Martin-Luther-Universität Halle-Wittenberg
Institut für Informatik
Studiengang Informatik, B.Sc.

Kontextabhängige Termgewichtung für Total-Recall-Suchen

Bachelorarbeit

Wilhelm Beiche

1. Gutachter: Prof. Dr. Matthias Hagen
2. Gutachter: Ferdinand Schlatt

Datum der Abgabe: 14. Dezember 2021

Erklärung

Hiermit versichere ich, dass ich diese Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Halle, 14. Dezember 2021

. . . . . .
Wilhelm Beiche

Zusammenfassung

Bei einem systematischen Review sollen alle relevanten Dokumente für ein Informationsbedürfnis in eine Analyse aufgenommen werden. Das Screening innerhalb von systematischen Reviews kann Monate in Anspruch nehmen, da es wichtig ist, alle verwandten Dokumente für das Informationsbedürfnis zu finden. Wir benutzen einen Machine-Learning-Ansatz, um das Screening für das systematische Review durchzuführen. Dafür wird ein logistisches Regressionsmodell auf dem expliziten Relevanzfeedback der Nutzer trainiert. Dieses explizite Relevanzfeedback wollen wir einsetzen, um das Ranking der verbleibenden Dokumente zu verbessern. Die Dokumente werden von dem logistischen Regressionsmodell als TF-IDF-Vektoren betrachtet. Wir stellen ein System vor, welches die Termhäufigkeiten von Termen, die in ihrem Kontext wichtig sind, erhöht und die Termhäufigkeiten von Termen, die in ihrem Kontext unwichtig sind, verringert. Die Wichtigkeit der Wörter in dem Kontext werden dabei von einem neuronalen Sprachmodell bestimmt, welches wir mit einem Feintuning-Ansatz für unsere Aufgabe trainiert haben. Für das Feintuning unserer Modelle entwickelten wir drei unterschiedliche Strategien, mit denen die Relevanzlabel für das neuronale Sprachmodell, anhand der internen und externen Dokumentstruktur bestimmt werden. Die trainierten Modelle werden genutzt, um die Dokumentrepräsentationen kontextabhängig zu gewichten. Die erzeugten kontextabhängigen Dokumentrepräsentationen werden dann im Screening verwendet, um das bereits bestehende logistische Regressionsmodell weiter zu verbessern. Wir vergleichen unsere erzeugten kontextabhängigen Dokumentrepräsentationen mit der ungewichteten Dokumentrepräsentation beim Screening von systematischen Reviews anhand von drei simulierten systematischen Reviews über die Themen Umwelt und Genom Editierung bei Pflanzen. Die Evaluation zeigt, dass die erzeugten Dokumentrepräsentationen in der Lage sind, den Aufwand für das systematische Review bei hohen Recall-Werten, reduzieren zu können.

Inhaltsverzeichnis

Kapitel 1

Einführung

Im Jahr 2015 wurden rund zwei Millionen neue wissenschaftliche Artikel veröffentlicht [1]. Diese Informationsflut an neuen Erkenntnissen muss gesammelt, ordentlich zusammengefasst und kritisch überprüft werden, sodass sich z. B. niedergelassene Ärzte, Kliniker und Wissenschaftler, über ein medizinisches Themengebiet umfassend und aktuell informieren können. Darum werden unter anderem in der Medizin und in der Biologie regelmäßig systematische Reviews verfasst. Weitere Anwendungszwecke von systematischen Reviews sind die Zusammenfassung der Forschungshistorie zu einem Informationsbedürfnis, die Entdeckung von noch bestehenden Forschungslücken und eine Möglichkeit neue Forschungsansätze zu finden [2].
Der erste Schritt eines systematischen Reviews ist die Sammlung von wissenschaftlichen Artikeln zu einem Themengebiet. Beim nächsten Schritt wird im Screening überprüft, ob die zur Verfügung stehenden Artikel relevant zum Themengebiet sind. Der Screening-Prozess muss für jedes systematische Review durchgeführt werden, da die herkömmliche Ad-Hoc-Suche zu ungenau ist. Um das Screening für ein systematisches Review durchzuführen, muss deshalb für jedes systematische Review eine große Ansammlung von Dokumenten durchgelesen und auf die Relevanz zum Themengebiet bewertet werden. Von jedem relevanten Dokument werden dann die wichtigen Informationen extrahiert, sodass diese im systematischen Review kritisch evaluiert und im Zusammenhang mit der anderen relevanten Forschung präsentiert werden kann.
In dieser Arbeit beschäftigen wir uns mit der Verbesserung des Screening-Prozesses bei systematischen Reviews. Wir betrachten in unserer Arbeit die Anwendung von kontextabhängigen Termgewichtungen für das Screening bei systematischen Reviews. Das Ziel unserer Arbeit ist es, den Zeitaufwand beim Erstellen des Screenings zu verringern.
Wir verwenden für die Durchführung des Screenings einen Machine-Learning basierten Ansatz [3]. Dieser benutzt ein logistisches Regressionsmodell, wel-

ches auf dem expliziten Relevanzfeedback des Nutzers trainiert wird. Der Sinn des Modells ist es, vorherzusagen, wie relevant ein Dokument zu einem Informationsbedürfnis ist. Das Modell gibt dann die wahrscheinlich relevantesten Dokumente an einen menschlichen Gutachter, der wiederum die tatsächliche Relevanz der Dokumente beurteilt. Die gekennzeichneten Dokumente werden dann zum Trainingsset des Modells hinzugefügt, um dieses weiter zu verbessern. [4]
Bei dem State-of-the-Art System werden die Termhäufigkeiten der Dokumentterme der relevanten Dokumente benutzt, um das logistische Regressionsmodell zu trainieren. Die Termhäufigkeiten sind aber in ihrer Effektivität begrenzt, da sie nicht die Bedeutung der Wörter im Kontext einer Passsage zuverlässig darstellen können. So besitzen Homonyme (z. B. *Strauß*) die gleiche Termhäufigkeit, obwohl sie eine unterschiedliche Bedeutung im Kontext ihrer Passage besitzen. Deswegen haben wir mit Entwicklungen im Natural Language Processing (NLP) versucht, die Termhäufigkeiten mit der Bedeutung der Terme im Kontext des Satzes zu erweitern. Wir benutzen ein vortrainiertes Sprachmodell, um die Terme kontextuell zu gewichten. Für das Finetuning des Sprachmodells haben wir drei Strategien entwickelt, mit denen wir unterschiedliche Gewichtungen von Termen im Kontext untersuchen können. Zusätzlich haben wir untersucht, welche Dokumentfelder am besten geeignet sind, um das Screening effektiv mit HiCal durchzuführen. Eine Übersicht über die verwandte Arbeit zu unserer Forschung ist im Kapitel 2 zu finden. Im dritten Kapitel stellen wir die verwendeten Technologien vor, sowie unsere eigenen Strategien, um das Finetuning des verwendeten neuronalen Sprachmodells zu bestimmen. Im 4. Kapitel evaluieren wir unsere Strategien und wir vergleichen unterschiedliche Dokumentrepräsentationen, um deren Effektivität beim Screening von systematischen Reviews festzustellen. Im letzten Abschnitt wird die Arbeit zusammengefasst und es wird eine Übersicht erstellt, wie die Total-Recall-Suchen mit kontextabhängigen Termgewichtungen möglicherweise noch verbessert werden könnte.

Kapitel 2

Related Work

Total-Recall-Suchen werden verwendet, um alle relevanten Dokumente zu einem Informationsbedürfnis zu finden [5].Es gibt unterschiedliche Anwendungsmöglichkeiten für Total-Recall-Suchen, wie eDiscovery [6], systematische Reviews [7] oder die Entwicklung von Suchmaschinen [8]. Die vielen Anwendungsmöglichkeiten für Total-Recall-Suchen führten zu unterschiedlichen Ansätzen, wie das Total-Recall-Problem gelöst werden kann. Diese Ansätze sind entweder suchbasiert oder Machine-Learning basiert [9]. Wir untersuchen in diesem Abschnitt zuerst suchbasierte Methoden und dann Machine-Learning-Modelle die dabei helfen, einen hohen Recall zu erreichen, sodass der Aufwand für das Screening von systematischen Reviews verringert werden kann. Danach beschreiben wir die Grundlagen aus dem Bereich des Natural Language Processing, welche für die kontextualisierten Termgewichtungen eingesetzt werden.
Bei suchbasierten Methoden müssen gewöhnlich die Forscher, die das Screening durchführen, die Suchanfrage manuell bearbeiten. Dabei ist das Ziel, eine oder mehrere modifizierte Suchanfragen an die Suchmaschine zu stellen, um viele relevante Dokumente für ihr Informationsbedürfnis zu erhalten.
Eine suchbasierte Methoden, die verwendet wird, um den Recall zu erhöhen, ist die Modifizierung der Suchanfrage mit booleanschen Operatoren. Dabei werden Operatoren wie AND, OR oder NOT in die Suchanfrage eingefügt, sodass mehr relevante Dokumente zu einem Informationsbedürfnis gefunden werden können. Forscher nutzen spezielle Suchoperatoren, um Suchanfragen für Total-Recall zu bauen [10]. Das Bauen dieser Anfragen ist jedoch zeitaufwendig, und gleichzeitig ist die Effektivität dieser Anfragen zum Teil eingeschränkt [11]. Zum Beispiel konnten Blair und Maron zeigen, dass bei der Durchführung des Screening-Prozesses mit booleanschen Operatoren nur 20 % Recall erreicht wurde. Um den Recall zu erhöhen und um die Komplexität von den Suchanfragen nicht zu stark zu erhöhen, kann Interactive Search and Judging verwendet werden. Interactive Search and Judging erzielt einen höheren Recall

mit weniger Aufwand als mit den booleanschen Operatoren.
Beim Interactive Search and Judging (ISJ) werden mehrere Anfragen an eine Ad-Hoc-Suchmaschine gestellt und deren Ergebnisse evaluiert. Durch das Reformulieren der Anfrage, sowie mit der Betrachtung von den Resultaten der Anfragen ist es möglich den Recall zu steigern. Cormack et al. [11] zeigen, dass es möglich ist, mit ISJ, die meisten relevanten Dokumente in einem Testdatensatz zu erkennen. Dabei verglichen die Autoren bei ihrem Ansatz das ISJ mit dem Relevanzset von NIST, bei dem die Relevanz der Dokumente schon vorher bekannt war. Das Experiment zeigte, dass es möglich ist, die Relevanz der Dokumente herauszufinden, mit einem kleineren Aufwand für den Screening-Prozess, als wenn alle Dokumente manuell durchsucht werden müssten.
Bei dem Screening für systematische Reviews mithilfe von ISJ wird das Feedback der Nutzer verwendet, um die Anfrage zu reformulieren, sodass ein höherer Recall erzielt wird. Auch bei Machine-Learning basierten Ansätzen wird das Feedback der Nutzer verwendet, um einen höheren Recall mit wenig Aufwand zu erzielen.
In 2009 wurde ein Machine-Learning-Ansatz von Settles et al. vorgestellt, um relevante Dokumente aus einem Pool von noch nicht gekennzeichneten Dokumenten zu extrahieren. Als ersten Schritt werden dabei relevante Dokumente mithilfe von ISJ aus dem Datensatz extrahiert. Die gekennzeichneten relevanten Dokumente werden dann verwendet, um ein aktives Lernmodell zu trainieren. Dieses Modell wählt dann Dokumente aus, die noch manuell gekennzeichnet werden müssen. Mithilfe der weiteren verfügbaren relevanten Dokumente kann das aktive Lernmodell stetig verbessert werden.
Es gibt unterschiedliche Methoden dem Nutzer, die ungekennzeichneten Dokumente zur manuellen Überprüfung vorzulegen. Lewis et al. [12] haben eine Studie durchgeführt, bei der drei Methoden untersucht und miteinander verglichen wurden. Bei dem unsicheren Sampling werden dem Nutzer schwierig zu klassifizierende Dokumente vorgelegt. Beim relevanten Sampling wählt der Nutzer die am wahrscheinlichsten relevanten Dokumente aus. Im zufälligen Sampling werden dem Nutzer zufällige Dokumente präsentiert.
Die drei Sampling-Methoden wurden bei einem Textklassifizierungsansatz für Titel von Zeitungsartikeln eingesetzt. Die Resultate der Studie zeigen, dass die Effektivität vom unsicheren Sampling und dem relevanten Sampling bei großen Datensätzen ungefähr gleich ist. Das zufällige Sampling ist weniger effektiv, als die anderen beiden Sampling-Methoden, jedoch kann mithilfe des zufälligen Samplings ein sehr guter Klassifizierer schon auf kleinen Datensätzen trainiert werden.
Ein Ansatz, um das bereits bestehende aktive Lernmodell [13] zu verbessern, wurde von Cormack and Mojdeh vorgestellt. Diese haben das bereits bestehende Aktive-Lernmodell mit einem kontinuierlich aktiven Lernmodell erweitert

