Universität Leipzig
Institut für Informatik
Studiengang Informatik, B.Sc.

Verbesserung der Extraktion Kausaler Zusammenhänge aus dem Internet

Bachelorarbeit

Charly Zimmer
geb. am: 19.06.2000 in Oschatz

Matrikelnummer 3714714

1. Gutachter: Prof. Dr. Martin Potthast
2. Gutachter: Ferdinand Schlatt

Datum der Abgabe: 29. September 2022

Erklärung

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

Leipzig, 29. September 2022

. . . . . .
Charly Zimmer

Zusammenfassung

Diese Arbeit stellt eine Verbesserung von CauseNet [9] hauptsächlich bezüglich der Größe des generierten Kausalgraphen vor. Dafür wurden die ursprünglichen kausalen Muster evaluiert und durch neue ersetzt. Es wurde eine deutliche Steigerung des Recall erzielt, wobei ebenfalls die Precision von 84,5% auf 87,7% erhöht wurde. Das semi-automatische Bootstrapping wurde modifiziert um unter anderem Muster als Seeds zuzulassen. Der verwendete ClueWeb12 Web Crawl wurde gegen Common Crawl ausgetauscht. Außerdem wurden manuell Muster hinzugefügt, die implizite Kausalität in Form von kausalen Verben aufdecken. Durch diese Maßnahmen kann ein umfassenderer Kausalgraph für die Nutzung durch anschließende Arbeiten erzeugt werden.

Inhaltsverzeichnis

Kapitel 1

Einleitung

Es liegt in der Natur des Menschen, sich Beobachtungen erklären zu wollen. Als Werkzeug dafür dient uns das Konzept der Kausalität, mit dem wir erfassen, welche Ursachen einem Effekt zugeordnet werden. Jede Person besitzt andere Kausalvorstellungen. Diese Vorstellungen helfen uns ganz grundlegend dabei, Entscheidungen zu treffen. Bei jeder Entscheidung, selbst bei sehr einfachen, gibt es unzählige Faktoren, die abgewägt werden müssten. Aber niemand betrachtet bei jeder Entscheidung alle Faktoren. Stattdessen hilft uns unser kausales Wissen. Garcia-Retamero and Hoffrage [5] haben experimentell gezeigt, dass wir durch kausales Wissen in der Lage sind, uns auf die wichtigsten Aspekte bei der Entscheidungsfindung zu konzentrieren [5]. Nicht zuletzt deswegen ist es erstrebenswert, dieses Wissen zu sammeln. Und um das zu erreichen, versucht diese Arbeit bestehende Ansätze auf dem Gebiet der Extraktion Kausaler Zusammenhänge zu verbessern. Die Extraktion Kausaler Zusammenhänge (englisch: Causal Extraction) (CE) beschreibt das automatisierte Auffinden von kausalen Relationen („Ursache-Wirkung-Beziehungen“) zwischen Entitätspaaren in natürlicher Sprache. Sie stellt damit ein Teilproblem von Relation Extraction (RE) dar, also der Extraktion beliebiger Relationen (z.B. Synonym-Relation oder „ist-Teil-von-Relation“).

Relevanz der Kausalität Der Kausalität wird in vielen Bereichen, vor allem in der Wissenschaft, eine besonders wichtige Rolle zugesprochen. In der Medizin z.B. werden Annahmen über Kausalbeziehungen zwischen Krankheiten und Symptomen getroffen, ebenso über Medikamente und ihre Wirkung. Die Politikwissenschaft ist besonders von Kausalvorstellungen abhängig, da sie sich hauptsächlich mit der Entscheidungsfindung befasst, welche wie erwähnt eng mit der Kausalität verstrickt ist. In der philosophisch-empirischen Diskussion beschäftigt man sich vor allem mit der Frage, wann eine kausale Annahme gerechtfertigt ist [20], also wann wir von kausalem Wissen sprechen

können. Diese Frage ist insofern spannend, als dass selbst Expertenwissen nicht ausreicht, um eine Annahme zweifelsfrei als wahr zu klassifizieren, denn auch Experten verfügen nicht immer über alle Informationen und können sich auch untereinander widersprechen. Die einzige Validierung einer Kausalvorstellung, die objektiv gesehen bleibt, ist, dass viele andere Menschen die gleiche Vorstellung haben. Aus diesem Grund macht diese Arbeit nicht den Versuch, kausales Wissen zu extrahieren, sondern beschäftigt sich nur mit der Extraktion von kausalen Vorstellungen, um so ein Bild über die im Internet verbreiteten Kausalbeziehungen zu schaffen.

CauseNet Diese Arbeit baut auf einem bestehenden Kausalgraphen namens CauseNet [9] auf und zielt darauf ab, die Verfahren zur Erstellung dieses Graphen zu verbessern. Der grundlegende Ablauf des Systems wird im Folgenden erläutert. Zunächst einmal werden Sätze aus dem Internet extrahiert. Das geschieht mit Hilfe eines Web Crawls. Ein Web Crawl ist eine Sammlung von Websites und deren Inhalt zu einem bestimmten Zeitpunkt. Das Programm iteriert also über den Text jeder Website und extrahiert die einzelnen Sätze. Hier sind beispielhaft zwei Sätze, welche in dem Web Crawl enthalten sein könnten:

(i) Science tells us that smoking causes cancer.
(ii) The cancer eventually resulted in his death.

In einem zweiten Schritt werden alle Sätze auf kausale Relationen geprüft. Kausale Relationen werden durch kausale Muster gekennzeichnet. Ein kausales Muster ist eine Struktur in der Sprache, die verwendet wird, um Kausalität auszudrücken. Es handelt sich hierbei meistens um Verben (z.B. causing, triggering) oder kausale Bindewörter (z.B. because, as, since). Ebenfalls Teil einer kausalen Relation sind Ursache und Wirkung. Im Folgenden sind die zuvor betrachteten Sätze zuzüglich einer Makierung der enthaltenen kausalen Relationen dargestellt (Ursache-Entitäten gepunktet unterstrichen, Wirkung-Entitäten gestrichelt unterstrichen, Kausale Muster fett; Markierung wird im Folgenden fortgeführt).

(i) Science tells us that smoking **causes** cancer.
(ii) The cancer eventually **resulted in** his death.

Schlussendlich werden alle gefundenen kausalen Relationen in einem Graphen gespeichert. Im Beispielsatz (i) ist "cancer" die Wirkung. In Beispielsatz (ii) ist "cancer" wiederum die Ursache. Der daraus entstehende Graph ist in Abbildung 1.1 zu sehen.

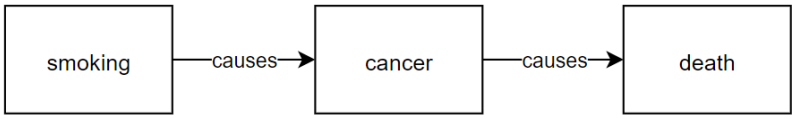

Abbildung 1.1: Kausaler Graph

CauseNet enthält 11 Mio. kausale Relationen und erreicht eine Precision von 83%. Dafür wird der ClueWeb12 Web Crawl[1] verwendet. Dieser beinhaltet 733.019.372 englische Websites aus dem Jahr 2012. Zusätzlich wird insbesondere Wikipedia als Datenquelle verwendet.

Verbesserungsansätze Im Wesentlichen haben wir drei Ansätze zur Verbesserung von CauseNet festgestellt – Precision, Recall und Entitäten. Precision kann im Deutschen mit "Genauigkeit" übersetzt werden. Sei TP die Menge der korrekt als Kausalbeziehung klassifizierten Entitätspaare (True Positives) und FP die Menge der fälschlicherweise als Kausalbeziehung klassifizierten Entitätspaare (False Positives), so ist Precision $= \frac{|TP|}{|TP|+|FP|}$. Um sie zu verbessern könnte man beispielsweise das Problem von mehrdeutigen sprachlichen Mustern adressieren. Ein mehrdeutiges Muster tritt in der natürlichen Sprache einerseits auf, um kausale Beziehungen zu formulieren, kann aber auch in einem nicht-kausalen Kontext auftreten (z.B. leading to). Wüsste man, wann das Muster welche Bedeutung vermitteln soll, so könnte man die Anzahl an False Positives verringern und die Precision steigern. False Positives können ebenfalls durch Negation auftreten. Hier ist ein Beispiel für einen Satz dargestellt, in dem CauseNet eine Kausalität festgestellt hat, obwohl das verwendete Muster negiert wird. Ein Precision-orientierter Ansatz würde sich darauf konzentrieren, einen solchen Satz richtiger Weise als negativ zu klassifizieren:

The symptoms cannot be attributed to another disorder.
CauseNet erkennt: disorder -> symptoms

Recall kann mit "Vollständigkeit" übersetzt werden. Sei zusätzlich FN die Menge der Entitätspaare, die fälschlicherweise als negativ klassifiziert werden (False Negatives), d.h. die Kausalbeziehungen, die nicht erkannt werden. Dann ist Recall $= \frac{|TP|}{|TP|+|FN|}$. Um den Recall zu verbessern könnte man z.B. mehr Muster zulassen oder andere Typen von Mustern implementieren. Im Folgenden ist ein Satz zu sehen, der das kausale Verb "melting" enthält. Kausale Verben sind Verben, die eine Kausalbeziehung implizieren. Um diesen Zusammenhang hervorzuheben, kann der entsprechende Satz semantisch äquivalent umgeformt werden:

[1] `http://lemurproject.org/clueweb12/`

The sun **melted** the ice cream.
The sun **caused** the melting of the ice cream.

