nur ein Patent zugeteilt ist, während andere tausende Patente umfassen.

Der folgende Auszug aus der IPC-Hierarchie zeigt, wie die thematische Unterscheidung beim Abstieg in den Baum zunehmend feiner wird und auf unterster Ebene meist nur noch begriffliche Feinheiten eine Rolle spielen (z. B. Vorderrad, Hinterrad).

```
B: Sektion B Arbeitsverfahren; Transportieren
B62: Gleislose Landfahrzeuge
B62K: Fahrräder; Motorräder; Rahmen; Lenkvorrichtungen;
      vom Fahrer betätigte Steuerungsvorrichtungen; Achsaufhängungen;
B62K 25/00: Achsaufhängungen
B62K 25/04: . zum Anbau von Achsen federnd am Fahrradrahmen oder an der -gabel
B62K 25/06: . . mit Teleskopgabel, z.B. einschließlich schwingender Hilfsarme
B62K 25/08: . . . für das Vorderrad
B62K 25/10: . . . für das Hinterrad
```

Abb. 5: Auszug aus der IPC-Hierarchie

Unser Klassifikationssystem beruht auf dem `fastText Supervised`-Verfahren [Jo16]. Es handelt sich dabei um eine Abwandlung des `Word2Vec` `CBOW`-Modells. Während `Word2Vec` ein unüberwachtes Lernverfahren ist, das nur Text als Eingabe erhält, ist `fastText Supervised` ein überwachtes Verfahren, das auf Sätzen und den zugehörigen Kategorien trainiert wird. Bei `CBOW` besteht die Lernaufgabe in der Vorhersage eines Zielwortes unter Berücksichtigung der umgebenden Wörter. Bei `fastText Supervised` hingegen wird nicht ein Zielwort, sondern die Kategorie zu einem gegebenen Satz vorhergesagt.

Das Klassifikationssystem adressiert die gegebene Aufgabenstellung in folgender Weise:

1. Word Embeddings reduzieren die Dimension der für die Klassifikation benötigten Features um Größenordnungen im Vergleich zum *Term Vector Space Model*, das klassische Klassifikationsalgorithmen verwendet.
2. Mit dem *Negative Sampling*-Trick ([Mi13], [Jo16]) wird die Zeit-Komplexität reduziert und bleibt mit zunehmender Anzahl an Klassen konstant. Negative Sampling ist eine Annäherung der *Softmax*-Funktion. Bei jedem Trainingsschritt wird nur eine zufällige Auswahl an Negativ-Beispielen (Kategorien) betrachtet.
3. Die Trainingsdokumente werden über die IPC-Hierarchie ausbalanciert. Dabei werden die IPC-Symbole den Trainingsdokumenten dynamisch zugewiesen: Je mehr Dokumente in einen Zweig des IPC-Baums fallen, desto tiefer wird in den Baum abgestiegen. Der Klassifikator betrachtet die Hierarchie nicht. Die Dokumente werden flach klassifiziert, da sich eine hierarchische Klassifikation für die IPC nicht bewährt hat [Ba13].

3.1 Evaluation

Für die Klassifikation werden ca. 300.000 deutsche Patent- und Offenlegungsschriften der Jahrgänge 2010 - 2015 verwendet. Es werden nur der Titel und die Textfelder (Zusammenfassung, Ansprüche, Beschreibung) der Schriften herangezogen. Die Dokumente wurden in ein Trainings- und Testset unterteilt (90/10). Kategorien mit wenigen Beispieldokumenten wurden von der Klassifikation ausgeschlossen.

Wir vergleichen `fastText Supervised` mit einer klassischen, linearen *Support Vector Machine* (`SVM`). Für den `SVM`-Algorithmus wird die Implementierung von *liblinear* [Fa08] verwendet. Zusätzlich wird der für Anwendungsfälle dieser Größenordnung ausgelegte lineare Lernalgorithmus von `Vowpal Wabbit` [La07] evaluiert. Bei der Evaluation wird auch die Trainingszeit bewertet, da Performance für die Umsetzung der Patentklassifikation ein maßgeblicher Faktor ist.