[14]. Hierbei wurden zur Initialisierung des kontinuierlich aktiven Lernmodells relevante Dokumente mithilfe von Interactive Search and Judging aus dem Datensatz gesucht. Das Modell wird dann mithilfe der gefundenen relevanten Dokumente trainiert. Das trainierte Modell bewertet dann die übrigen ungekennzeichneten Dokumente nach ihrer Relevanz. Die Dokumente werden dann wie beim relevanten Sampling den Nutzern zur manuellen Überprüfung der Relevanz zur Verfügung gestellt. Die manuell gekennzeichneten Dokumente werden dann zum Trainingssets des kontinuierlich aktive Lernmodells hinzugefügt, sodass es kontinuierlich weiter verbessert wird.
In 2014 haben Cormack und Grossman das aktive Lernmodell(SAL) mit dem kontinuierlichen aktiven Lernmodell (CAL) verglichen [4]. Diese haben das für das SAL-Modell ein unsicheres Sampling mit k-Dokumenten verwendet und für CAl das relevante Sampling mit k-Dokumenten. Das k steht dafür, wie viele Dokumente vom Modell zur manuellen Überprüfung der Relevanz übergeben werden. Cormack und Grossman fanden heraus, dass ihr CAL-Modell einen geringeren Aufwand erzielt, als das SAL-Modell bei unterschiedlichen Recall-Werten. Da es zeitaufwändig ist, für jede Initialisierung des Modells relevante Dokumente mit ISJ zu finden, haben Cormack und Grossman eine Verbesserung ihres CAL-Modells veröffentlicht [15]. Beim Auto-TAR-Modell ist es nicht mehr notwendig, mit ISJ die relevanten Dokumente zur Initialisierung des Modells zu extrahieren. Stattdessen wird eine Beschreibung des Informationsbedürfnisses genutzt und 100 zufällige ungekennzeichnete Dokumente als Initialisierung des Modells. In 2017 wurde eine Evaluation mit dem AutoTar-Modell beim Screening eines systematischen Reviews durchgeführt [16]. Dabei stellte sich heraus, dass das Auto-TAR-Modell einen höheren Recall mit weniger Aufwand erreicht als die herkömmliche Methode mit ISJ. Im TREC-Total-Recall-Track von 2015 haben Teilnehmer versucht, mit ihren Ansätzen Total-Recall zu erreichen [5]. Das Ziel von dem Total-Recall-Track ist es unterschiedliche Retrievalmodelle zu evaluieren, die versuchen einen möglichst hohen Recall zu erreichen.
Als Baseline für die Evaluation der unterschiedlichen Verfahren wurde eine Baseline Model Implementation (BMI) verwendet [17]. Die BMI-Methode nutzt ein logistisches Regressionsmodell, welches auf dem iterativen Relevanzfeedback von den Nutzern trainiert wird. Dabei wird den Nutzern das am wahrscheinlich relevanteste Dokument gezeigt, welches nach der Relevanzbewertung zum Trainingsset vom Modell hinzugefügt wird. Die BMI-Methode ist die Implementation der Auto-TAR-Methode.
Zhang et al. haben in ihrer Arbeit die Baseline Model Implementation verwendet und diese weiter verbessert [18]. Dabei haben sie in einer Studie untersucht, ob der Aufwand des Screenings reduziert werden kann, indem man dem Nutzer nicht die gesamten Dokumente zur Relevanzbewertung übergibt, sondern nur

einzelne isolierte Sätze, da deren Relevanzbewertung weniger Zeit in Anspruch nimmt. Aus der Evaluation ihrer Ansätze hat sich ergeben, dass meist schon Sätze aus relevanten Dokumenten genug sind, um ausreichend die Relevanz des Dokumentes beschreiben. Somit stellen die Autoren fest, dass mit ihrer Methode das Screening von systematischen Reviews effizienter durchgeführt werden kann, als mit der BMI-Methode.
Die BMI-Methode benutzt ein logistisches Regressionsmodell, welches auf dem explizitem Relevanzfeedback der Nutzer trainiert wird. Das Modell benutzt dabei TF-IDF-Vektoren, um die Neugewichtung durchzuführen. Ein Experiment zeigt [19], dass diese Vektoren bei Ad-Hoc-Suchen schlechtere Rankings erzeugen als kontextabhängige Termgewichtungen. Deswegen beschreiben wir in den nächsten Absätzen, welche Verbesserungen das Natural Language Processing für die Gewichtung des logistischen Regressionsmodells bereitstellen kann.
Beim Natural Language Processing wird untersucht, wie natürliche Sprache in Form von Text- oder Sprachdaten mithilfe des Computers algorithmisch verarbeitet werden kann. Über viele Jahre setzen Data Scientists zur Textanalyse und -klassifikation hauptsächlich auf sogenannte Bag-of-Words Modelle beziehungsweise die davon abgeleiteten TF-IDF Modelle[20]. Bag-of-Words Modelle betrachten ausschließlich die Termhäufigkeiten der Terme innerhalb der Dokumente. Das Bag-of-Words Modell vernachlässigt die Reihenfolge und die Grammatik von Termen im Text.
2013 schaffte es Mikolov et al. in ihrer Arbeit, die Semantik von Termen im Text mit Wortvektoren zu erfassen [21]. Die Autoren haben ein neuronales Netz trainiert, mit dem Wörter als Wortvektoren dargestellt werden können. Die Methode um dies durchzuführen heißt word2vec. Diese Wortvektoren sind in der Lage syntaktische und semantische Eigenschaften der Wörter untereinander zu beschreiben.
Das Problem der Wortvektoren ist, dass Text meist eine grammatikalische Struktur, besitzt, weswegen die Bedeutung von Termen im Kontext unterschiedlich interpretiert werden kann. Die Bedeutung eines Wortes ist meist erst aus dem Kontext erschließbar und nicht aus dem Wort selbst. [22] Zum Beispiel besitzen Homonyme wie *Strauß* meist mehr als eine Bedeutung. Der nächste logische Schritt ist die Erschließung des Kontextes von den Wörtern untereinander mit neuronalen Sprachmodellen, sodass die Wortvektoren erneut mit der Gewichtung des Kontexts der Terme erweitert werden können. Diese erweiterten Wortvektoren nennen wir kontextualisierte Worteinbettungen.
Mithilfe von kontextualisierten neuronalen Sprachmodellen lassen sich State-of-the-Art Resultate erzielen, zum Beispiel im Sprachmodellierung [23] oder im Sprachparsing [24]. Diese Sprachmodelle können in der Medizin verwendet werden, um automatisch Notizen und Texte in elektronischen Gesundheitsakten auszuwerten [25].

Die Sprachmodelle sind in der Lage, die Beziehung zwischen Termen im Kontext untereinander zu erfassen. Einer der ersten neuronalen Sprachmodelle ist ELMo [26]. ELMo betrachtet Sätze im Text als kontextualisierte Einheiten. Dafür benutzt das neuronale Netz LSTMs [27], um die Sätze zu analysieren. Dabei wurden von ELMo Sätze vorwärts und rückwärts trainiert, um die kontextualisierten Worteinbettungen so gut wie möglich zu erzeugen. ELMo besitzt aber trotzdem einige Nachteile. Die Trainingszeit von ELMo ist sehr hoch und zusätzlich muss für jede neue Aufgabe das Modell neu trainiert werden [20]. Deswegen wurde der Ansatz von ELMo überarbeitet. BERT ist ein weiteres vortrainiertes Sprachmodell, was verwendet wird, um aus Text kontextualisierte Worteinbettungen zu erhalten [22]. Dabei verwendet BERT wie ELMo Wortvektoren, um diese kontextuell zu gewichten. Für BERT wird ein neuronales Netz trainiert, welches darauf ausgelegt ist den nächsten Satz vorherzusagen. Es werden bestimmte Wörter beim Training von BERT vernachlässigt, sodass diese beim Finetuning wieder hinzugefügt werden können. Mithilfe des Hinzufügens dieser Wörter verändert sich der Kontext des Satzes und diese Tatsache wird beim Finetuning verwendet, um das Sprachmodell BERT bei unterschiedlichen Anwendungsmöglichkeiten einzusetzen.
Mithilfe der kontextualisierten Worteinbettungen kann das Finetuning vom Sprachmodell auf spezifische Aufgaben, wie Sprachmodellierung oder Sprachparsing ausgeführt werden. Es gibt unterschiedliche Versionen von BERT, die auf unterschiedlichen Datensätzen trainiert wurden und somit unterschiedliche Anwendungszwecke besitzen wie SciBERT [28] oder BioBERT[29]. Es gibt bereits bestehende Finetuning-Ansätze für den BERT-Encoder, die bei Ad-Hoc-Suchen verwendet werden.
Ein Finetuning-Ansatz wurde von Santos et al. entwickelt[30]. In dem vorgestellten Versuch wurde die Auswirkung von kontextualisierten Dokumentrepräsentationen bei biomedizinischen Literaturrecherchen untersucht. Mithilfe der erzeugten Dokumentrepräsentationen soll das Retrieval der biomedizinischen Dokumente verbessert werden. Die Autoren haben in ihrer Arbeit die Nutzung von Textabschnitten aus dem Abstract der biomedizinischen Dokumente zum Finetuning von vortrainierten kontextbezogenen Sprachmodellen betrachtet. Als kontextbezogenes Sprachmodell wurde das Sprachmodell SciBERT verwendet, welches auf wissenschaftlichen Artikeln vortrainiert wurde. In ihrem Experiment haben sie untersucht, inwiefern das Finetuning von dem SciBERT-Sprachmodell die Abschnittsstruktur von den biomedizinischen Abstracts effektiv berücksichtigen kann. Um die Effekte von der Abschnittsstruktur zu untersuchen haben die Forscher einen strukturierten, sowie einen unstrukturierten Ansatz zum Finetuning vom SciBERT-Encoder vorgestellt, sodass diese dann miteinander beim Retrieval verglichen werden konnten. Der unstrukturierte Ansatz bekommt als Eingabe den gesamten Abstract. Beim

strukturierten Ansatz wird der Abstract in vier unterschiedliche Textabschnitte unterteilt. Als Eingabe bekommt der Ansatz dann die einzelnen Textabschnitte anstatt den gesamten Abstract. Durch eine Studie haben sie dann herausgefunden, dass auf einzelne Textabschnitte abgestimmte Modelle in der Lage sind, besser die Wortkontexte zu erfassen, als die Modelle, die die Struktur vom Abstract vernachlässigen. Ein Nachteil an diesem Ansatz ist, dass er sich nur auf die Zerlegung von Abstracts von den Dokumenten beschränkt. Ein Finetuning-Ansatz für BERT sollte in der Lage sein, Dokumente zu klassifizieren, die über BERTs maximale Tokenlänge hinausgehen.
Deswegen wurde ein weiterer Finetuning-Ansatz für BERT von Dai et al. entwickelt, um kontextbewusste Termgewichtungen aus Text zu erzeugen[31]. DeepCT ermittelt dabei die kontextabhängigen Termgewichtungen von kleineren Passagen, indem die von BERT generierten Worteinbettungen in Termhäufigkeiten umgewandelt werden. Die erhaltenen Termhäufigkeiten können dann mithilfe eines invertierten Index gespeichert werden. Retrieval-Modelle können dann den erzeugten Index verwenden, um Ad-Hoc-Suchen auszuführen. Ein Nachteil von DeepCT ist, dass nicht längere Textabschnitte von BERT erfasst werden können. Deshalb haben die Autoren von DeepCT eine Erweiterung von DeepCT veröffentlicht[32]. Die Textabschnitte werden dabei aufgeteilt, sodass die maximale Tokenlänge vom Encoder nicht überschritten wird. Die geteilten Textabschnitte werden dann in kontextualisierte Worteinbettungen umgewandelt und ihre Rankings aggregiert, sodass es möglich wird einen längeren Text mithilfe von DeepCT zu erfassen.
Das Erstellen von kontextualisierten Dokumentrepräsentationen mit DeepCT ähnelt unserem Verwendungszweck für Total-Recall-Suchen. Deswegen benutzen wir die bereits bestehenden Strategien [32] zur Ermittlung der Relevanzlabel für den BERT-Encoder und wenden den auf unser Anwendungsbeispiel an. Dabei entwickeln wir unterschiedliche Strategien, mit denen die Terme im Kontext unterschiedlich gewichtet werden können, im nächsten Kapitel.