Der letzte Verbesserungsansatz sind die Entitäten. Als Entität werden die Ursachen und Wirkungen innerhalb eines Satzes bezeichnet. Das korrekte Abgrenzen der Entitäten stellt ein umfangreiches Problem im Gebiet der Causal Extraction dar. Das liegt vor allem daran, dass eine Entität oftmals mehrere Wörter umfasst. Das Ziel ist, die Entitäten möglichst genau zu erfassen, um so Kausalrelationen mit einem hohen Informationsgehalt zu extrahieren, ohne dabei zu spezifisch zu werden, d.h. Einzelfälle aufzunehmen, die nicht auf abstraktere Konzepte generalisiert werden können. Aber auch schon die Abtrennung von Satzteilen kann ein Problem für das CE-System darstellen. Unten ist ein Satz abgebildet, bei dem die Ursache "'s negligence" und die Wirkung "loss" gefunden wurden. Dabei ist offensichtlich, dass die Ursache eigentlich "another's negligence" hätte sein sollen, von CauseNet aber nicht richtig erkannt wurde.

Someone who suffers loss **caused by** another's negligence may be able to sue for damages to compensate for their harm.
CauseNet erkennt: 's negligence -> loss

Resultate Diese Arbeit legt den Fokus vor allem auf die Vergrößerung des Kausalgraphen unter anderem durch eine Erhöhung des Recall. Dabei muss natürlich trotzdem darauf geachtet werden, die Precision des Systems zu erhalten. Um den Recall zu erhöhen wurden verschiedene Ansätze angewendet und ausgewertet. Zunächst wurde versucht, mehr Kausalitätspaare aus dem in CauseNet verwendeten Corpus zu extrahieren. Der einfachste Weg um das zu erreichen ist das Hinzufügen von neuen Mustern – einerseits manuell und andererseits durch automatische Generierung. Es wurden gezielt Muster hinzugefügt, die unter bisher in CauseNet unbeachtete sprachliche Kategorien fallen, wie beispielsweise kausale Verben. Die Arbeit beschäftigt sich des Weiteren mit der Nutzung von Mustern als Seeds für ein semi-automatisches Bootstrapping-Verfahren, was unseres Wissens nach im CE-Bereich noch nicht zuvor durchgeführt wurde. Außerdem wird die Möglichkeit betrachtet, einen größeren Corpus zu verwenden um so die absolute Anzahl der Kausalrelationen zu erhöhen. Diese Ansätze wurden experimentell getestet und sowohl isoliert als auch im Zusammenspiel miteinander bezüglich verschiedener Metriken evaluiert.

Kapitel 2

Related Work

Extraktionsmethode Systeme zur Bearbeitung von CE-Problemen lassen sich grundlegend in drei Arten einteilen: *knowledge-based*, *statistical machine learning-based* und *deep learning-based* [21]. Das bei CauseNet verwendete Bootstrapping fällt dabei unter *knowledge-based*, da nicht etwa Ansätze aus den Bereichen Machine Learning oder Deep Learning verwendet werden, sondern syntaktische Muster die Gegenstände des Verfahrens sind. Es gibt andere Arbeiten, die kausale Muster nutzen. Garcia [4] verwendet eine statische Liste von syntaktischen Mustern, um Kausalzusammenhänge domänenunabhängig in französischen Texten zu finden. Es handelt sich dabei um 23 Verben, die manuell ausgewählt wurden. Girju [6] stellt ein System vor, in dem kausale Muster verwendet werden, um Kausalzusammenhänge für "Question Answering" zu finden. Als Grundlage dafür dient WordNet [16]. Die Muster, die CauseNet verwendet, entstehen semi-automatisch in einem iterativen Bootstrapping-Verfahren. CauseNet basiert unter anderem auf SNOWBALL [1]. Hier wird ein generelles Verfahren zur iterativen Erzeugung von Relationspaaren und Relationsmustern beschrieben, welches in CauseNet auf die Kausalrelation angewendet wird.

Quellen für Kausalzusammenhänge Diese Arbeit tauscht den in CauseNet verwendeten Web Crawl aus. Im originalen System wird ClueWeb12 verwendet. Dieser wird ersetzt durch Common Crawl[1]. Common Crawl ist eine gemeinnützige Organisation, die einen Web Crawl bereitstellt und kostenlos ihre Datensätze veröffentlicht. Li et al. [14] verwenden einen vorverarbeiteten Corpus [3], der auf Common Crawl basiert für Causal Extraction. Sie nutzen diesen Datensatz, um CausalNet [15] zu reproduzieren und erreichen eine Steigerung der Anzahl von Kausalzusammenhängen von 13.3 Mio. auf 89.1 Mio.

[1] `https://commoncrawl.org/`

Muster-Seeds Die Innovation dieser Arbeit liegt in der Umstrukturierung des Bootstrappings, die es dem System erlaubt, Muster als Seeds zu verwenden. Ein semi-automatischer Prozess mit Mustern als Seeds wurde unseres Wissens nach im Bereich Causal Extraction noch nicht durchgeführt. Saha et al. [18] implementieren ein Bootstrapping-Verfahren, das Muster als Seeds verwendet, allerdings für das generellere Problem der Information Extraction. Li et al. [13] nutzen ganze Kausalzusammenhänge, also (Ursache, Muster, Wirkung)-Tupel, als Seeds für ihr Bootstrapping. Zusätzlich dazu wird ein Algorithums verwendet, der evaluiert, wie relevant die erzeugten Entitätspaare für das jeweilige Muster sind. Dieses System dient allerdings auch nicht für Causal Extraction, sondern für Relation Extraction.

Implizite Kausalität Eine weitere sinnvolle Unterscheidung von CE-Systemen ensteht durch die Betrachtung von impliziter und expliziter Kausalität. Explizite Kausalität wird von Mustern ausgedrückt, die in erster Linie dazu dienen, kausale Zusammenhänge aufzuzeigen (z.B. because, therefore). Implizite Kausalität hingegen tritt dann auf, wenn Formulierungen verwendet werden, die nur selten eine Kausalität markieren (z.B. as, after) oder wenn Formulierungen verwendet werden, mit denen eine Kausalität indirekt einhergeht (z.B. kills, breaks). CauseNet deckt vor allem explizite Kausalität mittels Verben wie "causing" oder "leading to"ab. Implizite Kauslität erreicht das System lediglich durch ambige Muster, was aber eine Beeinträchtigung der Precision mit sich bringt. Diese Arbeit beschäftigt sich auch mit impliziter Kausalität in Form von kausalen Verben. Diese werden häufig nur von CE-Ansätzen betrachtet, die Algorithmen aus dem Machine Learning beinhalten. So z.B. Bethard and Martin [2], die unter anderem den WordNet-Corpus nutzen um einen "Causal Classifier" zu trainieren. Rink et al. [17] wiederum nutzt einen SVM (Support Vector Machine) Classifier, um implizite Kausalzusammenhänge in Graphdarstellungen von Sätzen zu finden.

Satzübergreifende Kausalität Außerdem kann unterschieden werden, ob ein vorliegendes CE-System satzübergreifende Zusammenhänge feststellen kann. Dafür sind andere Muster nötig. CauseNet beschäftigt sich allerdings nur mit Zusammenhängen innerhalb eines Satzes. Auch diese Arbeit bleibt dabei. Es gibt allerdings schon CE-Arbeiten, die sich mit satzübergreifenden Kausalzusammenhängen beschäftigen. Jin et al. [10] stellen ein System vor, das mit Hilfe eines CSNN (Cascaded multi-Structure Neural Network) solche Kausalzusammenhänge auf einem chinesischen Corpus findet.