3.1.1 Klassifizierung auf Unterklassen-Ebene

Ziel der Klassifikation ist es, jeder Schrift die jeweils korrekte Unterklasse aus **584** Unterklassen zuzuweisen.

Verfahren	Accuracy @1	Accuracy @3	Trainingszeit	Laufzeit
`SVM`	**70,0**	**89,5**	1h 5m	**18s**
`Vowpal Wabbit`	69,0	86,8	25m	5m
`fastText Supervised`	68,5	87,1	**17m**	38s

Tab. 4: Ergebnisse der Klassifikation auf Unterklassen-Ebene

Der Vergleich zeigt, dass `fastText Supervised` und `Vowpal Wabbit` etwas schlechter abschneiden als die `SVM`. Die Unterschiede im Ressourcenbedarf sind jedoch signifikant. Im Gegensatz zu `fastText Supervised` sind Trainingszeit und Speicherauslastung bei der `SVM` wesentlich höher.

3.1.2 Klassifizierung mit dynamischer Verteilung

Eine Klassifikation bis zur Unterklassen-Ebene der IPC reicht für die Vorklassifikation beim DPMA nicht aus. Je tiefer in die IPC-Hierarchie abgestiegen wird, desto besser kann eine Anmeldung dem richtigen Prüfer zugeordnet werden. Wünschenswert wäre daher eine feinere Klassifizierung nach Haupt- oder Untergruppe. Es erfolgt eine Ausbalancierung der Kategoriezuteilung im Trainingsset, dies führt zu einer Unterscheidung von **4.448** Kategorien (Unterklassen, Haupt- und Untergruppen).

Verfahren	Accuracy @1	Accuracy @3	Trainingszeit	Laufzeit
`SVM`	-	-	-	-
`Vowpal Wabbit`	41,4	58,6	6h 31m	24m
`fastText Supervised`	**68,2**	**83,8**	**17m**	**1m 10s**

Tab. 5: Ergebnisse der Klassifikation mit dynamischer Verteilung

Mit steigender Anzahl von Kategorien kommen die Vorteile von `fastText Supervised` deutlich zum Tragen. Das Hinzunehmen von Klassen wirkt sich auf Grund von Negative Sampling nicht nachteilig auf Speicher- und Zeitkomplexität aus. Die Genauigkeit fällt nur leicht ab. Für die *liblinear*-Implementierung der `SVM` ist die Komplexität des Problems nicht mehr beherrschbar. `Vowpal Wabbit` benötigt im Vergleich zur Klassifikation von Unterklassen mehr Zeit, um zu konvergieren und fällt in der Genauigkeit deutlich ab.

Der `fastText Supervised`-Algorithmus skaliert hervorragend und eröffnet dadurch die Möglichkeit, mit vielen Kategorien und auf großen Datenbeständen schnell zu trainieren.

4 Ausblick

Es wurde gezeigt, dass kognitive Verfahren mächtige Instrumente liefern, um die Patentrecherche und die Patentklassifikation effektiver und effizienter zu gestalten.

Naturgemäß geht bei unscharfen, assoziativen Verfahren die Exaktheit verloren. So findet zum Beispiel eine Suche nach *Fahrrad* auch Dokumente zu *Kleinfahrzeugen* oder *Motorrädern*. Je nach Anwendungsfall kann dieser Effekt gewollt oder ungewollt sein. Im hochdimensionalen Raum sind viele verschiedene syntaktische und semantische Aspekte kodiert, die Grenzen zwischen ähnlich und unähnlich fließend. In diesem Zusammenhang muss über alternative Darstellungsformen und neue Ausdrucksmöglichkeiten für den Nutzer, zum Beispiel durch interaktive graphische Benutzeroberflächen, nachgedacht werden. Durch eine Interaktion mit dem Nutzer wird seine Rechercheintention deutlich und er kann mit seiner Fachkompetenz die Vorschläge des Systems optimieren. Hierin liegt der Schwerpunkt künftiger Weiterentwicklungen.