Kapitel 3

Kontextabhängige Termgewichtung bei Total-Recall-Suchen

In diesem Kapitel zeigen wir, wie der Screening-Prozess bei systematischen Reviews mithilfe von kontextabhängigen Termgewichtungen verbessert werden kann. Für die Durchführung des Screenings benutzen wir ein bereits bestehendes Total-Recall-System [3]. Dieses System benutzt ein logistisches Regressionsmodell, welches auf dem expliziten Relevanzfeedback der Nutzer trainiert wird. Das HiCal-System gewichtet das logistische Regressionsmodell dabei mithilfe von TF-IDF-Vektoren der gekennzeichneten Dokumente. Wir versuchen die TF-IDF-Vektoren mit der Bedeutung der Terme im Kontext einer Passage erweitern. Deshalb benutzen wir ein neuronales Sprachmodell, um die Dokumente vor dem Screening-Prozess mit HiCal neu zu gewichten [22]. BERT generiert aus den Passagen der Dokumente kontextualisierte Worteinbettungen, die mithilfe eines Finetuning-Ansatzes[31] in einem invertierten Index gespeichert werden. Wir verwenden die erzeugten kontextabhängigen Termgewichtungen, um den Screening-Prozess mit HiCal durchzuführen. Im ersten Abschnitt zeigen wir, wie das Screening mithilfe von HiCal durchgeführt werden kann. Dann beschreiben wir, wie HiCal mithilfe von kontextabhängiger Termgewichtung weiter verbessert werden kann. Dabei benutzen wir unterschiedliche Strategien, um das Finetuning für die kontextualisierte Termgewichtungen mit DeepCT durchzuführen. Bei den unterschiedlichen Strategien werden die Titel/Referenzen/Zitate von den Dokumenten verwendet, um die Terme in den Dokumenten kontextuell zu gewichten. Wir beschreiben in diesem Kapitel auch die Anwendung von kontextabhängigen Termgewichtungen bei dem Screening des systematischen Reviews.

3.1 Dokumentscreening mithilfe von HiCal

Das Ziel vom Screening ist es, alle relevanten und irrelevanten Dokumente im Korpus zu identifizieren. Mithilfe von Total-Recall-Suchen kann dieser Prozess effizienter durchgeführt werden. Bei einer Total-Recall-Suche wird versucht alle relevanten Dokumente mithilfe einer Anfrage zu erhalten. Wir verwenden einen State-of-the-Art Algorithmus bei Total-Recall-Suchen, um das Screening durchzuführen. Das HiCal-System [3] besteht aus zwei Komponenten. Diese sind das kontinuierliche aktive Lernmodell und das Suchmodell.
HiCal benutzt Indri [33] als Suchkomponente, sodass Forscher Suchanfragen formulieren können. Die gefundenen Dokumente können dann manuell überprüft werden.
Das Lernmodell besteht aus einem logistischen Regressionsmodell. Dieses basiert auf der Implementierung von der Baseline Model Implementation(BMI) von dem TREC-Total-Recall-Track von 2015 und 2016 [5, 34]. Diese extrahiert TF-IDF basierte Feature-Vektoren aller Dokumente, die im Korpus enthalten sind. Die Feature-Vektoren werden dann verwendet, um vorherzusagen, welche Dokumente relevant zum Themengebiet sind. Für die Extraktion der Feature-Vektoren werden spezifische Textabschnitte verwendet, wie der Abstract oder die Zusammenfassung.
Die Baseline Model Implementation, auf der HiCal aufbaut, wird mithilfe des iterativen Relevanzfeedbacks der Nutzer trainiert. Dabei bewerten die Nutzer die Relevanz der gezeigten Dokumente zum Themengebiet. Je mehr Nutzerfeedback das Modell bekommt, desto besser kann es die Relevanz von Dokumenten vorhersagen.
Eine Total-Recall-Suche mit BMI läuft wie folgt ab: Als Initialisierung von BMI werden 100 zufällige, nicht gekennzeichnete Dokumente aus dem Datensatz entnommen und als nicht relevant bewertet. Die Anfrage der Total-Recall-Suche wird als relevantes Dokument bewertet. Mit den von HiCal bewerteten Dokumenten und den Dokumenten, die zusätzlich vom Nutzer gelabelt wurden, wird dann der das logistische Regressionsmodell trainiert. Dann werden die ungekennzeichneten Dokumente klassifiziert, nach ihrem Score sortiert und in iterativen Batches an den Nutzer zurückgegeben. Der Nutzer kann dann die Relevanz der ungekennzeichneten Dokumente bewerten und so weiter das Modell verbessern.
Die Autoren von Hical haben dann die Baseline Model Implementation(BMI) noch zusätzlich modifiziert: Wenn der Nutzer die ungekennzeichneten Dokumente bewertet, tut er dies, indem er die Relevanz der ihm vorgelegten Dokumentabschnitte bewertet. Diese Dokumente enthalten oft viel Text, weswegen das Dokumentscreening meist viel Zeit in Anspruch nehmen kann. Deswegen haben die Autoren von HiCal eine Methode entwickelt, bei der nur spezifische

Textabschnitte den Nutzern zur Überprüfung zur Verfügung gestellt wird. Die Nutzer überprüfen die Relevanz eines Dokumentes dann ausschließlich anhand des spezifischen Textabschnittes. Eine Nutzerstudie hat herausgefunden, dass diese Methode die Effizienz des systematischen Reviews steigern kann, sowie die Leistung von der logistischen Regression verbessert[35].
Um zusätzlich die Effizienz des systematischen Reviews zu verbessern, haben die Autoren von HiCal unterschiedliche Ansätze zur erneuten Gewichtung des Modells präsentiert. Wenn nach jedem Nutzerfeedback das Modell neu berechnet wird, so wird es für den Nutzer sehr ineffizient, da bei großen Datensätzen die Neugewichtung des logistischen Regressionsmodells sehr zeitintensiv ist. Deswegen wurden von den Autoren Heuristiken entwickelt, mit denen die Anzahl der Relevanzlabel die nötig sind, um das Modell neu zu gewichten, dynamisch berechnet werden kann. Die Anzahl an Relevanzlabeln, die nötig sind, um das Modell neu zu gewichten, wird Batch-Größe genannt. HiCal benutzt unterschiedliche Methoden, um die Batch-Größe zu bestimmen. Diese sind z. B. dynamische oder statische Funktionen, die dabei helfen, die Batch-Größe anzupassen. HiCal benutzt mehrere Methoden auf dem gleichen Korpus, um die Effizienz vom Screening bei systematischen Reviews verbessern.

3.2 Gewichtung von Termen in Ihrem Kontext mit BERT und DeepCT

HiCal lernt ein logistisches Regressionsmodell basierend auf den Dokumentenrepräsentationen der gekennzeichneten Dokumente. Die TF-IDF-Vektoren werden dabei aus Dokumentfeldern berechnet. Sie werden verwendet, um den Klassifizierer zu trainieren und um das Modell immer neu zu gewichten. Mithilfe von BERT und DeepCT wollen wir das bereits bestehende TF-IDF-Maß verbessern. Die TF-IDF-Vektoren basieren auf der Termhäufigkeit eines Wortes. Ein Nachteil vom TF-IDF-Maß ist, dass nur die Häufigkeit von einem Term betrachtet wird und die Bedeutung des Wortes im Kontext der Passage vernachlässigt wird. Für jedes Wort sollte die Bedeutung und der Zusammenhang mit dem Kontext ermittelt werden, um auch die Terme zu identifizieren, die vom TF-IDF-Maß vernachlässigt werden. Mithilfe der Entwicklungen innerhalb von Machine Learning im Natural Language Processing versuchen wir die Terme in den Dokumenten neu zu gewichten, um so die Durchführungszeit des Screening-Prozesses bei systematischen Reviews zu beschleunigen. Dabei wollen wir die Termhäufigkeiten für alle Terme in den Dokumenten entsprechend ihrer Bedeutung im Kontext gewichten. Das Ziel davon ist es, einen niedrigeren Aufwand bei Total-Recall zu erhalten, als mit den unveränderten

Dokumenten. Für die kontextabhängige Termgewichtung benutzen wir eine modifizierte Version von DeepCT, welche auf dem BERT-Sprachmodell aufbaut. Das BERT-Sprachmodell ist ein kontextualisiertes Sprachmodell. Mit BERT kann man erfassen, wie sich ein Term zu anderen Termen im Kontext verhält. Als Eingabe bekommt BERT eine Passage. Das Sprachmodell tokenisiert dann die Passage und generiert kontextualisierte Worteinbettungen, die anhand der Aufmerksamkeit zu anderen Termen im Kontext erstellt wird. Diese Worteinbettungen sind in der Lage, die syntaktischen/semantischen Eigenschaften eines Terms darzustellen, sodass daraus die Bedeutung des Terms im Kontext zur Passage berechnet werden kann[36]. DeepCT verwendet die kontextualisierten Worteinbettungen, und berechnet für jeden Term ein Termgewicht aus der Worteinbettung. Die Termgewichte werden dann in Termhäufigkeiten transferiert und können dann in einem invertierten Index gespeichert werden. Wir verwenden die erzeugten Termhäufigkeiten, um das Screening für das systematische Review durchzuführen.

3.3 Strategien zum Lernen für kontextabhängige Termgewichtung

Wir verwenden unterschiedliche Strategien, um Finetuning am BERT-Encoder mit DeepCT zu betreiben. Dabei ist es das Ziel für jedes Dokument die Termhäufigkeit der semantisch wichtigen Terme zu erhöhen und die Termhäufigkeit der semantisch unwichtigen Terme zu reduzieren. Wir haben drei Strategien entwickelt, mit denen die Bedeutung der Dokumentterme im Kontext der Passage bestimmt werden können.

3.3.1 Titel-Strategie

Ein gutes Suchsystem sollte in der Lage sein, eine gute Suchmaschine nur aus der Dokumentsammlung zu erstellen [32]. Deswegen wird die Bedeutung der Wörter im Kontext bei der Titel-Strategie nur aus den zur Verfügung stehenden Dokumenten berechnet. Dabei benutzen wir die interne Dokumentstruktur der Dokumente, um die Bedeutung der Terme im Kontext zu bestimmen. Jedes Dokument besteht jeweils aus einem Abstract und einem Titel. Wir nehmen an, dass der Titel viele qualitativ hochwertige Terme besitzt, die das gesamte Dokument beschreiben. Das Ziel der Strategie ist es, den Titel zu benutzen, um die zentralen Terme vom Abstract zu bestimmen. Dabei wird von jedem Term im Abstract überprüft, ob dieser auch im Titel vorkommt. Ist der Term sowohl im Titel, als auch im Abstract enthalten, so bekommt der Term das Relevanzlabel „1“. Tut er dies nicht, so bekommt der Term das Label „0“. Mithilfe

dieser Relevanzlabel sind wir in der Lage, das Finetuning des BERT-Encoders mit DeepCT durchzuführen.

3.3.2 Zitate-Strategie

Die zitierenden Dokumente eines wissenschaftlichen Artikels können Informationen über diesen enthalten. Bei unserer Strategie wollen wir versuchen, diese Informationen aus den zitierenden Dokumenten zu verwenden, um die Bedeutung der Wörter im Kontext der Passage zu ermitteln. Wir verwenden die Zitattitel, um die Bedeutung der Worte im Kontext der Passage zu ermitteln. In der zweiten Strategie werden deshalb die Relevanzlabel, sowohl aus der internen Dokumentstruktur, als auch von verwandten Dokumenten gewonnen. Die Titel der zitierenden Dokumente werden verwendet, um die zentralen Terme im Abstract zu bestimmen. Dabei überprüfen wir von jedem Term im Abstract, ob dieser auch in den Titeln der zitierenden Dokumente enthalten ist. Wenn der Term in einem Titel vorkommt, so bekommt er ein Relevanzlabel. Dabei wird das Relevanzlabel aus der Auftrittshäufigkeit des Terms in den Termen der unterschiedlichen Titel berechnet. Wenn der Term im Abstract in keinem Zitattitel enthalten ist, so bekommt er das Relevanzlabel „0“. Ist der Term im Abstract, sowie in mindestens einem Zitattitel enthalten, so bekommt er ein Relevanzlabel von „0“ bis „1“. Dabei entsprechen die Relevanzlabel der prozentualen Auftrittswahrscheinlichkeit in den unterschiedlichen Titeln. Kommt ein Term im Abstract, sowie in allen Zitattiteln vor, dann erhält er das Relevanzlabel "1". Mithilfe der Relevanzlabel können wir dann das Finetuning des BERT-Encoders mit DeepCT durchführen.