Kapitel 3

Methoden

3.1 Bootstrapping

Die Extraktion des CauseNet-Kausalgraphen geschieht in drei Phasen – Bootstrapping, Extraction und Causal Concept Spotting. Ein entsprechender Programmablaufplan ist in Abbildung 3.1 zu sehen. Das Bootstrapping dient dazu, syntaktische Muster zu generieren, die in der natürlichen Sprache mit Kausalität assoziiert werden. Im folgenden wird das Bootstrapping genauer erläutert. Dieser Teil des Programms läuft auf einem Datensatz von Wikipedia-Einträgen. Die Extraction-Phase verwendet die im Bootstrapping gefundenen Muster, um Sätze zu extrahieren, die eines der Muster enthalten. Als Corpus dient dafür zum einen der ClueWeb12 Web Crawl und zum anderen Wikipedia. In den Wikipedia-Artikeln werden auch "Infoboxes" und "Listings" verwendet, um weitere kausale Zusammenhänge extrahieren zu können. Nach der Extraction wurde also eine Liste von Sätzen generiert, die alle ein kausales Muster beinhalten. In der letzten Phase – Causal Concept Spotting – werden in jedem dieser Sätze die Entitäten gefunden, also Ursache und Wirkung. Diese Entitätspaare werden verwendet, um den Kausalgraphen aufzubauen.

Da das Ziel verfolgt wird, mehr Muster zu generieren, ist eine Verbesserung des Bootstrappings naheliegend. Dabei handelt es sich um einen iterativen Prozess. Der Ablauf eines Iterationsschrittes soll im Folgenden geschildert werden. Der Input für das Bootstrapping sind so genannte "Seeds". Hierbei handelt es sich lediglich um Entitätspaare, die sehr oft in einen Kausalzusammenhang gebracht werden. In einem ersten Schritt werden alle Sätze des Wikipedia-Datensatzes darauf geprüft, ob einer der Seeds in ihnen enthalten ist, d.h. dass sowohl Ursache als auch Wirkung in dem Satz vorkommen. Wenn das der Fall ist, sucht das System nach dem kausalen Muster in dem Satz. Sätze werden als Enhanced Dependency Graph des Stanford NLP Parsers [19] dargestellt. Ein Dependency Graph ist ein gerichteter Graph, der die Beziehungen zwischen

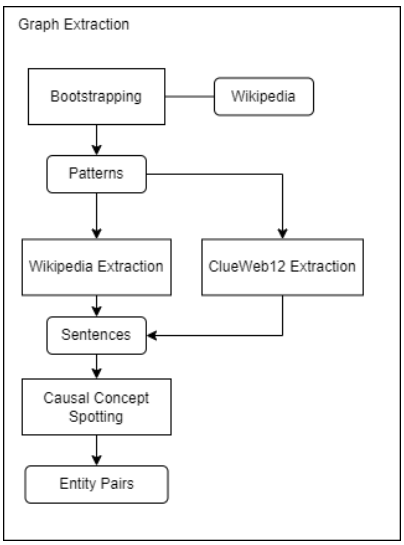

Abbildung 3.1: Programmablaufplan CauseNet

den verschiedenen Teilen innerhalb eines Satzes abbildet. Wörter können dabei zu einer grammatikalischen Struktur zusammengefasst werden. Eine Kante innerhalb eines Dependency Graphs könnte z.B. zwischen einem Verb und seinem entsprechenden Subjekt bestehen. Das kausale Muster ist definiert als der kürzeste Weg zwischen den Entitäten innerhalb des Dependency Graphs. Dieser Prozess, in dem eine Liste von Mustern entsteht heißt Pattern Extraction. Doch nicht alle Muster werden gespeichert. Um Muster vergleichen zu können wird in CauseNet die Metrik "Pattern Support" angewandt. Der Support eines Muster gibt an, wie viele verschiedene Seeds zu seiner Extraktion geführt haben. In der ersten Iteration werden die besten 25 Muster ausgewählt. Die Anzahl der ausgewählten Muster wird danach jede Iteration um 25 erhöht und mit den Mustern aus der vorherigen Iteration verschmolzen, d.h. doppelte Muster werden nur einmal aufgeführt. Die Muster werden wiederum als Input für die Instance Extraction verwendet. Hier wird in allen Sätzen des Corpus nach den zuvor entstandenen Mustern gesucht. Wenn ein Satz ein Muster enthält, dann beinhaltet der Satz auch eine Instanz des Musters. Eine Instanz eines Musters ist ein Entitätspaar, das durch dieses Musters gefunden wurde. Hier wird analog zur Pattern Extraction "Seed Support" als Metrik verwendet. Der Seed Support eines Entitätspaares gibt an, wie viele verschiedene Muster zur Extraktion des Entiätspaares geführt haben. Die besten Entitätspaare werden als Seeds für die nächste Iteration verwendet. Die Anzahl der ausgewählten Seeds wird in jeder Iteration um 10 erhöht. Im CauseNet-System werden zwei Bootstrapping-Iterationen durchgeführt.

3.1.1 Anzahl der Iterationen

Ein erster offensichtlicher Ansatz zur Erhöhung der Anzahl von Mustern ist eine Erhöhung der Anzahl an Bootstrapping-Iterationen. Eine Steigerung des Recall ist damit garantiert, da alle vorher verwendeten Muster und damit auch alle Instanzen dieser Muster weiterhin vorhanden sind. Problematisch bei dieser Methode ist die Erhaltung der Precision des Systems. Werden beispielsweise Muster hinzugefügt, die nur sehr selten verwendet werden, um Kausalität darzustellen beeinflusst dieses Muster die Precison des Gesamtsystems negativ, obwohl der Recall gesteigert wird. Heindorf et al. schreiben in ihrem Paper zu CauseNet: *"Increasing the number of iterations increases recall at the cost of precision"* [9]. Aber warum leidet die Precision bei mehr Boostrapping-Iterationen? Es besteht die Vermutung, dass Muster, die in einer späteren Iteration gefunden werden weniger stark mit der Kausalität assoziiert werden und somit mehr False Positives generieren als Muster aus früheren Iterationen. Als alleinstehende Maßnahme ist dieser Ansatz also nicht sehr vielversprechend. Dennoch kann er in Kombination mit weiteren Anpassungen am Bootstrapping hilfreich sein. Diese Arbeit testet verschiedene Seedlisten mit bis zu vier Iterationen.

Muster im originalen CauseNet Nach zwei Iterationen werden 53 Muster gefunden. Es werden acht Seeds übergeben: *smoking -> cancer, HIV -> AIDS, rain -> flood, rainfall -> flooding, dehydration -> death, radiation -> cancer, earthquake -> tsunami, poison -> death.* Die ersten zehn gefundenen Muster sind in Tabelle 3.1 zu sehen. Muster werden als Enhanced Dependency Graph des Stanford NLP Parsers angegeben. Die Ausdrücke "cause/N" und "effect/N" sind Platzhalter für die jeweiligen Entitäten. Sie werden begleitet von einer Dependency, von der das Muster fordert, dass sie zwischen der Entität und dem Token besteht. Der Token ist das Verbindungsstück zwischen Ursache und Wirkung. Hinter dem Token steht noch, um welche Art von Wort es sich handelt. Die Bezeichnung "VBD" steht beispielsweise für ein Verb in Vergangenheitsform. Die Vorzeichen geben die Richtung einer Dependency an. "+" bedeutet, dass die Kante von der Entität ausgeht, "-" bedeutet, dass die Kante zur Entität hinführt. Es ist intuitiv zu erkennen, dass diese Muster gut sind. Sie sind insofern gut, als dass sie oft verwendet werden, um eine Kausalität auszudrücken. Außerdem kann man hier schon feststellen, dass teilsweise das gleiche Verb in verschiedenen Konjugationen oder mit anderen Dependencys auftritt. Vor allem "causing " ist sehr oft vertreten.