Literatur

[ALM17] Arora, S.; Liang, Y.; Ma, T.: A simple but tough-to-beat baseline for sentence embeddings, 2017, URL: `https://openreview.net/pdf?id=SyK00v5xx`, Stand: 17. 10. 2017.

[Ba13] Babbar, R.; Partalas, I.; Gaussier, E.; Amini, M. R.: On Flat versus Hierarchical Classification in Large-Scale Taxonomies. In (Burges, C. J. C.; Bottou, L.; Welling, M.; Ghahramani, Z.; Weinberger, K. Q., Hrsg.): Advances in Neural Information Processing Systems 26. Curran Associates, Inc., S. 1824–1832, 2013, URL: `http://papers.nips.cc/paper/5082-on-flat-versus-hierarchical-classification-in-large-scale-taxonomies.pdf`.

[BG11] Benzineb, K.; Guyot, J.: Automated Patent Classification. In: Current Challenges in Patent Information Retrieval. Springer-Verlag, Berlin Heidelberg, 2011.

[Co17] Conneau, A.; Kiela, D.; Schwenk, H.; Barrault, L.; Bordes, A.: Supervised Learning of Universal Sentence Representations from Natural Language Inference Data, 2017, URL: https://arxiv.org/abs/1705.02364v5, Stand: 08. 07. 2018.

[Co18] Conneau, A.; Lample, G.; Ranzato, M.; Denoyer, L.; Jégou, H.: Word Translation Without Parallel Data, 2018, URL: https://arxiv.org/abs/1710.04087v3.

[DP18] DPMA: Internationale Patentklassifikation, Handbuch zur IPC Ausgabe 2018. Deutsches Patent- und Markenamt, München, 2018.

[Fa08] Fan, R.-E.; Chang, K.-W.; Hsieh, C.-J.; Wang, X.-R.; Lin, C.-J.: LIBLINEAR: A Library for Large Linear Classification. J. Mach. Learn. Res. 9/, S. 1871–1874, Juni 2008, ISSN: 1532-4435, URL: http://dl.acm.org/citation.cfm?id=1390681.1442794.

[HCK16] Hill, F.; Cho, K.; Korhonen, A.: Learning Distributed Representations of Sentences from Unlabelled Data, 2016, URL: https://arxiv.org/abs/1602.03483v1, Stand: 10. 02. 2016.

[Jo16] Joulin, A.; Grave, E.; Bojanowski, P.; Mikolov, T.: Bag of Tricks for Efficient Text Classification, 2016, URL: https://arxiv.org/abs/1607.01759v3, Stand: 09. 08. 2016.

[Ki15] Kiros, R.; Zhu, Y.; Salakhutdinov, R.; Zemel, R. S.; Torralba, A.; Urtasun, R.; Fidler, S.: Skip-Thought Vectors, 2015, URL: https://arxiv.org/abs/1506.06726v1, Stand: 22. 06. 2015.

[Ko16] Korger, C.: Clustering of Distributed Word Representations and its Applicability for Enterprise Search, Diploma Thesis, Dresden University of Technology, 2016.

[La07] Langford, J. et al.: Vowpal Wabbit, 2007, URL: http://hunch.net/?p=309.

[Mi13] Mikolov, T.; Chen, K.; Corrado, G.; Dean, J.: Efficient Estimation of Word Representations in Vector Space, 2013, URL: https://arxiv.org/abs/1301.3781v3, Stand: 07. 09. 2013.

[PGJ17] Pagliardini, M.; Gupta, P.; Jaggi, M.: Unsupervised Learning of Sentence Embeddings using Compositional n-Gram Features, 2017, URL: https://arxiv.org/abs/1703.02507v2, Stand: 17. 10. 2017.