3.3.3 Referenzen-Strategie

Die referenzierenden Dokumente behandeln meist das gleiche spezifische Themengebiet, wie die eigentlichen Dokumente. Dadurch nehmen wir an, dass diese Titel verwandte „Anfragen“ des Dokumentes sind. Deswegen werden in der Referenzen-Strategie die Relevanzlabel, sowohl aus der internen Dokumentstruktur, als auch von verwandten Dokumenten gewonnen. Wir benutzen die Referenztitel, um zentrale Terme aus dem Abstract zu extrahieren. Dabei überprüfen wir von jedem Term im Abstract, ob dieser auch in den Referenztiteln enthalten ist, wie wir es auch bei der Zitate-Strategie getan haben. Wenn der Term in einem Titel vorkommt, so bekommt er ein Relevanzlabel. Dabei wird das Relevanzlabel aus der Auftrittshäufigkeit des Terms des Abstracts in den Termen der unterschiedlichen Referenztitel berechnet. Die Relevanzlabel reichen dann von „0“ bis „1“. Mithilfe der Relevanzlabel können wir dann das Finetuning des BERT-Encoders mit DeepCT durchführen.

3.4 Datensatz zum Lernen für Kontextabhängige Termgewichtung

Für das Finetuning mit DeepCT wurden zwei Trainingsdatensätze verwendet. Ein Trainingsdatensatz kommt vom Genome-TREC-Track[37] aus dem Jahr 2019. Dieser Datensatz enthält über 5000 Dokumente über das Thema Genom Editierung. Das Informationsbedürfnis von dem Genome Editing-Datensatz ist hier, ob Vorteile existieren für Genom-Editierung in Pflanzen und Nutzpflanzen. Für dieses Informationsbedürfnis sind 27 % relevante Dokumente im Datensatz. Zum Training von DeepCT benutzen wir dabei die Dokumente, die uns im Volltext aus dem Genome Editing-Datensatz vorliegen. Insgesamt sind dies 2000 Dokumente, von denen wir den Volltext besitzen und die wir verwenden, um weitere Trainingsdaten zu bekommen. Die Dokumente im Volltext sind dabei eine Teilmenge des gesamten Genome Editing-Datensatzes. Wir verwenden nur die Dokumente im Volltext, da es erheblich an Aufwand spart, um für diese Dokumentrepräsentationen an die Titel der Zitate/Referenzen zu gelangen. Dabei untersuchen wir jedes Dokument vom Volltext-Datensatz und speichern uns den Titel, Abstract und die Referenzen-/Zitattitel. Wir haben zusätzlich den Datensatz von Genome Editing erweitert, um so unsere Strategien auf mehr Daten zu trainieren, sodass unsere Strategien mit DeepCT bessere Vorhersagen bei Dokumenten über Genom-Editierung treffen können. Dabei benutzen wir den bereits bestehenden Datensatz und erweitern diesen mit den Dokumenten aus den Zitaten/Referenzen. Wir verwenden die SemanticScholar-API, um an die Titel der Zitate/Referenzen zu gelangen. SemanticScholar ist eine Suchmaschine, mit der es möglich ist Informationen über wissenschaftliche Artikel zu erlangen.
Der Genome-Datensatz, der zum Trainieren der unterschiedlichen Strategien benutzt wird, besteht aus 214,712 Dokumenten. Jedes dieser Dokumente besteht dabei aus einem Abstract, Titel des Dokumentes, Titel der Referenzen und aus den Titeln der Zitate. Der zweite Datensatz ist auch vom Julius-Kühn-Institut (JKI) und heißt Ceeder'18 und Ceeder'19. Der JKI-Datensatz enthält Dokumente über ökologische und umweltbezogene Themen. Das Informationsbedürfnis in dem Datensatz behandelt die Auswirkungen von Umweltprüfungen auf die Umweltrichtlinien und -praktiken. Der JKI-Datensatz, der zum Trainieren der unterschiedlichen Strategien und Modelle genutzt wird, besteht aus 8000 Dokumenten, mit jeweils einem Titel, Abstract und Referenzen-/Zitattitel.

3.5 Korrelation von Termgewichtungsmodellen

Wir haben die Korrelation von der Gewichtung der zentralen Terme im Abstract bei unterschiedlichen Strategien untersucht. Für die Berechnung der Korrelation werden die beiden Trainingsdatensätze verwendet. Die Korrelation wird dabei aus den Relevanzlabeln der Terme berechnet, die von den Strategien als zentrale Terme im Abstract gekennzeichnet werden. Dabei wird jeder Term aus dem Abstract auf seine Termgewichtungen in den drei unterschiedlichen Strategien untersucht. Wir verwenden zum Berechnen der Korrelation die Spearman-Korrelation. Wir untersuchen dabei die Korrelation von einzelnen Dokumenten. Dabei werden die unterschiedlich gewichteten zentralen Terme im Abstract von jedem Dokument miteinander verglichen. Die gesamte Korrelation von allen Dokumenten berechnen wir dann als Durchschnitt von den Korrelationen der einzelnen Dokumente.
Wie zu sehen in Abbildung 3.1 ist die Korrelation der untersuchten Terme in den unterschiedlichen Strategien niedrig beim Genome Editing-Trainingsdatensatz. Nur bei den Strategien Referenzen und Zitate steigt die Korrelation an. Dies könnte daran liegen, dass Dokumente, die oft zitiert werden, auch von anderen Dokumenten referenziert werden können, vor allem wenn sie das gleiche Themengebiet besitzen und noch wenig Forschung auf dem spezifischen Themengebiet betrieben wurde. Daraus können wir folgen, dass unsere drei Strategien die gleichen Terme unterschiedlich stark gewichten.
Beim Ceeder-Datensatz werden die Terme auch unterschiedlich stark gewichtet. Die Korrelation von den Ceeder-Dokumenten ist jedoch höher als die Korrelation der Relevanzlabel vom Genome Editing-Daten. Zusätzlich überschneiden sich die Referenzen und Zitate von den Ceeder-Daten so sehr, dass genau die gleichen Terme die gleichen Relevanzlabel bekommen. Auch bei den Strategien mit Stemming und Stoppwortlisten bekommen die Terme im Abstract von der Referenzen-Strategie und von der Zitate-Strategie die gleichen Relevanzlabel. s

3.6 Anwendung von kontextabhängiger Termgewichtung bei HiCal

Nachdem die Strategien auf dem Datensatz trainiert wurden, können wir die Modelle benutzen, um unsere Dokumentrepräsentationen zu gewichten. Dabei benutzen wir die trainierten DeepCT-Modelle, um eine kontextabhängige Termgewichtung zu erhalten. Wir wenden unsere Strategien dabei auf jedes Dokumentfeld einzeln an, sodass das komplette Dokument kontextabhängig gewichtet wird.

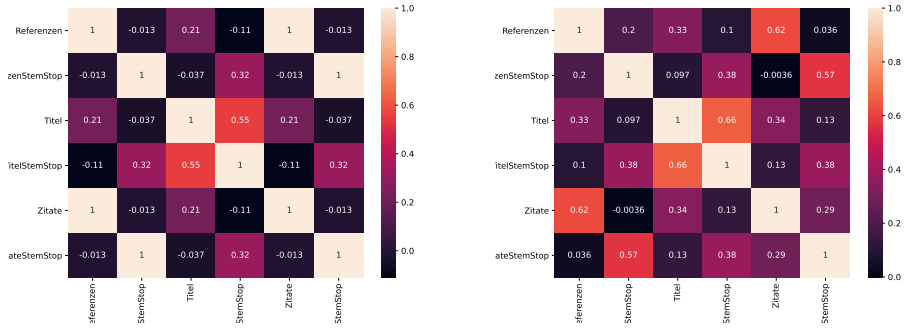

Abbildung 3.1: Spearman-Korrelation der Terme beim Trainingsdatensatz Ceeder (links) und Genome Editing (rechts)

Wenn wir DeepCT auf einen Abstract eines Dokumentes anwenden, kann es dazu führen, dass nicht der gesamte Abstract von BERT erfasst wird. Dies liegt daran, dass BERT eine maximale Passagenlänge von 512 Tokens besitzt. Um dieses Problem zu lösen haben wir Passagen die länger als 512 Tokens sind aufgesplittet, um so zu gewährleisten, dass die gesamte Passage von BERT erfasst werden kann.
Für jede Strategie wird eine eigene Dokumentrepräsentation erzeugt. Die Dokumente werden nach der Gewichtung von DeepCT an HiCal übergeben. Auch die originale Anfrage wird kontextabhängig gewichtet, um ein vollständig kontextabhängiges systematisches Review durchführen zu können. BERT erzeugt für jedes einzelne Dokumentfeld kontextualisierte Worteinbettungen. Diese Worteinbettungen werden von DeepCT verwendet und in eine Bag-of-Words Repräsentation gebracht. Diese Repräsentation können wir dann verwenden, um sie an HiCal zu übergeben. HiCal gewichtet dann das eigene Modell mit den kontextabhängigen Dokumentrepräsentationen und führt das systematische Review mit der modifizierten Anfrage durch. Es gibt noch viele weitere Methoden, die wir verwenden können, um den Aufwand für das Screening des systematischen Reviews zu reduzieren.

Kapitel 4

Evaluation

In diesem Kapitel wird der Einfluss von Dokumentrepräsentationen für den Screening-Prozess in systematischen Reviews mit HiCal evaluiert. Dafür werden drei Dokumentrepräsentationen, die durch unterschiedliche Kombinationen von Dokumentfeldern (Titel, Abstract, Volltext) erzeugt werden, miteinander verglichen. Wir evaluieren auch die mit unseren Strategien erzeugten kontextabhängigen Dokumentrepräsentationen. Dabei evaluieren wir die Auswirkungen, den unsere dediziert trainierten DeepCT-Modelle auf unterschiedliche Dokumentrepräsentationen beim Screening von systematischen Reviews besitzen, anhand von Experimenten auf drei systematischen Reviews aus dem Bereich Umwelt und Genom Editierung bei Pflanzen.

4.1 Experiment Aufbau

Zur Durchführung der Evaluation benutzen wir Dokumente und Relevanz-Annotationen aus drei echten systematischen Reviews, die uns freundlicherweise vom Julius-Kühn-Institut (JKI) zur Verfügung gestellt worden. Jeder der Datensätze umfasst ein vollständiges systematisches Review. Das systematische Review unter dem Namen Genom Editierung befasst sich mit Genom-Editierung bei Pflanzen. Das Informationsbedürfnis von diesem Datensatz bestimmt, welche Eigenschaften bei Genom-Editierung von Modell-/Nutzpflanzen für die landwirtschaftliche Produktion verändert werden. Die systematischen Reviews mit dem Namen Ceeder'18 bzw. Ceeder'19 befassen sich mit ökologischen und umweltbezogenen Themen. Das Informationsbedürfnis in den Ceeder-Datensätzen behandelt dabei die Auswirkungen von Umweltprüfungen auf die Umweltrichtlinien und -praktiken.
Wir verwenden die JKI-Datensätze zum Trainieren unserer DeepCT-Modelle, sowie um die Effektivität des Screening-Prozesses mit HiCal zu beurteilen.
Wir benutzen als Ausgangspunkt zum Training der Strategien für die Genom-

Modelle den Genom Editierung-Datensatz von 2019. Davon benutzen wir, wie im vorherigen Kapitel beschrieben, einen Teil der Dokumente, die uns im Volltext zur Verfügung gestellt wurden, als Ausgangspunkt zum Erstellen eines großen Trainingsdatensatzes mithilfe der Zitate/Referenzen/Titel von jedem Dokument. Der Trainingsdatensatz enthält dabei 214712 Dokumente über Genom-Editierung.
Die Ceeder-Modelle werden auf allen Dokumenten trainiert, die im Ceeder'18 und Ceeder'19 enthalten sind, weil sie auf Basis des initialen Retrievals potenziell relevant zu dem Informationsbedürfnis von Ceeder'18/19 sind. Der Trainingsdatensatz von den Ceeder-Modellen enthält 47913 Dokumente über Umweltrichtlinien und -praktiken. Wir erweitern auch diesen Datensatz mit den Zitattiteln/Referenztiteln/Titeln der Dokumente, um die dedizierten Ceeder-Modelle zu trainieren.
Wir evaluieren die trainierten Modelle auf den JKI-Datensätzen. Jedes Dokument aus den Testdatensätzen besteht aus den Feldern Abstract und Titel. Die Terme in den Dokumentfeldern werden mit den dedizierten DeepCT-Modelle kontextabhängig gewichtet. Wir verwenden die erzeugten kontextabhängigen Dokumentrepräsentationen, um die Effektivität unterschiedlicher Dokumentrepräsentationen im Screening-Prozess mit HiCal zu untersuchen.
HiCal benutzt ein logistisches Regressionsmodell, welches mit dem expliziten Relevanzfeedback der Nutzer trainiert wird. Wir benutzen das Relevanzfeedback der drei JKI-Datensätzen, um die Effektivität des Screening-Prozesses zu untersuchen. Mithilfe des Relevanzfeedbacks und den erzeugten Dokumentrepräsentationen sind wir in der Lage, den Screening-Prozess bei systematischen Reviews mit HiCal zu untersuchen.
Als Baseline für unsere Ansätze haben wir HiCal mit der ungewichteten Repräsentation verglichen. Dabei übergeben wir die ungewichtete Dokumentrepräsentation (Titel, Abstract) an HiCal. Mit der erzeugten Baseline können wir dann vergleichen, ob unsere Ansätze dabei helfen einen niedrigeren Aufwand bei hohen Recall-Werten zu erreichen, als die kontextabhängigen Dokumentrepräsentationen. Wir benutzen die von HiCal erzeugten Rankings, um Plots zu erstellen, die den benötigten Aufwand für einen spezifischen Recall darstellen. Zusätzlich zu den Plots untersuchen wir das Work-Saved-over-Sampling-Maß (WSS), um die Effektivität des Screening-Prozesses bei unterschiedlichen Recall-Werten zu evaluieren. Beim WSS-Maß wird gezeigt, wie viel Arbeit mit dem technologischen systematischen Review gespart werden kann. Wir verwenden die Definition von Cohen et al. [38], um den WSS für unsere Screenings zu berechnen:

$$WSS@Recall = \frac{TrueNegatives+FalseNegatives}{Gesamte\ Anzahl\ an\ Dokumenten} - (1 - Recall)$$