Tabelle 3.1: Die ersten 10 extrahierten Muster aus CauseNet

Cause Dependency	Token/POS	Effect Dependency
cause/N -nsubj	cause/VB	+dobj effect/N
cause/N +acl	causing/VBG	+dobj effect/N
cause/N -nsubj	caused/VBD	+dobj effect/N
cause/N -nsubj	causes/VBZ	+dobj effect/N
cause/N -nmod:agent	caused/VBN	+nsubjpass effect/N
cause/N -nsubj	cause/NN	+nmod:of effect/N
cause/N -nmod:by	caused/VBN	-acl effect/N
cause/N -nmod:to	due/JJ	+nsubj effect/N
cause/N -nsubj	lead/VB	+nmod:to effect/N
cause/N -nsubj	led/VBD	+nmod:to effect/N

3.1.2 Muster als Seeds

Das Bootstrapping kann außerdem verbessert werden, indem man nicht nur Entitätspaare als Seeds zulässt, sondern auch Muster. Der in CauseNet verwendete Ablauf startet mit der Pattern Extraction, die die initialen Seeds als Input erhält. Die daraus entstehenden Muster werden an die Instance Extraction übergeben. Die Idee ist nun, den Startpunkt des Programms umzulegen. Man startet also mit der Instance Extraction und übergibt Muster als Seeds. Es stellt sich die Frage, welche Muster sich als Seeds eignen. Zum einen sollten die Muster häufig in der natürlichen Sprache vorkommen. Es besteht außerdem die Anforderung, dass die Seeds gut generalisieren, d.h. möglichst viele verschiedene Instanzen generieren. Beispielsweise ein Muster, das das Verb "causing" enthält generalisiert gut, während "melting" als kausales Verb nur einen sehr speziellen Kausalzusammenhang darstellt und damit wenige Instanzen zulässt. Und natürlich sollen die Seeds möglichst eindeutig sein, d.h. keine anderen Bedeutungen haben als die Kausalität. Eine gute Quelle für Muster-Seeds sind andere Arbeiten auf dem Gebiet der Causal Extraction. Viele Paper veröffentlichen die Liste der von ihnen genutzten kausalen Muster. Es wurden verschiedene Seed-Mengen verwendet, die aus zwei anderen Arbeiten auf dem Gebiet der Causal Extraction stammen. Dabei gilt zu beachten, dass die Muster meistens nur als Verb ohne Dependency vorliegen und zuvor noch in das richtige Musterformat (siehe Tabelle 3.1) überführt werden mussten. Beispielsweise wird in einer Arbeit das Verb "induce" aufgeführt. Um daraus ein richtiges Muster zu generieren, wurde im Internet nach einem Satz gesucht, der das Verb in einem kausalen Kontext verwendet. Es wurde "Her illness was induced by overwork." gefunden. Aus diesem Satz wurde nun mit Hilfe des in CauseNet verwendeten Algorithmus das kausale Muster extrahiert, das die Entitäten – in

dem Fall "overwork" und "illness" – verbindet. Natürlich kann ein Verb durch verschiedene Konjugationen und Dependencies dargestellt werden. Bei diesem Ansatz wird immer nur eine beliebige Variante abgebildet. Die erste Liste von Mustern, die als Seeds verwendet wurden stammt von Girju and Moldovan [7]. Sie untersuchen in ihrer Arbeit kausale Muster und unterteilen sie nach zwei Kriterien: *Frequency* (gibt an, wie häufig das Muster vorkommt)und *Ambiguity* (gibt an, wie viele verschiedene Bedeutungen das Muster hat). Es wurden die kausalen Muster als Seeds verwendet, denen die Eigenschaften *High Frequency* und *Low Ambiguity* zugeordnet wurden. Luo et al. [15] stellen ebenfalls kausale Muster vor. Als Seeds wurden die Muster der Kategorie "intra-sentence" verwendet. Der Unterschied ist hier, dass sie auch die Konjugation der Verben beachten. Beispielsweise sind "lead to" und "leads to" zwei verschiedene Einträge. Auch bei der Erstellung der Seed-Liste wurde das beachtet. Die entstehende Menge enthält also weniger verschiedene Verben, dafür aber mehrere Varianten eines Verbes. Ergänzend zu den Muster-Seeds können gleichzeitig auch Instanz-Seeds als Input für das Bootstrapping verwendet werden. Beide zuvor genannten Listen wurden sowohl mit den originalen CauseNet-Seeds als auch ohne diese getestet. Ein dritter Ansatz ist, Muster aus der ursprünglichen CauseNet-Mustermenge zu verwenden. Die ersten 15 Muster wurden als Input verwendet. Dadurch werden trotzdem andere Ergebnisse erzeugt als bei CauseNet, einerseits weil nur ein Teil der Liste aus der ersten Iteration verwendet wird, anderseits weil keine Instanz-Seeds verwendet werden.

3.2 Manuelle Anpassung der Mustermenge

Ein anderer Ansatz, um mehr Muster hinzuzufügen ist die Liste der Muster manuell anzupassen. Man setzt also nach dem Bootstrapping an. Durch die ohnehin geringe Anzahl von Mustern ist diese Methode vielversprechend. Mit einer geringen absoluten Anzahl von neu hinzugefügten Mustern kann man die Liste anteilig stark erweitern. Der Arbeitsaufwand für eine solche Erweiterung beläuft sich also auf einen umsetzbaren Umfang. Indem man die Muster nicht generiert, sondern händisch hinzufügt kann man sicher stellen, dass nur solche Muster hinzugefügt werden, die eine ausreichend starke Assoziation mit der Kausalität aufweisen. Das grundlegende Ziel bei der Erhöhung von Recall ist es, mehr kausale Relationen zu extrahieren und so einen umfassenderen Kausalgraphen aufzubauen. Dafür ist es wünschenswert, Muster hinzuzufügen, die möglichst viele neue Entitätspaare aufzeigen. CauseNet versucht genau das durch den zuvor genannten Pattern Support. Es werden die Muster aufgenommen, die durch viele Seeds instanziiert werden. Nun fällt auf, dass sehr viele der CauseNet-Muster Verben sind, die die Kausalität explizit ausdrücken.

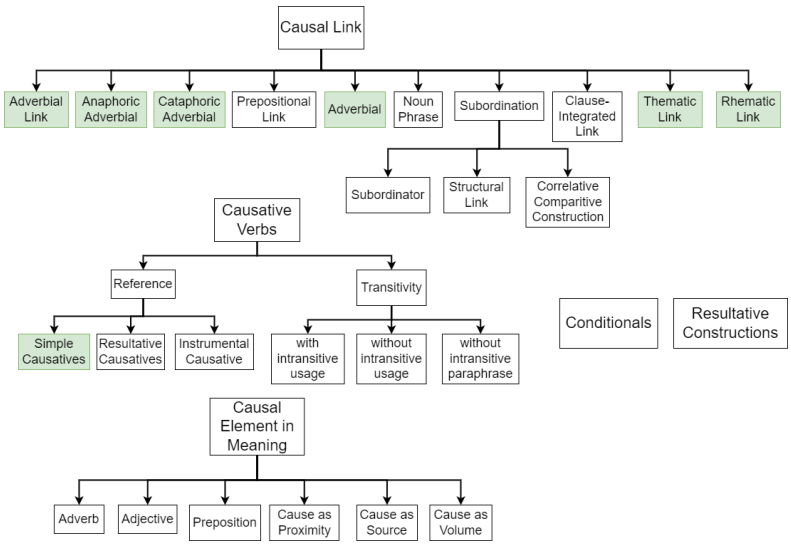

Abbildung 3.2: Taxonomie der Kausalität in natürlicher Sprache (Von CauseNet abgedeckte Bereiche sind grün markiert)

Auch bei einer Erhöhung der Anzahl von Iterationen ändert sich das nicht. Der iterative Bootstrapping-Prozess erschließt kaum neue sprachliche Strukturen, die verwendet werden, um Kausalität auszudrücken. Insofern kann man hier nur von einer vertikalen Skalierung sprechen. Das Bootstrapping ist ziemlich beschränkt auf gewisse Formulierungen.

Es stellt sich die Frage, wie man eine horizontale Skalierung erreicht. Um das zu beantworten muss zunächst festgestellt werden, welche Bereiche der Sprache bisher überhaupt von CauseNet abgedeckt werden. In Abbildung 3.2 ist eine Taxonomie nach Green et al. zu sehen. Es wurde der Versuch gemacht, alle Arten, in denen die Kausalität in der natürlichen Sprache auftritt zu klassifizieren. In der Grafik sind die Strukturen markiert, die von den CauseNet-Mustern erfasst werden. Zunächst einmal muss festgehalten werden, dass das System auf solche Muster beschränkt ist, bei denen beide Entitäten eine Nomenphrase sind. Dadurch fallen viele Bereiche in der Taxonomie von vornherein weg. Auch diese Arbeit betrachtet nur Muster zwischen Nomenphrasen, da hier die Extraktion von Entitäten am einfachsten ist. Trotzdem hindert diese Einschränkung nicht daran, neue sprachliche Strukturen in den kausalen Graphen aufzunehmen. So gibt es z.B. so genannte kausale Verben. Das sind Verben, deren Bedeutung eine Kausalität impliziert. Das Verb "melting" kann beispielsweise als kausales Verb verwendet werden. Der Satz "The sun melted the ice cream." kann umformuliert werden zu "The sun caused the melting of the ice cream.", um so die Kausalität offensichtlicher zu machen. Kausale