Die Summe der True Negatives und der False Negatives sind die Dokumente, die noch nicht vom Nutzer gekennzeichnet wurden. Wir berechnen die Summe der Negatives aus Anzahl der Dokumente minus der Anzahl an Dokumenten, die bereits vom Nutzer gekennzeichnet wurden. Der Aufwand gibt bei uns an, wie viele Dokumente im Laufe des Screenings betrachtet wurden. Je höher der WSS ist, desto weniger Dokumente mussten beim Screening-Prozess vom Nutzer gekennzeichnet werden, um einen bestimmten Recall zu erreichen. Wir haben den WSS für unterschiedliche Recall-Werte bei jedem Screening zusätzlich zu den Plots berechnet. Dabei haben wir vorrangig hohe Recall-Werte untersucht, da wir mit unseren Termgewichtungen versuchten in genau diesem Bereich eine Verbesserung zu erzielen. Auch versuchen wir den Aufwand bei Total-Recall zu verringern, um die Effektivität des Screenings mit HiCal zu erhöhen.Wir vergleichen den WSS von den Strategien mit der ungewichteten Dokumentrepräsentation des Dokumentes. Diese Repräsentation wird von uns als Baseline verwendet, mit der wir den unsere Strategien vergleichen.

4.2 Vergleich von Dokumentfeldern beim Screening

Forscher, die ein systematisches Review durchführen möchten, stehen vor der Frage, in welcher textuellen Repräsentation die zu screenenden Dokumente in das systematische Review eingebracht werden sollen. In der Regel besteht die Möglichkeit, Dokumente als Volltext (entsprechende Lizenz für Volltext vorausgesetzt), als Abstract, als Titel, oder aus Kombination daraus für den Screening-Prozess zu verwenden. Wir haben den Effekt von drei unterschiedlichen Dokumentfeldern bei Total-Recall-Suchen untersucht. Dabei haben wir die Klassifizierung von Dokumenten mit dem Titel, Abstract und dem Volltext verglichen. Das Ziel war es herauszufinden, welche Dokumentrepräsentation den geringsten Aufwand bei hohen Recall-Werten erzeugt und damit als Baseline dient, um das Screening für das systematische Review durchzuführen. Dabei haben wir einen Testdatensatz von 1200 Dokumenten verwendet, da wir nur von dem Datensatz den Volltext, sowie den Abstract und den Titel besitzen. Der Testdatensatz ist Teil des Genom Editierung-Datensatzes und wurde auch als Ausgangspunkt für das Scraping der Trainingsdaten der Strategien verwendet. In dem Datensatz besitzen wir für jedes Dokument den Volltext/Abstract und den Titel, welche wir austesten, um das Dokument bei Total-Recall-Suchen zu klassifizieren. Die Dokumente in dem Datensatz sind wissenschaftliche Artikel zum Thema Genom-Editierung.
Das erste Dokumentfeld ist der Titel. Der Titel besteht aus wenigen Termen, von denen wir annehmen, dass sie ein hohes Informationsgehalt besitzen.

Durchschnittlich besitzt der Titel im Testdatensatz 10 Terme. Das zweite Dokumentfeld ist der Abstract. Dieser enthält Informationen über das gesamte Dokument, da er von Forschern als eine Art Zusammenfassung ihrer Arbeit verwendet wird [39]. Der Abstract enthält durchschnittlich 250 Terme in unserem Testdatensatz. Das dritte Dokumentfeld ist der Volltext. Dieser enthält alle Informationen über das Dokument. Der Volltext beinhaltet alle Dokumentfelder, die der wissenschaftliche Artikel besitzt. Im Volltext sind sowohl wichtige, als auch unwichtige Terme enthalten. Im Durchschnitt hat der Volltext 4000-8000 Terme, die verwendet werden, um das Dokument zu klassifizieren.
Wir haben dann mit den drei unterschiedlichen Repräsentationen systematische Reviews durchgeführt. Dabei haben wir bei allen Repräsentation HiCal verwendet, sodass wir feststellen konnten, welche Dokumentrepräsentationen wir als Baseline benutzen, um unsere Strategien zu evaluieren.
In der Tabelle 4.1 steht die Auswertung für den WSS von den unterschiedlichen Dokumentfeldern. Es stellt sich heraus, dass der Volltext am effektivsten ist, um mit HiCal das Screening für systematische Reviews durchzuführen. Für das Screening vom Abstract mit Hical wird ein 4 % niedriger WSS bei 95 % Recall als beim Volltext erreicht. Dies bestätigt unsere Vermutung, dass der Abstract viele dokumentbeschreibende Terme enthält. Der Titel erreicht beim Screening einen wesentlich geringeren WSS als der Abstract oder der Volltext.
Wissenschaftliche Artikel im Volltext sind meist in Online-Bibliotheken verfügbar. Die Online-Bibliotheken besitzen eine Lizenz für die vollen wissenschaftlichen Artikel, sodass diese meist hinter einer Paywall versteckt sind. Von den Lizenzinhabern wird trotzdem meist der Abstract und der Titel veröffentlicht, sodass die Leser sich einen Überblick über den Artikel verschaffen können, bevor sie eine Lizenz für den wissenschaftlichen Artikel erwerben. Auch ist es nicht möglich automatische Downloads von Volltexten durchzuführen, da dies gegen die Nutzungsbedingungen von Anbietern (z.B. ACM [40]) verstoßen. Da die Verfügbarkeit zu den Volltexten von wissenschaftlichen Artikel begrenzt ist, können wir nicht für jeden Datensatz die Dokumente im Volltext beim Screening-Prozess verwendet werden.
Mithilfe des Screening-Prozess (Abbildung 4.1) haben wir herausgefunden, dass der Abstract eines Dokumentes meist schon reicht, um diesen vollständig zu klassifizieren. Zusätzlich zum Abstract können wir noch den Titel verwenden, da dieser meist bei fehlenden Abstracts das Dokument noch ausreichend repräsentieren kann. Wir benutzen deshalb den Titel und den Abstract als Baseline für die Anwendung von kontextabhängiger Termgewichtung bei Total-Recall-Suchen.

Tabelle 4.1: WSS von einzelnen Dokumentfeldern (Titel, Abstract, Volltext) beim Screening für systematische Reviews mit HiCal

Repräsentation	Datensatz	WSS@85	WSS@90	WSS@95
Abstract	Genom-Volltext	0.389	0.439	0.489
Volltext	Genom-Volltext	**0.408**	**0.458**	**0.508**
Titel	Genom-Volltext	0.281	0.331	0.381

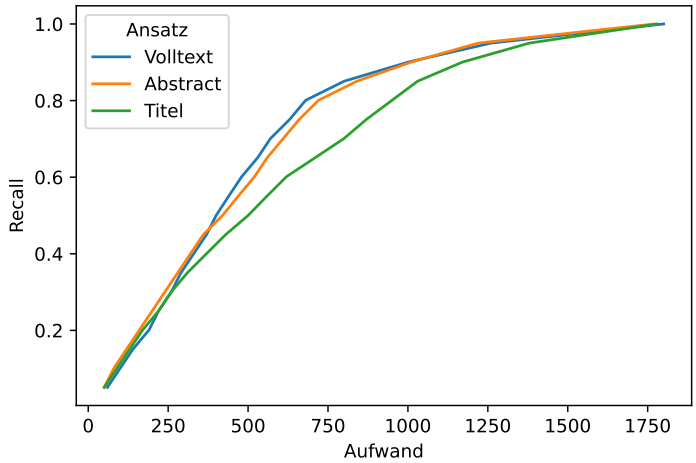

Abbildung 4.1: Aufwand gegen Recall bei Titel, Abstract und Volltext

4.3 Evaluation der Strategien

In den folgenden Abschnitten besprechen wir die Anwendung und die Evaluation von der Titel-/Zitate-/Referenzenstrategie. Dabei untersuchen wir die Auswirkungen von den unterschiedlichen Trainingsstrategien auf den Screeningprozess für das systematische Review, welches wir mithilfe von HiCal durchführen.

4.3.1 Evaluation der Titel-Strategie

Für die Titel-Strategie haben wir zwei Modelle pro Trainingsdatensatz trainiert. Das Modell Titel benutzt alle Terme im Abstract und im Titel zur Gewichtung der kontextabhängigen Terme. Für das Modell Titel + Stem/Stop wird der gleiche Trainingsdatensatz verwendet, jedoch werden Terme, die in der Stoppwortliste enthalten sind, nicht vom Modell gewichtet. Die Terme, welche einen gemeinsamen Wortursprung haben, werden vom Modell Titel + Stem/Stop gleich gewichtet.

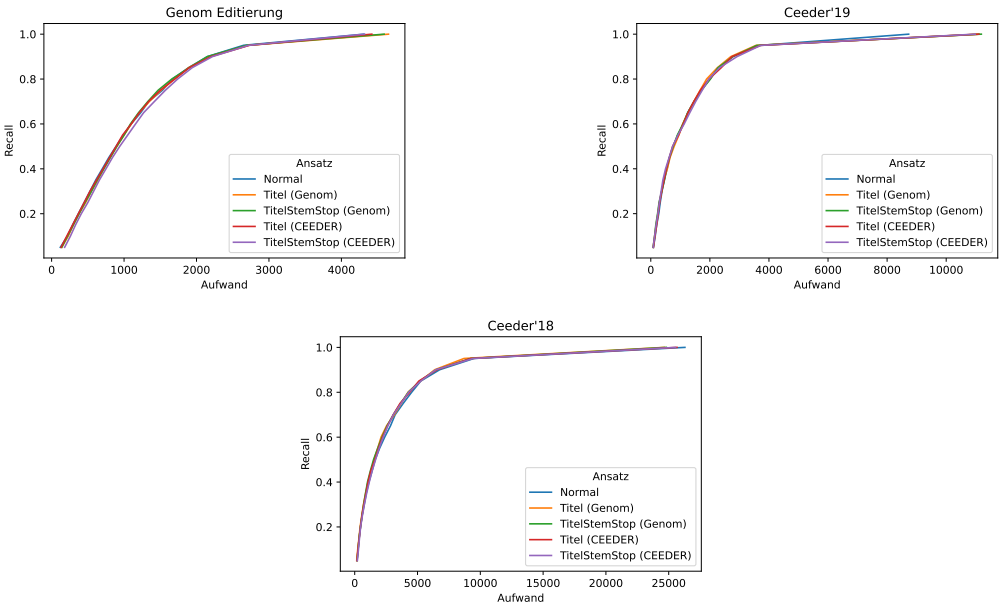

Abbildung 4.2: Plot über den Aufwand gegen den Recall beim Screening-Prozess von systematischen Reviews mit der Titel-Strategie. Als Testdatensätze wurden der Ceeder'18, der Ceeder'19 und der Genom-Editierung-Datensatz verwendet.