Verben sind in der Taxonomie unter "Causative Verbs" zu finden. CauseNet beinhaltet vor allem so genannte "Simple Causatives". Das sind Verben, die hautsächlich verwendet werden, um einen Kausalzusammenhang direkt auszudrücken. Ein Beispiel ist das Wort "causing". Es wird also keine Kausalität impliziert wie bei den kausalen Verben, sondern sie wird explizit formuliert. Das ist die einfachste Form von kausalen Mustern. Das Ziel ist jetzt, auch kausale Verben zu erfassen. Die Muster, die dafür hinzugefügt werden, stammen wiederum aus einer Arbeit von Levin and Hovav. Sie versuchen, kausale Verben zu klassifizieren und stellen dabei eine Liste von solchen bereit. Die ersten 30 Verben wurden gemäß des zuvor genannten Verfahrens in das nötige Musterformat überführt und für die Extraktion auf dem Corpus verwendet. Auch hier wird immer nur eine Variante jedes Verbs betrachtet. Nun kann man argumentieren, dass kausale Verben nicht für das System geeignet sind. Der oben genannte Beispielsatz würde das Entitätspaar [sun, ice cream] erzeugen. Ohne das Wissen über das verwendete Muster ist diese Relation nicht schlüssig. Ein Weg damit umzugehen wäre die Kante des Kausalgraphen mit dem jeweiligen Muster zu versehen. Wenn allerdings das Ziel verfolgt wird, ohne Kantenbeschriftungen auszukommen könnte man die Entitäten entsprechend umformulieren, in diesem Beipsiel zu [sun, melting of ice cream]. In jedem Fall aber drücken kausale Verben eine Kausalität aus und sollten deswegen bei der Causal Extraction berücksichtigt werden.

3.3 Web Crawl

Um die Anzahl von Kausalrelationen zu erhöhen kann neben der Erhöhung des Recalls bezüglich des originalen Corpus auch ein komplett neuer Corpus betrachtet werden. Dazu wurde in dieser Arbeit Common Crawl verwendet. Ein erster Vorteil ist die Größe des Corpus. Der Common Crawl 2021-10, der für diese Arbeit verwendet wurde, enthält 2,7 Milliarden Websites, während der in CauseNet verwendete ClueWeb12 Web Crawl nur 733 Millionen Websites umfasst. Common Crawl ist außerdem aktueller. Das spielt eine große Rolle, da Websites – und damit das Internet – sich sehr schnell verändern. Durch aktuellere Websites können Kausalzusammenhänge aus aktuelleren Themenbereichen extrahiert werden. So kann ClueWeb12 z.B. keine Kausalzusammenhänge über die Covid19-Pandemie enthalten.

3.4 Evaluation

Um auswerten zu können, wie gut die beschriebenen Methoden funktionieren wurden verschiedene Metriken verwendet. Tabelle 3.2 zeigt eine Auflistung der

Tabelle 3.2: Anzahl der generierten Muster nach verwendeten Methoden

Methode	Muster	Bootstrapping-Iterationen
Original CauseNet Patterns	53	2
Top15CauseNet	101	4
Girju Pattern Seeds	113	4
Girju With Instance Seeds	113	4
Luo Pattern Seeds	116	4
Luo With Instance Seeds	116	4
Causative Verbs	30	0

Methoden und wie viele Muster generiert wurden. Zunächst ist die originale Mustermenge aus CauseNet aufgelistet. Außerdem findet sich der Eintrag "Top15CauseNet". Das sind die 15 CauseNet-Muster, die als Seeds verwendet wurden. Außerdem gibt es die Muster-Seeds aus den Arbeiten von Girju [7] und Luo [15], jeweils mit und ohne Instance-Seeds aus CauseNet. Der letzte Eintrag sind die manuell hinzugefügten kausalen Verben. Es wurden jeweils vier Bootstrapping-Iterationen angewendet, außer bei den orginalen CauseNet Mustern, die durch zwei Iterationen entstanden sind und den kausalen Verben, die überhaupt nicht durch Bootstrapping entstanden sind. Insgesamt finden sich in diesen sieben Mengen 205 einzigartige Muster. Diese Mustermenge bestehend aus den 205 Mustern wird im Folgenden als Gesamtmenge K aller Muster bezeichnet. Mit K wurde die Extraction-Phase auf einem Teil des Common Crawls durchgeführt, wobei 3,6 Mio. Instanzen gefunden wurden. Der daraus enstandene Datensatz wurde für die Evaluation verwendet.

3.4.1 Evaluation von Mustern

Frequency Die *frequency* $f(m_i)$ eines Musters $m_i \in K$ beschreibt, wie häufig das Muster vorkommt. Es handelt sich dabei um die absolute Anzahl von Kausalzusammenhängen innerhalb der betrachteten Menge von Sätzen, die eine Instanz des Musters sind. Ein Kausalzusammenhang ist gekennzeichnet durch das Tupel *(cause, pattern, effect)*. Dabei gilt zu beachten, dass ein Satz mehrere Kausalzusammenhänge und auch mehrere Muster enthalten kann.

Precision Seien außerdem $TP(m_i)$ die Menge der True Positives und $FP(m_i)$ die Menge der False Positives eines Musters m_i. Dann ist die *precision* $p(m_i)$ von m_i wie folgt definiert:

$$p(m_i) = \frac{|TP(m_i)|}{|TP(m_i)| + |FP(m_i)|} \tag{3.1}$$

Sätze, die als False Positiv markiert werden sind solche, in denen das Muster nicht verwendet wird, um einen Kausalzusammenhang darzustellen, d.h. in denen das Muster eine andere Bedeutung hat. Eine Feststellung aller False Positives jedes Musters erfordert allerdings sehr viele Ressourcen, weswegen diese Metrik durch manuelles Labeling angenährt wurde. Dabei wurde einer Person ein Satz vorgelegt, der eines der kausalen Muster aus K enthält. Diese sollte dann entscheiden, ob die Bedeutung des Musters innerhalb des Satzes kausal ist. Aufgrund von fehlenden Kapazitäten für die manuelle Klassifizierung wurden nur die 100 Muster mit der höchsten Frequenz betrachtet. Diese 100 Muster machen 97,2% aller gefundenen Kausalzusammenhänge aus. Für jedes der Muster wurden 20 Sätze, die das Muster enthalten, klassifiziert. In der Praxis wird die *precision* also dadurch gekennzeichnet, wie viele der 20 Testsätze als kausal markiert wurden. Zusätzlich wurden einer zweiten Testperson jeweils 3 der 20 Sätze von jedem Muster für eine Kontrollklassifizierung vorgelegt. Bei 84% der Sätze haben beide Testpersonen das gleiche Label vergeben. Es ergibt sich ein Wert für Cohens Kappa[1] von 0,66, was laut Landis and Koch [11] als "beachtliche Übereinstimmung" klassifiziert wird.

Quality Die Idee der Metrik *quality* ist es, *frequency* und *precision* zusammenzuführen, um so Muster anhand einer einzigen Metrik vergleichen zu können. Dafür wurde das gewichtete harmonische Mittel gewählt. Das harmonische Mittel[2] wird verwendet, um den Mittelwert von Verhältniszahlen (Quotienten zweier Größen) zu berechnen. Es ist dem arithmetischen Mittel in diesem Fall vorzuziehen, weil es die Eigenschaft besitzt, dass wenn eine der beiden Größen sehr gering ist, auch die *quality* gering ist. So wird vermieden, dass Muster, die eine niedrige *precision* haben, durch eine hohe *frequency* trotzdem einen guten Wert erreichen. Es wird die normierte *frequency* $f'(m_i) = \frac{f(m_i)}{\max_{m \in M} f(m)}$ eines Musters m_i verwendet. Die *precision* ist bereits normiert. Die *quality* $q_x(m_i)$ eines Musters m_i ist folgendermaßen definiert:

$$q_x(m_i) = \frac{x+1}{\frac{x}{p(m_i)} + \frac{1}{f'(m_i)}} \tag{3.2}$$

Der Parameter x gibt somit die Gewichtung der *precision* an, während die Gewichtung für die *frequency* auf 1 festgelegt wird. Je nachdem, welche der beiden Größen maximiert werden soll kann dieser Parameter anders gewählt werden. Allerdings ist zu empfehlen, die *precision* stärker zu gewichten, also einer Wert zu wählen, der größer als 1 ist. So haben nur Muster, die wenige False Positives generieren, eine hohe *quality*.

[1] `https://de.wikipedia.org/wiki/Cohens_Kappa`
[2] `https://de.wikipedia.org/wiki/Harmonisches_Mittel`

3.4.2 Evaluation von Mustermengen

Die drei oben genannten Metriken sollen ebenfalls auf Mustermengen angewendet werden können, um diese untereinander vergleichen zu können. Am offensichtlichsten ist es, jeweils den Durchschnitt über alle Muster in der Liste zu bilden. Die durchschnittliche *frequency* bezieht allerdings die Anzahl von Mustern in der Liste nicht mit ein und macht damit keine klare Aussage über die Anzahl von Kausalzusammenhängen, die von der Liste gefunden werden. Und auch die durchschnittliche *precision* ist nicht sehr gut geeignet, da hier nicht darauf geachtet wird, wie hoch die *frequency* des jeweiligen Musters ist. Deswegen werden für die Bewertung von Mustermengen zusätzliche Größen eingeführt.

Proportional Frequency Die *proportional frequency* $\tilde{f}(M)$ einer Menge $M \subset K$ von Mustern ist folgendermaßen definiert:

$$\tilde{f}(M) = \sum_{m \in M} \frac{f(m)}{|K|} \tag{3.3}$$

Die *proportional frequency* gibt also an, wie groß der Anteil an den gefundenen Kausalbeziehungen ist, der von den Mustern in der Liste abgedeckt wird. Sie bezieht sich immer auf K. Die *proportional frequency* der Menge von allen 205 Mustern ist demnach gleich 1. Diese Größe macht eine Aussage über den Recall der jeweiligen Mustermenge. Wenn die *proportional frequency* steigt, so steigt auch der Recall des Systems.

Estimated Precision Die *precision* $p(M)$ einer Mustermenge $M \subset K$ ist definiert als:

$$p(M) = \frac{\sum_{m \in M} |TP(m)|}{\sum_{m \in M} |TP(m)| + |FP(m)|} \tag{3.4}$$

Dabei gilt zu beachten, dass diese Mengen nur mit sehr hohem Aufwand bestimmt werden können. Deswegen wurde zur Berechnung die angenährte *precision* (siehe oben) der Muster verwendet, welche durch manuelles Labeling ensteht. In der Praxis wird also diese Formel für die *estimated precsion* verwendet:

$$p(M) \approx \frac{\sum_{m \in M} p(m) f(m)}{\sum_{m \in M} f(m)} \tag{3.5}$$

Da nicht alle Muster manuell evaluiert wurden, sondern nur die 100 Muster mit der höchsten *frequency* kann diese Formel nicht vollumfänglich angewendet werden. Muster ohne eine *precision* werden nicht beachtet.

Kapitel 4

Experimente

4.1 Web Crawl

Um die Web Crawls vergleichen zu können, wurde die Extraction-Phase auf beiden nur mit den originalen CauseNet-Mustern durchgeführt. Dabei wurden 30,7 Mio. Kausalrelationen auf ClueWeb gefunden. Auf Common Crawl wurden mit den gleichen Mustern 91,5 Mio. Mio. Relationen extrahiert. Damit wurde ein ähnliches Ergebnis wie in der zuvor genannten Arbeit von Li et al. [14] erreicht, die ebenfalls Common Crawl verwendet haben. Es sollte allerdings beachtet werden, dass wir das Causal Concept Spotting aufgrund von limitierten Ressourcen nicht durchgeführt haben. Durch diesen Programmteil würde ein erheblicher Teil der gefundenen Relationen wieder verworfen werden. Laut Angaben der CauseNet-Autoren bestehen nach dem Causal Concept Spotting auf ClueWeb noch 11. Mio Relationen [9]. Um eine Schätzung für Common Crawl zu erhalten wurde für einen Teil der Relationen das Causal Concept Spotting durchgeführt. Daraus ergab sich, dass 66,5% der Relationen verworfen werden. Hochgerechnet auf den gesamten Datensatz ergibt das eine Schätzung von 32,0 Mio. Relationen nach dem Causal Concept Spotting. In den folgenden Experimenten wird die *frequency* immmer bezüglich Common Crawl betrachtet. Außerdem wurden Sätze gefunden, die aktuellere Themengebiete abdecken, als es bei ClueWeb12 der Fall war. Ein Beispielsatz aus Common Crawl mit einem Kausalzusammenhang zum Thema Covid-19 ist folgender: *"The disruption of school classes, along with other problems associated with the COVID pandemic, brought into sharper focus just how big this problem was."*

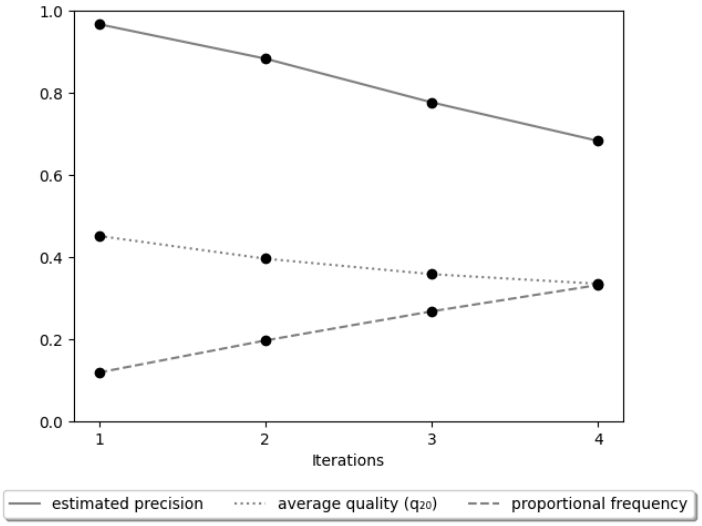

Abbildung 4.1: Entwicklung der CauseNet-Mustermenge bei Erhöhung der Anzahl von Bootstrapping-Iterationen

4.2 Bootstrapping-Iterationen

In einem Experiment wurde das Bootstrapping mit den ursprünglichen Instanz-Seeds aus CauseNet ohne zusätzliche Muster-Seeds durchgeführt. So kann der Effekt von Bootstrapping-Iterationen isoliert ausgewertet werden. Durch bis zu vier Iterationen wurden Mustermengen erstellt, die auf *proportional frequency*, *estimated precision* und *average quality* analysiert wurden. Abbildung 4.1 zeigt die Ergebnisse. Da bei jeder Iteration Muster zur Liste hinzugefügt werden, steigt erwartungsgemäß die *proportional frequency* mit jeder Iteration. Bei der *estimated precision* ist ein linearer Abfall zu erkennen. Die zuvor genannte Vermutung, dass Muster, die in späteren Iterationen gefunden werden, tendenziell weniger eindeutig sind, wird dadurch bestärkt. Die *average quality* sinkt ebenfalls bei höheren Iterationsanzahlen, allerdings nicht sehr stark.

4.3 Mustermengen

Einerseits wurden Mustermengen durch Muster-Seeds generiert, anderseits wurde manuell eine Liste von kausalen Verben händisch erstellt. Abbildung 4.2 zeigt einen Vergleich von diesen Listen bezüglich der oben beschriebenen Kenngrößen. An vorderster Stelle im Diagramm ist außerdem die CauseNet-Mustermenge aufgeführt. Alle Listen wurden durch zwei Bootstrapping-Iterationen generiert, ausgenommen von den kausalen Verben, weil diese nicht durch das Bootstrap-

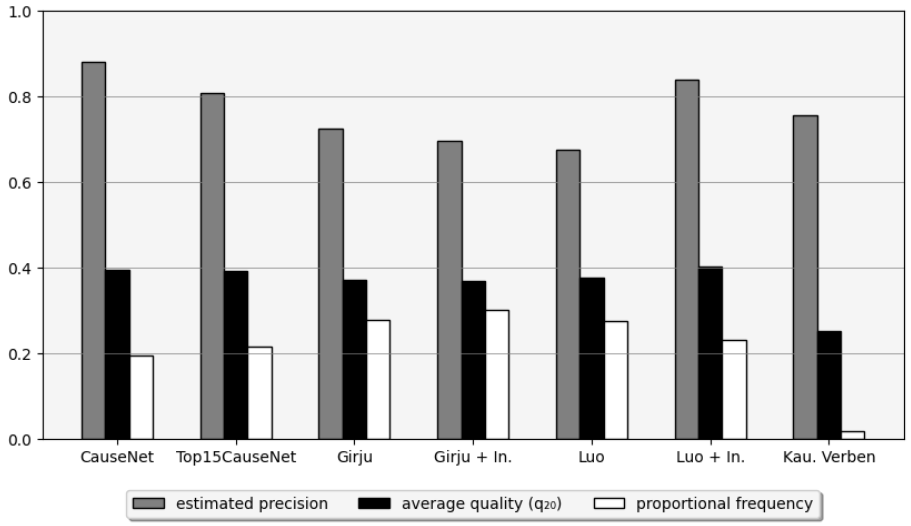

Abbildung 4.2: Vergleich der neu erzeugten Mustermengen, jeweils 2 Bootstrapping-Iterationen

ping entstanden sind. Die Seed-Listen aus den Arbeiten von Girju and Moldovan [7] und Luo et al. [15] wurden jeweils einmal mit CauseNet-Instanz-Seeds und einmal ohne angewendet. Die Liste "Top15CauseNet" ist die Liste, die Muster aus CauseNet als Seeds verwendet. Auffällig ist, dass die kausalen Verben eine sehr niedrige *frequency* haben, dafür aber einen guten Wert bei *precision* aufweisen. Insgesamt ist die *quality* hier trotzdem am geringsten. Keiner der beiden Listen, die nur durch Muster-Seeds entstanden sind erreichen eine *estimated precision*, die so hoch wie die von CauseNet ist.