Wir haben eine Stoppwortliste verwendet, sodass Terme, die keinen Informationsgehalt haben und häufig auftreten, nicht vom Modell kontextabhängig gewichtet werden. Wir haben einen Stemmer verwendet, um Worte, die einen ähnlichen Wortursprung haben, stärker zu gewichten, sodass verbesserte Rankings von Hical erzeugt werden können.
Pro Trainingsdatensatz wird ein Titel-Modell und ein Titel + Stem/Stop-Modell mithilfe der Titel-Strategie trainiert. Dabei wurden von jedem Dokument aus den Trainingsdatensätzen mithilfe der Titel-Strategie die Relevanzlabel gewonnen. Wir benutzen die Dokumente und deren Relevanzlabel, um den BERT-Encoder mit DeepCT zu trainieren.
Wir testen unsere trainierten Modelle am Screening-Prozesses des systematischen Reviews. Dabei verwenden wir den Ceeder'18, Ceeder'19 und Genome Editing, um unsere erzeugten Dokumentrepräsentationen zu evaluieren.
In der Tabelle 4.2 wird die Auswertung des WSS beim Screening von systematischen Reviews mit den erzeugten Dokumentrepräsentationen der Titel-Strategie vorgenommen. Bei dem Ceeder'18-Testdatensatz erreichen wir mit beiden Trainingsdatensätzen einen höheren WSS als mit der herkömmlichen Methode. Die Titel-Methode, welche auf den Ceeder-Daten trainiert wurde, schafft es beim Ceeder'18, sowie beim Genom Editierung-Datensatz den höchsten WSS zu erreichen.

Tabelle 4.2: Übersicht über den WSS beim Screening-Prozess von systematischen Reviews mit der Titel-Strategie mit Stemming/Stopping. Als Testdatensätze wurden der Ceeder'18, der Ceeder'19 und der Genom-Editierung-Datensatz verwendet.

Methode	Trainingsdaten	Datensatz	WSS@85	WSS@90	WSS@95
Normal	-	Ceeder'18	0.6536	0.7036	0.7536
Titel	Genom	Ceeder'18	0.6544	0.7044	0.7544
+ Stem/Stop	Genom	Ceeder'18	0.6562	0.7062	0.7562
Titel	Ceeder	Ceeder'18	**0.6592**	**0.7092**	**0.7592**
+ Stem/Stop	Ceeder	Ceeder'18	0.6536	0.7036	0.7536
Normal	-	Ceeder'19	0.6491	0.6991	0.7491
Titel	Genom	Ceeder'19	**0.6518**	**0.7018**	**0.7518**
+ Stem/Stop	Genom	Ceeder'19	0.6501	0.7001	0.7501
Titel	Ceeder	Ceeder'19	0.6405	0.6905	0.7405
+ Stem/Stop	Ceeder	Ceeder'19	0.6457	0.6957	0.7457
Normal	-	Genom	0.4599	0.5099	0.5599
Titel	Genom	Genom	**0.4605**	**0.5105**	**0.5605**
+ Stem/Stop	Genom	Genom	0.4586	0.5086	0.5586
Titel	Ceeder	Genom	0.4580	0.5080	0.5580
+ Stem/Stop	Ceeder	Genom	0.4485	0.4985	0.5485

Auf dem Ceeder'19-Datensatz erreichen die Genom-Modelle einen höheren WSS als die Ceeder-Modelle. Das Stemming und Stopping bewirkt beim Screeningprozess von systematischen Reviews meist einen niedrigeren WSS bei hohen Recall-Werten. Die Titel-Strategie schafft es, den Aufwand für das Screening zu verringern.

4.3.2 Evaluation der Zitate-Strategie

Für die Zitate-Stategie haben wir pro Trainingsdatensatz jeweils zwei Modelle trainiert. Für jeden Trainingsdatensatz haben wir jeweils zwei Modelle trainiert. Das Modell Zitate benutzt alle Terme im Abstract zur Gewichtung der kontextabhängigen Terme. Das Modell Zitate + Stem/Stop gewichtet die Terme gleich wie das Modell Zitate, jedoch wird zusätzlich ein Stemmer und eine Stoppwortliste verwendet. Das Modell Zitate + Stem/Stop benutzt eine Stoppwortliste und einen Stemmer. Wir verwenden eine Stoppwortliste, sodass die Termhäufigkeit von häufig auftretenden Termen, die nur einen geringen Informationsgehalt besitzen, verringert wird. Zusätzlich benutzen wir das Stemming bei dem Modell Zitate +Stem/Stop, damit die Termhäufigkeit von Worten, die den gleichen Wortursprung besitzen erhöht wird.

Tabelle 4.3: Übersicht über den WSS beim Screening-Prozess von systematischen Reviews, mit der Zitate-Strategie, mit Stemming/Stopping. Als Testdatensätze wurden der Ceeder'18, der Ceeder'19 und der Genom-Editierung-Datensatz verwendet.

Methode	Trainingsdaten	Datensatz	WSS@85	WSS@90	WSS@95
Normal	-	Ceeder'18	**0.6536**	**0.7036**	**0.7536**
Zitate	Genom	Ceeder'18	0.6453	0.6953	0.7453
+ Stem/Stop	Genom	Ceeder'18	0.6502	0.7002	0.7502
Zitate	Ceeder	Ceeder'18	0.6453	0.6953	0.7453
+ Stem/Stop	Ceeder	Ceeder'18	0.6501	0.7001	0.7501
Normal	-	Ceeder'19	**0.6491**	**0.6991**	**0.7491**
Zitate	Genom	Ceeder'19	0.6421	0.6921	0.7421
+ Stem/Stop	Genom	Ceeder'19	0.6453	0.6953	0.7453
Zitate	Ceeder	Ceeder'19	0.6472	0.6972	0.7472
+ Stem/Stop	Ceeder	Ceeder'19	0.6430	0.6930	0.7430
Normal	-	Genom	**0.4599**	**0.5099**	**0.5599**
Zitate	Genom	Genom	0.4557	0.5057	0.5557
+ Stem/Stop	Genom	Genom	0.4452	0.4952	0.5452
Zitate	Ceeder	Genom	0.4502	0.5002	0.5502
+ Stem/Stop	Ceeder	Genom	0.4433	0.4933	0.5433

Wir trainieren die Modelle auf den zwei Trainingsdatensätzen. Dabei wird von jedem Term im Abstract eines Dokumentes überprüft, ob dieser auch in den Zitattiteln enthalten ist. Wenn ein Term in den Zitattiteln enthalten ist, so bekommt der Term ein Relevanzlabel abhängig von der Auftrittswahrscheinlichkeit des Terms in den unterschiedlichen Titeln der Zitate. Mithilfe der generierten Relevanzlabel sind wir in der Lage, das BERT-Sprachmodell mit DeepCT zu trainieren. Unsere trainierten Modelle evaluieren wir auf den Testdatensätzen, die uns vom Julius-Kühn-Institut zur Verfügung gestellt wurden. Die erzeugten Dokumentrepräsentationen werden dann beim Screening-Prozess mit HiCal evaluiert. Wir überprüfen von jedem Modell den WSS bei hohen Recall-Werten, da wir uns von den kontextabhängigen Termgewichtungen dort eine Verringerung des Aufwands versprechen. Zusätzlich plotten wir den Aufwand gegen den Recall, sodass der Aufwand über die gesamte Recall-Kurve dargestellt werden kann. In der Tabelle 4.3 werden die erzeugten Dokumentrepräsentationen der Zitate-Strategie beim Screening mit dem WSS bei hohen Recall-Werten evaluiert. Wir untersuchen das Screening bei 85 %/90 % und bei 95 % Recall und bestimmen dafür das WSS-Maß. Bei dem Ceeder'18 schafft es kein Modell, die Terme in der Dokumentrepräsentation mit der Bedeutung der Terme im Kontext zu erweitern, sodass der Aufwand im Vergleich zur ungewichteten Dokumentrepräsentation verringert wird. Die Modelle, welche auf unter-

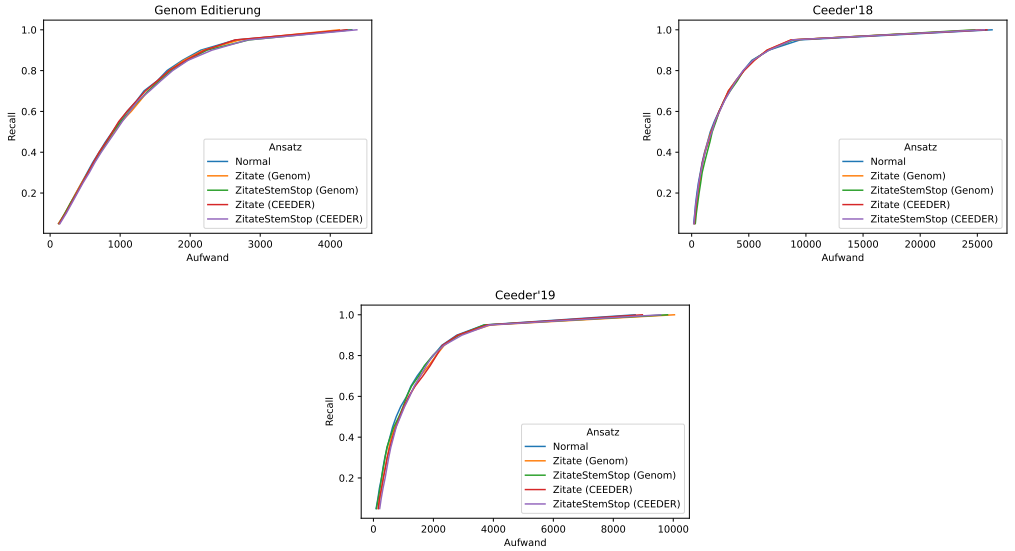

Abbildung 4.3: Plot über den Aufwand gegen den Recall beim Screening-Prozess von systematischen Reviews mit der Zitate-Strategie. Als Testdatensätze wurden der Ceeder'18, der Ceeder'19 und der Genom-Editierung-Datensatz verwendet.

schiedlichen Daten trainiert wurden, erreichen fast den gleichen WSS, bei dem Ceeder'18. Das Stemming und die Stoppwortliste verbessern das WSS-Maß der erzeugten Dokumentrepräsentationen bei hohen Recall-Werten auf diesem Datensatz. Für den Ceeder'19 erreichen wir fast das gleiche Ergebnis wie beim Ceeder'18. Die erzeugten Dokumentrepräsentationen für den Ceeder'19 erhöhen den Aufwand für das Screening, im Vergleich zur ungewichteten Dokumentrepräsentation. Das Stemming und die Stoppwortliste verbessern nur das Modell, welches auf den erweiterten Genom Editierung-Daten trainiert wurde. Bei dem Modell, welches auf Ceeder trainiert wurde, bewirkt das Stem/Stop eine Verschlechterung des WSS-Maßes. Auf den Genome Editing-Daten bewirken die erzeugten Dokumentrepräsentationen eine Verschlechterung des WSS-Maßes bei hohen Recall-Werten. Auch die Stem/Stop-Modelle bewirken einen erhöhten Aufwand im Vergleich zu den Modellen, die keinen Stemmer oder eine Stoppwortliste benutzen. Bei beiden Trainingsdatensätzen der Zitate-Strategie zeigen sich die gleichen Tendenzen bei der Evaluation der drei Testdatensätze. Die Zitate-Strategie erzeugt keine Dokumentrepräsentation für das Screening, welches den Aufwand im Vergleich zur ungewichteten Dokumentrepräsentation reduziert. Stattdessen bewirken die erzeugten Dokumentrepräsentationen einen erhöhten Aufwand für die Durchführung des Screenings.

4.3.3 Evaluation der Referenzen-Strategie

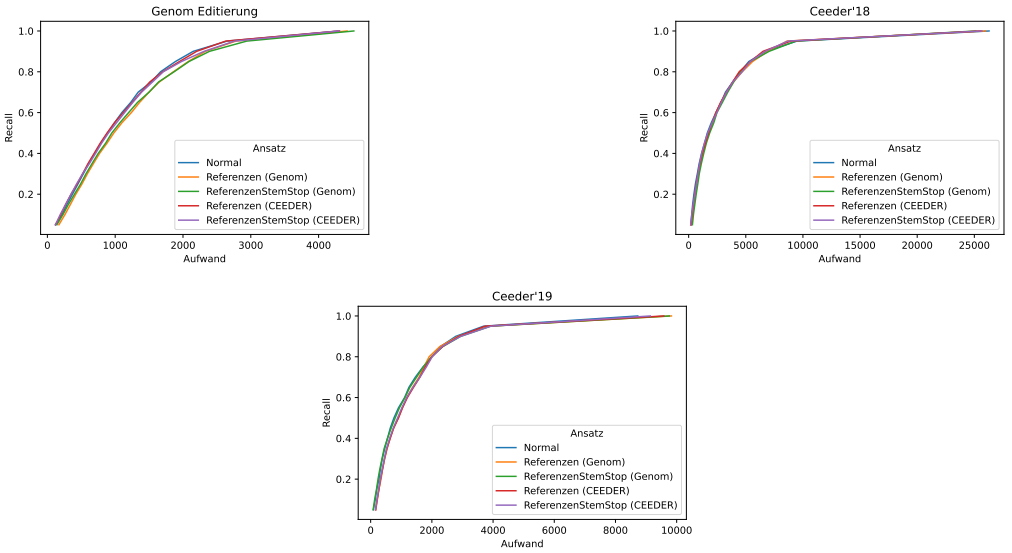

Abbildung 4.4: Plot über den Aufwand gegen den Recall beim Screening-Prozess von systematischen Reviews mit der Referenzen-Strategie. Als Testdatensätze wurden der Ceeder'18, der Ceeder'19 und der Genom-Editierung-Datensatz verwendet.