4.4 Bestmögliche Mustermengen

Nachdem nun alle Ansätze isoliert betrachtet wurden, sollen im Folgenden die besten Muster unter allen 205 generierten Mustern zu Mengen zusammengefasst werden. In Tabelle 4.2 sind die Muster mit der höchsten *frequency* zu sehen. Tabelle 4.3 zeigt die Muster mit der höchsten *precision*. Zu beachten ist dabei, dass alle Muster in dieser Tabelle einen Wert von 1 haben, d.h. alle Sätze wurden als True Positiv klassifiziert. Es gibt noch 29 weitere Muster, die diesen perfekten Wert erreicht haben. Das ist einerseits damit zu erklären, dass 20 eine relativ geringe Größe für eine Testmenge ist, und anderseits damit, dass es viele Muster gibt, die tatsächlich nur eine Bedeutung haben. Die Muster mit den höchsten Werten für *quality* sind in Tabelle 4.4 zu finden. Hier sind die zehn Muster mit der höchsten q_{20}-*quality* dargestellt und

Tabelle 4.1: Verbesserung zu CauseNet

Vergleich der finalen Mustermengen				
	f	$\tilde{f}$	p(train)	p(test)
CauseNet Patterns	91,5 Mio.	19,6%	88,3%	84,5%
High Frequency, High Precision	135,6 Mio.	29,2%	92,5%	87,7%
Top50 Quality (q_{20})	163,2 Mio.	35,3%	81,1%	70,1%
Top50 Precision	102,5 Mio.	22,0%	98,1%	94,7%
Top50 Frequency	390,9 Mio.	84,3%	30,6%	26,5%

ebenfalls die entsprechenden Werte für anders gewählte Parameter. Tabelle 4.1 zeigt einen Vergleich von CauseNet zu gezielt ausgewählten Mustermengen aus K. In der ersten Zeile ist die Menge der 53 Muster aufgelistet, die bei CauseNet verwendet wurden. Die Menge "High Frequency, High Precision" beinhaltet nur die Muster, die eine *precision* von mindesten 75% aufweisen und zu den 100 Mustern mit der höchsten *frequency* gehören. Insgesamt sind hier 57 Muster vertreten. Außerdem wurden die 50 Muster mit der höchsten *frequency*, *precision* und *quality (q_{20})* jeweils zu Mengen zusammengefasst und analysiert. Es wird sowohl die *frequency* (f) als auch die *proportional frequency* ($\tilde{f}$) dargestellt. Insgesamt wurden durch alle in K enthaltenen Muster 463,6 Mio. Relationen extrahiert. Für die *estimated precision* (p) der Mengen gibt es einerseits eine Spalte für den Datensatz, auf dem die *precision* ermittelt wurde (train) und eine Spalte für einen unabhängigen Testdatensatz (test). Für die Erstellung dieses Testdatensatzes wurden zusätzlich fünf Sätze pro Muster auf die gleiche Weise wie zuvor beschrieben manuell klassifiziert.

Tabelle 4.2: Die 10 Muster mit der höchsten frequency

Cause Dependency	Token/POS	Effect Dependency	frequency	precision
cause/N -nsubj	include/VBP	+dobj effect/N	6.338.023	0,0
cause/N -nsubj	includes/VBZ	+dobj effect/N	4.792.862	0,05
cause/N -nsubj	had/VBD	+dobj effect/N	3.700.453	0,15
cause/N -dobj	have/VB	+nsubj effect/N	3.190.629	0,0
cause/N -nsubj	include/VB	+dobj effect/N	1.477.418	0,0
cause/N -dobj	made/VBD	+nsubj effect/N	1.258.210	0,0
cause/N -nsubjpass	based/VBN	+nmod:on effect/N	1.231.264	0,0
cause/N -nmod:with	associated/VBN	-acl effect/N	1.039.405	0,75
cause/N -nsubj	included/VBD	+dobj effect/N	1.018.878	0,1
cause/N -nmod:to	linked/VBN	-acl effect/N	983.707	0,35

Tabelle 4.3: Die 10 Muster mit der höchsten precision

Cause Dependency	Token/POS	Effect Dependency	precision	frequency
cause/N +acl	causing/VBG	+dobj effect/N	1,0	113.653
cause/N +acl	resulting/VBG	+nmod:in effect/N	1,0	138.001
cause/N +acl:to	prevent/VB	+dobj effect/N	1,0	195.512
cause/N -nmod:agent	caused/VBN	+nsubjpass effect/N	1,0	204.321
cause/N -nmod:by	caused/VBN	-acl effect/N	1,0	408.424
cause/N -nmod:from	resulting/VBG	-acl effect/N	1,0	141.503
cause/N -nmod:of	result/NN	+nsubj effect/N	1,0	248.603
cause/N -nmod:of	result/NN	-nmod:as effect/N	1,0	66.939
cause/N -nmod:to	due/JJ	+nsubj effect/N	1,0	276.424
cause/N -nmod:to	due/JJ	-amod effect/N	1,0	239.340

Tabelle 4.4: Die 10 Muster mit der höchsten quality (q_{20})

Cause Dependency	Token/POS	Effect Dependency	frequency	precision	q_{20}	q_{40}	q_1
cause/N -nsubj	cause/VB	+dobj effect/N	749.484	1,0	0,738	0,846	0,211
cause/N -nsubj	lead/VB	+nmod:to effect/N	544.180	1,0	0,664	0,794	0,158
cause/N -nmod:with	associated/VBN	-acl effect/N	1.039.405	0,75	0,641	0,69	0,269
cause/N -nsubj	results/VBZ	+nmod:in effect/N	479.733	1,0	0,632	0,771	0,141
cause/N -nmod:on	based/VBN	-acl effect/N	934.295	0,75	0,628	0,682	0,246
cause/N -nsubj	result/VB	+nmod:in effect/N	447.352	1,0	0,615	0,757	0,132
cause/N -nsubj	responsible/JJ	+nmod:for effect/N	759.594	0,75	0,6	0,665	0,207
cause/N -nsubj	brought/VBD	+dobj effect/N	421.116	1,0	0,599	0,745	0,125
cause/N -nmod:by	caused/VBN	-acl effect/N	408.424	1,0	0,591	0,738	0,121
cause/N -nsubj	led/VBD	+nmod:to effect/N	378.349	1,0	0,571	0,722	0,113

Kapitel 5

Diskussion

Zu den Muster-Seeds lässt sich sagen, dass keine der beiden getesteten Seed-Mengen eine klar bessere Performance erreicht hat als die ursprünglichen CauseNet-Muster. Die Seeds aus den Arbeiten von Girju und Luo haben lediglich eine bessere *frequency* erzielt. Wenn man bedenkt, dass das Ziel dieser Arbeit vor allem eine Vergrößerung des Kausalgraphen war, kann man hier von einem Erfolg sprechen. Allerdings ist die *precision* der resultierenden Mustermengen deutlich niedriger (siehe Tabelle 4.2). Dennoch kann man daraus nicht schließen, dass Muster-Seeds schlechter als Instanz-Seeds sind. Es besteht die Vermutung, dass die betrachteten Muster-Seeds nicht optimal gewählt wurden. Für diese Vermutung spricht, dass die *precision* der Seeds nicht sehr hoch ist. In Abbildung 5.1 ist abzulesen, dass die Werte für *estimated precision* vor der ersten Bootstrapping-Iteration nur bei 72,9% (Girju) und 82,2% (Luo) liegen. Vergleichsweise liegt der Wert für die originalen CauseNet-Muster nach der ersten Iteration bei 96,7% (siehe Tabelle 4.1). Außerdem fällt auf, dass die Werte in Abbilung 5.1 in der ersten Iteration steigen, bevor sie dann stetig fallen. Auch das könnte dafür sprechen, dass die Muster-Seeds nicht gut gewählt wurden. Anschließende Arbeiten auf diesem Gebiet könnten mit besserern Seeds durchaus mehr Erfolg erzielen.