Um die Referenzen-Strategie zu evaluieren, haben wir zwei Trainingsdatensätze verwendet. Für jeden Trainingsdatensatz haben wir mithilfe der Referenzen und des Abstracts der Dokumente zwei Modelle trainiert. Um die Modelle zu trainieren werden alle Terme im Abstract mit einem Relevanzlabel gekennzeichnet, welches der relativen Auftrittswahrscheinlichkeit in den Titeln der Referenzen entspricht. Das Modell Referenzen benutzt alle Terme im Abstract zum Training des Modells. Das Modell Referenzen + Stem/Stop wird auf den gleichen Dokumenten, wie das Referenzen-Modell trainiert. Zusätzlich wird für das Modell Referenzen + Stem/Stop ein Stemmer und eine Stoppwortliste verwendet. Mithilfe der Stoppwortliste, werden Terme, welche oft im Text auftauchen, aber keinen hohen Informationsgehalt besitzen, nicht für die Erstellung der kontextabhängigen Termgewichtungen verwendet. Das Modell Referenzen + Stem/Stop benutzt auch einen Stemmer, sodass Terme mit gleichem Wortursprung auch die gleiche Relevanzlabel bekommen, für ein effizienteres Retrieval und bessere Rankings.
Wir verwenden die trainierten Modelle, um aus der ungewichteten Dokumentrepräsentation eine kontextabhängige Dokumentrepräsentation zu erzeugen. Die erzeugten Dokumentrepräsentationen werden dann an den Screening-Prozess mit HiCal übergeben. Zum Testen des Screening-Prozesses verwenden wir den Ceeder'18, den Ceeder'19 und den Genom Editierung-Datensatz. Die

Tabelle 4.4: Übersicht über den WSS beim Screening-Prozess von systematischen Reviews mit der Referenzen-Strategie mit Stemming/Stopping. Als Testdatensätze wurden der Ceeder'18, der Ceeder'19 und Genom-Editierung verwendet.

Methode	Trainingsdaten	Datensatz	WSS@85	WSS@90	WSS@95
Normal	-	Ceeder'18	**0.6536**	**0.7036**	**0.7536**
Referenzen	Genom	Ceeder'18	0.6412	0.6912	0.7412
+ Stem/Stop	Genom	Ceeder'18	0.6485	0.6985	0.7485
Referenzen	Ceeder	Ceeder'18	0.6483	0.6983	0.7483
+ Stem/Stop	Ceeder	Ceeder'18	0.6484	0.6984	0.7484
Normal	-	Ceeder'19	**0.6491**	**0.6991**	**0.7491**
Referenzen	Genom	Ceeder'19	0.6509	0.7009	0.7509
+ Stem/Stop	Genom	Ceeder'19	0.6425	0.6925	0.7425
Referenzen	Ceeder	Ceeder'19	0.6429	0.6929	0.7429
+ Stem/Stop	Ceeder	Ceeder'19	0.6452	0.6952	0.7452
Normal	-	Genom	**0.4599**	**0.5099**	**0.5599**
Referenzen	Genom	Genom	0.4206	0.4706	0.5206
+ Stem/Stop	Genom	Genom	0.4175	0.4675	0.5175
Referenzen	Ceeder	Genom	0.4467	0.4967	0.5467
+ Stem/Stop	Ceeder	Genom	0.4407	0.4907	0.5407

Referenzen-Strategie erzeugt den höchsten Aufwand für die systematischen Reviews wie in den Abbildungen 4.4 zu sehen ist. Alle erzeugten Dokumentrepräsentationen, verschlechtern den Screening-Prozess, wie es auch bei der Evaluation der Zitate-Strategie der Fall war. Wie in der Tabelle 4.4 zu sehen ist, verschlechtert sich der WSS bei hohen Recall-Werten für alle erzeugten Dokumentrepräsentationen. Beide Trainingsdatensätze erreichen keine Verringerung des Aufwands für die Screenings mit der Referenzen-Strategie. Das Stemming und Stopping bewirkt auf dem Ceeder'18 einen niedrigeren Aufwand bei hohen Recall-Werten. Beim Ceeder'19 bewirkt das Stemming/Stopping eine Verbesserung bei dem Ceeder-Modell, jedoch eine Verschlechterung des Genom-Modells. Auf dem Genom Editierung-Datensatz bewirkt das Stemming/Stopping eine Verschlechterung des WSS bei allen hohen Recall-Werten. Insgesamt bewirkt die Gewichtung der Terme im Kontext mit der Referenzen-Strategie keine Verbesserung zu der ungewichteten Dokumentrepräsentation, sondern sorgt für einen erhöhten Aufwand bei der Durchführung des Screenings.

4.4 Volltext-Dokumente mit kontextabhängiger Termgewichtung

Wir haben den Effekt von kontextabhängiger Termgewichtungen von Volltexten bei Total-Recall-Suchen evaluiert. Wir haben den geringsten Aufwand bei Total-Recall-Suchen mit dem Volltext erzielt, als wir unterschiedliche Dokumentrepräsentation verglichen haben (siehe Tabelle 4.1). Deswegen haben wir bei der Evaluation unserer Strategien auch Dokumente mit dem Volltext kontextabhängig gewichtet. Wir haben dafür die Modelle verwendet, die auf dem erweiterten Genom-Datensatz trainiert wurden. Wir haben dann mit den gewichteten Volltextdokumenten Screenings mit dem technologiebasierten Unterstützungssystem HiCal durchgeführt. Dabei haben wir zur Evaluation der Screenings den WSS berechnet und den Recall gegen den Aufwand geplottet. Beim Anwenden von DeepCT auf die Volltexte erhofften wir uns eine Verbesserung beim WSS, wie es auch bei der Titel-Strategie beim Genom Editierung-Testdatensatz zu sehen war. Leider konnte keine Strategie eine Verbesserung des WSS beim Screening für das systematische Review bewirken (siehe Tabelle **??**. Die Ursache von der schlechten Performance unserer Strategien hat mehrere Ursachen.
BERT betrachtet nur Passagen aus einem Text und produziert daraus die kontextualisierten Worteinbettungen. Die Dokumente im Volltext sind aber wesentlich länger als die maximale Tokeneingabelänge von 512 Tokens und können deshalb nicht vollständig von BERT erfasst werden. BERT verwendet die ersten 512 Tokens der Eingabe und alle Tokens, die danach folgen werden, von BERT vernachlässigt. Deswegen haben wir den Volltext in 512 Tokens lange Passagen unterteilt, sodass der gesamte Volltext von DeepCT kontextabhängig gewichtet werden kann. Aber auch mithilfe der Unterteilung vom Volltext in 512 Tokens lange Bereiche, kann kein höherer WSS beim Screening bei hohen Recall-Werten, als mit der ungewichteten Dokumentrepräsentation erzielt werden. Ein Vorteil wäre, wenn wir BERT auf den gesamten Volltext direkt anwenden könnten, sodass wir gar keine Trennungen vornehmen müssten.
Eine weitere Ursache ist das Training der Strategien. Wir haben unsere Strategien auf dem Abstract von den Dokumenten in den Trainingsdaten trainiert. Diese Abstracts sind aber wesentlich kürzer als der Volltext eines Dokumentes. Deshalb fällt es DeepCT auch schwerer, alle Aspekte des Volltextes vollständig zu erfassen. Als Verbesserung der Strategien könnten wir zusätzlich unsere Strategien auf den Volltexten der Dokumente trainieren. Dann wären unsere Modelle vielleicht imstande einen besseren WSS bei den Volltexten zu erzielen.

Tabelle 4.5: WSS beim Screening von systematischen Reviews mit Volltexten

Methode	Datensatz	WSS@85	WSS@90	WSS@95
Normal	Volltext	**0.4227**	**0.4727**	**0.5227**
Titel	Volltext	0.4106	0.4606	0.5106
+ Stem/Stop	Volltext	0.4122	0.4622	0.5122
Zitate	Volltext	0.4029	0.4529	0.5029
+ Stem/Stop	Volltext	0.4095	0.4595	0.5095
Referenzen	Volltext	0.3902	0.4402	0.4902
+ Stem/Stop	Volltext	0.3968	0.4468	0.4968

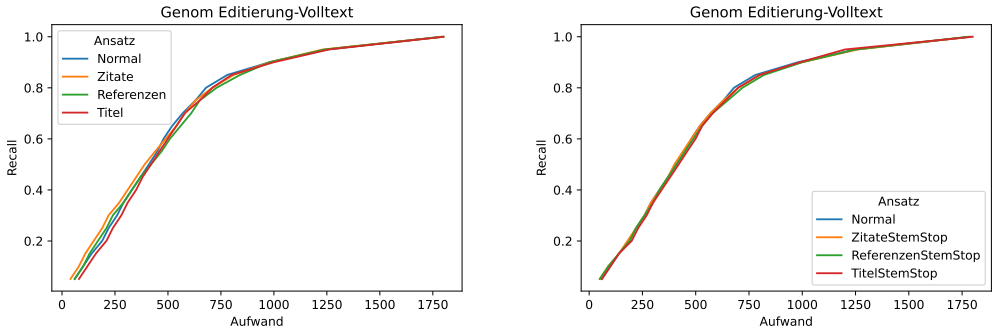

Abbildung 4.5: Aufwand gegen Recall bei Volltexten mit kontextabhängiger Termgewichtung

4.5 Korrelation der Dokumentrepräsentationen mit kontextabhängiger Termgewichtung

Um eine Erklärung für die Rankings von HiCal mit den erzeugten Dokumentrepräsentationen zu finden, haben wir die Korrelation der Rankings von HiCal berechnet. Wir vergleichen die Korrelation der Rankings auf dem Genome Editing Datensatz und überprüfen dabei die Modelle, welche sowohl auf den Ceeder-Datens, als auch auf den erweiterten Genome Editing-Daten trainiert wurden. Wir berechnen die Korrelation von den Rankings, um eventuelle Ähnlichkeiten zwischen diesen festzustellen. Insgesamt besitzen wir sechs termgewichtete Dokumentrepräsentationen pro Trainingsdatensatz, von denen wir die Korrelation untersuchen. Wir untersuchen zusätzlich die ungewichtete Dokumentrepräsentation, bevor sie mit einem Modell neu gewichtet wurde. Für die Rankings von HiCal mit den erzeugten Dokumentrepräsentationen auf den unterschiedlichen Trainingsdaten haben wir in Abbildung 4.6 die Spearman-Korrelation berechnet.

Wie in Abbildung 4.6 zu sehen ist die Korrelation der von Hical erzeug-

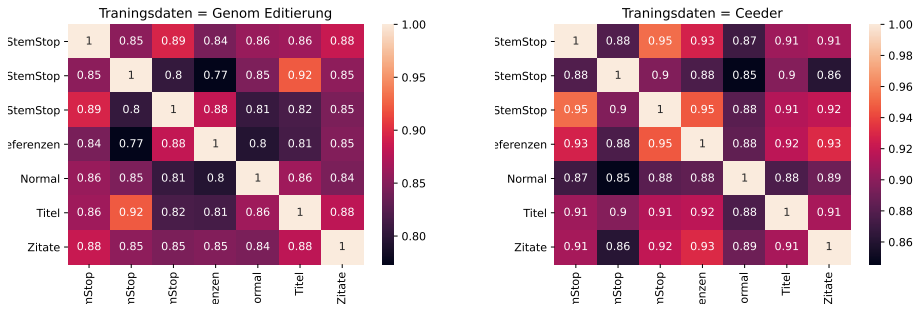

Abbildung 4.6: Korrelation der erzeugten Rankings von den kontextabhängigen Dokumentrepräsentationen der entwickelten Titel/Zitate/Referenzenstrategie. Zusätzlich werden die Modelle mit Stemming/Stopping verglichen.

ten Rankings sehr hoch. Die berechnete Korrelation von den Genom-Modellen reicht dabei von 0.77 bis zu 1. Die Korrelation von den Ceeder-Modellen reicht von 0.85 bis zu 1. Die Korrelation von den Ceeder-Modellen ist um 9.08 höher als bei den Genom-Modellen. Wenn sich die Korrelation erhöht, so erhöht sich auch der Aufwand, welcher benötigt wird, um das Screening für die systematischen Reviews durchzuführen.

Kapitel 5

Future Work und Zusammenfassung

5.1 Zusammenfassung

Das Ziel in einem Total-Recall-Szenario ist es, alle relevanten Dokumente für ein Informationsbedürfnis zu finden. Eingesetzt werden Total-Recall-Techniken beispielsweise im Screening von systematischen Reviews, die darauf abzielen, alle zu einem Thema relevanten Dokumente in eine Analyse einzubeziehen. Der Prozess des Screenings kann mehrere Jahre dauern, da es wichtig ist alle verwandten Dokumente zu erhalten. Anders als bei herkömmlichen Suchproblemen geben die Nutzer nach jedem angezeigten Dokument explizit Feedback, ob das zuletzt gezeigte Dokument relevant ist oder nicht. Dieses explizite Relevanzfeedback setzen wir ein, um das Ranking der verbleibenden Dokumente zu verbessern.
Das aktuell beste Verfahren betrachtet Dokumente als TF-IDF-Vektoren, und trainiert auf dem expliziten Nutzer-Feedback ein logistisches Regressionsmodell. Bei dieser Abschlussarbeit haben wir versucht, die Termhäufigkeit der semantisch wichtigen Terme in den Dokumenten zu erhöhen und die Termhäufigkeit der semantisch unwichtigen Terme zu verringern, um den Aufwand für das Screening zu reduzieren. Wir benutzen dafür ein neuronales Sprachmodell, welches mithilfe eines Finetuning-Ansatzes versucht, für jedes Dokument die Termhäufigkeit der semantisch wichtigen Terme zu erhöhen und die Termhäufigkeit der semantisch unwichtigen Terme zu reduzieren. Um das Finetuning mit DeepCT durchzuführen, haben wir drei Strategien entwickelt, mit denen die Relevanzlabel der Terme für das neuronale Sprachmodell automatisch mit den Titeln/Zitate/Referenzen der Dokumente bestimmt werden.
Zur Evaluation unserer Strategien haben wir die Modelle auf dem erweiterten Genom-Datensatz und auf dem kombinierten Ceeder-Datensatz vom Julius-

Kühn-Institut trainiert. Die trainierten Modelle werden von uns verwendet, um die ungewichteten Dokumentrepräsentationen mit der Bedeutung der Terme im Kontext zu erweitern. Die erzeugte Dokumentdarstellung wird dann verwendet, um das bereits bestehende logistische Regressionsmodell weiter zu verbessern, sodass der Aufwand für das Screening verringert wird.
In unserer Evaluation haben wir die Titel/Zitate/Referenzen-Strategie beim Screening von systematischen Reviews mit HiCal verglichen. Dabei fanden wir heraus, dass die Titelstrategie imstande ist, auf dem Ceeder'18, Ceeder'19 und Genom Editierung-Testdatensätzen vom Julius-Kühn-Institut einen geringeren Aufwand zu erzeugen als die ungewichtete Dokumentrepräsentation. Die Zitate-Strategie und die Referenzen-Strategie schafften es nicht, den Aufwand beim Screening im Vergleich zur ungewichteten Dokumentrepräsentation zu verringern.
Wir stellen fest, dass noch weitere Forschung auf dem Gebiet des Screenings von systematischen Reviews mithilfe von kontextabhängigen Termgewichtungen durchgeführt werden kann.

5.2 Future Work

Es gibt verschiedene Forschungsansätze, die noch untersucht werden können, um Total-Recall-Suchen mit kontextabhängigen Termgewichtungen weiter zu verbessern.
Es könnten unterschiedliche BERT-Modelle wie SciBERT[28] oder BioBERT[29] verwendet werden. Diese Sprachmodelle sind im Gegensatz zu dem von uns verwendeten Vanilla-BERT auf nur auf wissenschaftlichen Artikeln trainiert. Die Annahme wäre in diesem Fall, dass diese spezifisch trainierten Sprachmodelle, die Bedeutung der Terme im Kontext von wissenschaftlichen Artikeln besser erfassen können. Diese Annahme könnten wir evaluieren, indem wir die erzeugten Dokumentrepräsentationen beim Screening mit HiCal verwenden und überprüfen, ob die spezifischen Sprachmodelle besser sind als Vanilla-BERT.
Ein weiterer Forschungsansatz wäre es, den Ceeder-Trainingsdatensatz wie beim Genom-Trainingsdatensatz mit den Zitaten und Referenzen der Dokumente zu erweitern. Durch einen größeren Trainingsdatensatz könnten die DeepCT-Modelle ein besseres Verständnis für die Bedeutung der Terme im Kontext von wissenschaftlichen Artikeln, für das Informationsbedürfnis Umwelt, erlangen. Durch einen erweiterten Trainingsdatensatz könnten genauere kontextabhängige Termgewichtungen vorhergesagt werden, sodass der Aufwand vom Screening verringert werden könnte.
Ein weiterer Forschungsansatz, um die kontextabhängigen Termgewichtungen zu verbessern, wäre Query Expansion. Query Expansion wird bei Ad-Hoc-

Suchen verwendet, um das Retrieval zu verbessern. Wir könnten auch das Query Expansion bei den Dokumenten benutzen, um Total-Recall-Suchen durchzuführen. Dabei könnten z. B. verwandte Anfragen zu den Dokumenten hinzugefügt werden, sodass die Sprachmodelle ein besseres Verständnis über den wissenschaftlichen Artikel erlangen können.
Es gibt unterschiedliche Möglichkeiten, mit denen die kontextabhängigen Termgewichtungen weiter verbessert werden können, sodass ein niedrigerer Aufwand beim Screening von systematischen Reviews gebraucht wird.

Literaturverzeichnis

[1] Publikationsflut: Forscher veröffentlichen zu viel. https://www.spiegel.de/wissenschaft/mensch/publikationsflut-forscher-veroeffentlichen-zu-viel-a-1022970.html. Accessed: 09-12-2021. 1

[2] Geri Lobiondo Wood and Judith Haber. Nursing research: Methods and critical appraisal for evidence-based practice. *Journal of Nursing Regulation*, 5, 04 2014. 1

[3] Mustafa Abualsaud, Nimesh Ghelani, Haotian Zhang, Mark D. Smucker, Gordon V. Cormack, and Maura R. Grossman. A system for efficient high-recall retrieval. In *The 41st International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '18, page 1317–1320, New York, NY, USA, 2018. Association for Computing Machinery. 1, 3, 3.1

[4] Gordon V. Cormack and Maura R. Grossman. Evaluation of machine-learning protocols for technology-assisted review in electronic discovery. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '14, page 153–162, New York, NY, USA, 2014. Association for Computing Machinery. 1, 2

[5] Maura R. Grossman, Gordon V. Cormack, Adam Roegiest, and Charles L.A. Clarke. Trec 2015 total recall track overview. In *TREC*, 2015. 2, 3.1

[6] Gordon V. Cormack and Maura R. Grossman. *The Grossman-Cormack Glossary of Technology-Assisted Review.* Fed. Cts. L. Rev., 7(1). 2013. 2

[7] et al. Julian PT Higgins, Sally Green. *Cochrane handbook for systematic reviews of interventions.* olume 5. Wiley Online Library. 2008. 2

[8] Maura R. Grossman and Gordon V. Cormack. Comments on " the implications of rule 26 (g) on the use of technology-assisted review ". 2014. 2

[9] Zhang, Haotian. *Increasing the Efficiency of High-Recall Information Retrieval*. PhD thesis, 2019. 2

[10] Masaharu Yoshioka and Makoto Haraguchi. On a combination of probabilistic and boolean ir models for www document retrieval. *ACM Trans. Asian Lang. Inf. Process.*, 4:340–356, 09 2005. 2

[11] Gordon V. Cormack, Christopher R. Palmer, and Charles L. A. Clarke. Efficient construction of large test collections. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, page 282–289, New York, NY, USA, 1998. Association for Computing Machinery. 2

[12] David D. Lewis and William A. Gale. A sequential algorithm for training text classifiers. In Bruce W. Croft and C. J. van Rijsbergen, editors, *SIGIR '94*, pages 3–12, London, 1994. Springer London. 2

[13] Burr Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009. 2

[14] Gordon V. Cormack and Mona Mojdeh. Machine learning for information retrieval: Trec 2009 web, relevance feedback and legal tracks. In *TREC*, 2009. 2

[15] Gordon V. Cormack and Maura R. Grossman. Autonomy and reliability of continuous active learning for technology-assisted review. *CoRR*, abs/1504.06868, 2015. 2

[16] Evangelos Kanoulas, Dan Li, Leif Azzopardi, and René Spijker. Clef 2017 technologically assisted reviews in empirical medicine overview. 01 2017. 2

[17] Gordon V. Cormack and Maura R. Grossman. Evaluation of machine-learning protocols for technology-assisted review in electronic discovery. New York, NY, USA, 2014. Association for Computing Machinery. 2

[18] Haotian Zhang, Gordon V. Cormack, Maura R. Grossman, and Mark D. Smucker. Evaluating sentence-level relevance feedback for high-recall information retrieval. *Information Retrieval Journal*, 23:1–26, 2020. 2

[19] Zhuyun Dai and Jamie Callan. Context-aware document term weighting for ad-hoc search. In *Proceedings of The Web Conference 2020*, WWW '20, page 1897–1907, New York, NY, USA, 2020. Association for Computing Machinery. 2

[20] Wer, wie, was: Textanalyse über natural language processing mit bert. `https://www.heise.de/hintergrund/Wer-wie-was-Textanalyse-mit-BERT-4864558.html?seite=all`. Accessed: 2010-11-01. 2

[21] Tomas Mikolov, Kai Chen, G.s Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *Proceedings of Workshop at ICLR*, 2013, 01 2013. 2

[22] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018. 2, 3

[23] Rafal Józefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu. Exploring the limits of language modeling. *CoRR*, abs/1602.02410, 2016. 2

[24] Oriol Vinyals, Lukasz Kaiser, Terry Koo, Slav Petrov, Ilya Sutskever, and Geoffrey E. Hinton. Grammar as a foreign language. *CoRR*, abs/1412.7449, 2014. 2

[25] Alexander Turchin and Luisa F. Florez Builes. Using natural language processing to measure and improve quality of diabetes care: A systematic review. *Journal of Diabetes Science and Technology*, 15(3):553–560, 2021. PMID: 33736486. 2

[26] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. *CoRR*, abs/1802.05365, 2018. 2

[27] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, nov 1997. 2

[28] Iz Beltagy, Kyle Lo, and Arman Cohan. Scibert: A pretrained language model for scientific text, 2019. 2, 5.2

[29] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *CoRR*, abs/1901.08746, 2019. 2, 5.2

[30] Alberto Ueda, Rodrygo L. T. Santos, Craig Macdonald, and Iadh Ounis. Structured fine-tuning of contextual embeddings for effective biomedical retrieval. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21,

page 2031–2035, New York, NY, USA, 2021. Association for Computing Machinery. 2

[31] Zhuyun Dai and Jamie Callan. Context-aware sentence/passage term importance estimation for first stage retrieval. *CoRR*, abs/1910.10687, 2019. 2, 3

[32] Zhuyun Dai and Jamie Callan. Context-aware document term weighting for ad-hoc search. In *Proceedings of The Web Conference 2020*, WWW '20, page 1897–1907, New York, NY, USA, 2020. Association for Computing Machinery. 2, 3.3.1

[33] Trevor Strohman, Donald Metzler, Howard Turtle, and W. Croft. Indri: A language-model based search engine for complex queries. *Information Retrieval - IR*, 01 2005. 3.1

[34] Maura Grossman, Gordon Cormack, and Adam Roegiest. Trec 2016 total recall track overview. In *Text Retrieval Conference (TREC)*, 2016. 3.1

[35] Haotian Zhang, Mustafa Abualsaud, Nimesh Ghelani, Mark D. Smucker, Gordon V. Cormack, and Maura R. Grossman. Effective user interaction for high-recall retrieval: Less is more. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, CIKM '18, page 187–196, New York, NY, USA, 2018. Association for Computing Machinery. 3.1

[36] Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R Thomas McCoy, Najoung Kim, Benjamin Van Durme, Samuel R. Bowman, Dipanjan Das, and Ellie Pavlick. What do you learn from context? probing for sentence structure in contextualized word representations, 2019. 3.2

[37] Kirk Roberts, Dina Demner-Fushman, Ellen Voorhees, William Hersh, Steven Bedrick, Alexander Lazar, Shubham Pant, and Funda Meric-Bernstam. Overview of the trec 2019 precision medicine track. 11 2019. 3.4

[38] A.M. Cohen, William Hersh, Kim Peterson, and Po-Yin Yen. Reducing workload in systematic review preparation using automated citation classification. *Journal of the American Medical Informatics Association : JAMIA*, 13:206–19, 03 2006. 4.1

[39] Writing an abstract for your research paper. `https://writing.wisc.edu/handbook/assignments/`

writing-an-abstract-for-your-research-paper/. Accessed: 12.12.2021. 4.2

[40] Acm - policies. https://libraries.acm.org/digital-library/policies. Accessed: 14.12.2021. 4.2