Die kausalen Verben haben eine sehr niedrige *frequency*. Damit tragen sie nicht viel zur Erweiterung des Kausalgraphen bei. Und auch ihre *precision* ist entgegen den Erwartungen nicht besonders hoch. Das könnte unter anderem daran liegen, dass die Muster nicht richtig modelliert wurden. Bei der Evaluierung ist aufgefallen, dass oftmals Satzstrukturen von den Mustern erkannt wurden, die nicht vorgesehen waren. So z.B. bei dem kausalen Verb "cutting". Es wurde mit folgendem Muster modelliert: "[[cause]]/N -nsubj cut/VBD +dobj [[effect]]/N". Dieses Muster hat teilweise Sätze erfasst, in denen das kausale Verb nicht wie vorgesehen verwendet wurde. Ein Satz, der gefunden wurde ist: *"Peel the two large potatoes or six small potatoes, cut into two or four halfs and*

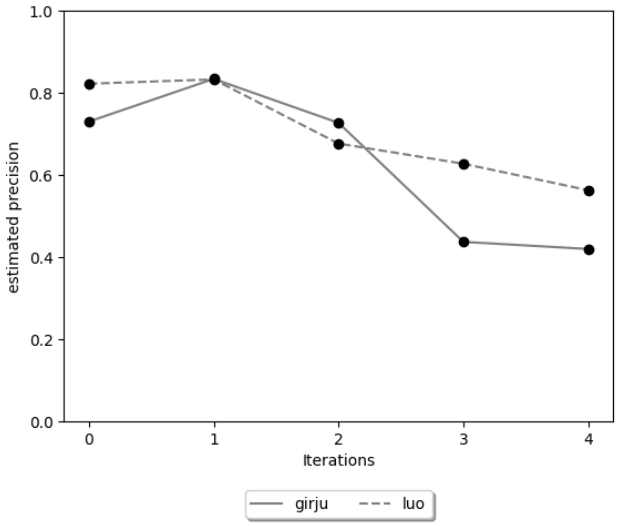

Abbildung 5.1: Precision der Muster-Seed-Mengen nach Bootstrapping-Iterationen

boil them." mit der Ursache "potatoes" und der Wirkung "Peel". Hier wird das Verb im Imperativ verwendet, es beinhaltet also nicht die Ursache-Entität, die für einen Kausalzusammenhang erforderlich ist. Ein Grund für diese Fehlklassifizierung könnte neben einem unpassend modellierten Muster auch ein falsch konstruierter Dependency Graph sein.

Das Austauschen des Web Crawls ist positiv zu bewerten. Zum einen konnte die Größe des Kausalgraphen signifikant erhöht werden, zum anderen ist Common Crawl aktueller und beinhaltet im Vergleich zu ClueWeb12 Kausalzusammenhänge in aktuelleren Themengebieten. Auch die durchgeführten Experimente haben diese Verbesserungen bestätigt.

Betrachtet man alle angewendeten Methoden im Zusammenspiel erreicht diese Arbeit eine Verbesserung gegenüber dem ursprünglichen CauseNet-System. Durch die neu generierten Muster wurde eine Steigerung des Recall zweifelsfrei erreicht, während gleichzeitig die *precision* erhöht werden konnte.

Anhang A

Muster

Tabelle A.1: Girju Muster-Seeds

Cause Dependency	Token/POS	Effect Dependency
cause/N -nmod:agent	induced/VBN	+nsubjpass effect/N
cause/N -nsubj	give/VBP	+nmod:to effect/N
cause/N -nsubj	give/VB +dobj rise/VB	+nmod:to effect/N
cause/N -nsubj	produced/VBD	+dobj effect/N
cause/N -nsubj	generates/VBZ	+dobj effect/N
cause/N -nmod:agent	effected/VBN	+nsubjpass effect/N
cause/N -nsubj	bring/VB	+dobj effect/N
cause/N -nsubj	provoke/VB	+dobj effect/N
cause/N -nsubj	arouse/VBP	+dobj effect/N
cause/N -nsubj	elicits/VBZ	+dobj effect/N
cause/N -nsubj	lead/VB	+nmod:to effect/N
cause/N +acl:relcl	trigger/VB	+dobj effect/N
cause/N -nmod:from	derive/VB	+dobj effect/N
cause/N -dobj	associate/VBP	+nmod:with effect/N
cause/N -nmod:to	related/VBN	+nsubj effect/N
cause/N -nmod:to	linked/VBN	+nsubjpass effect/N
cause/N -nmod:from	stem/VBP	+nsubj effect/N
cause/N -nmod:from	originate/VB	+nsubj effect/N
cause/N -nsubj	bring/VBP	+dobj effect/N
cause/N -nsubj	led/VBD	+nmod:to effect/N
cause/N -nsubj	trigger/VB	+dobj effect/N
cause/N -nmod:from	result/VB	+nsubj effect/N

Tabelle A.2: Luo Muster-Seeds

Cause Dependency	Token/POS	Effect Dependency
cause/N -nsubj	lead/VB	+nmod:to effect/N
cause/N -nsubj	leads/VBZ	+nmod:to effect/N
cause/N -nsubj	led/VBD	+nmod:to effect/N
cause/N +acl	leading/VBG	+nmod:to effect/N
cause/N -nsubj	give/VBP	+nmod:to effect/N
cause/N -nsubj	give/VB +dobj rise/VB	+nmod:to effect/N
cause/N -nsubj	gave/VBD +dobj rise/NN	+nmod:to effect/N
cause/N -nsubj	given/VBN	+nmod:to effect/N
cause/N +acl	giving/VBG	+nmod:to effect/N
cause/N -nsubj	induce/VB	+dobj effect/N
cause/N +acl	inducing/VBG	+dobj effect/N
cause/N -nsubj	induces/VBZ	+dobj effect/N
cause/N -nmod:agent	induced/VBN	+nsubjpass effect/N
cause/N -nsubj	cause/VB	+dobj effect/N
cause/N +acl	causing/VBG	+dobj effect/N
cause/N -nsubj	causes/VBZ	+dobj effect/N
cause/N -nsubj	caused/VBD	+dobj effect/N
cause/N -nmod:by	caused/VBN	-acl effect/N
cause/N -nsubj	bring/VB	+dobj effect/N
cause/N +acl	brought/VBN	+dobj effect/N
cause/N +acl	bringing/VBG	+nmod:on effect/N
cause/N -nsubj	brings/VBZ	+nmod:on effect/N
cause/N -nmod:from	result/VB	+nsubj effect/N
cause/N -nmod:from	resulting/VBG	-acl effect/N
cause/N -nmod:from	results/VBZ	+nsubj effect/N
cause/N -nmod:from	resulted/VBD	+nsubj effect/N
cause/N +nsubj	reason/NN	+nmod:for effect/N
cause/N +nsubj	reasons/NNS	+nmod:for effect/N
cause/N +nsubj	reason/NN	+nmod:of effect/N
cause/N +nsubj	reasons/NNS	+nmod:of effect/N
cause/N -nmod:of	effect/NN	-nsubj effect/N

Tabelle A.3: High Frequency, High Precision (1/2)

Cause Dependency	Token/POS	Effect Dependency
cause/N +acl	causing/VBG	+dobj effect/N
cause/N +acl	resulting/VBG	+nmod:in effect/N
cause/N +acl:to	prevent/VB	+dobj effect/N
cause/N -nmod:agent	caused/VBN	+nsubjpass effect/N
cause/N -nmod:by	caused/VBN	-acl effect/N
cause/N -nmod:from	resulting/VBG	-acl effect/N
cause/N -nmod:of	result/NN	+nsubj effect/N
cause/N -nmod:of	result/NN	-nmod:as effect/N
cause/N -nmod:to	due/JJ	+nsubj effect/N
cause/N -nmod:to	due/JJ	-amod effect/N
cause/N -nsubj	broke/VBD	+dobj effect/N
cause/N -nsubj	brought/VBD	+dobj effect/N
cause/N -nsubj	cause/NN	+nmod:of effect/N
cause/N -nsubj	cause/VB	+dobj effect/N
cause/N -nsubj	cause/VBP	+dobj effect/N
cause/N -nsubj	caused/VBD	+dobj effect/N
cause/N -nsubj	caused/VBN	+dobj effect/N
cause/N -nsubj	causes/VBZ	+dobj effect/N
cause/N -nsubj	causing/VBG	+dobj effect/N
cause/N -nsubj	created/VBD	+dobj effect/N
cause/N -nsubj	created/VBN	+dobj effect/N
cause/N -nsubj	generate/VB	+dobj effect/N
cause/N -nsubj	generates/VBZ	+dobj effect/N
cause/N -nsubj	killed/VBD	+dobj effect/N
cause/N -nsubj	lead/VB	+nmod:to effect/N
cause/N -nsubj	led/VBD	+nmod:to effect/N
cause/N -nsubj	led/VBN	+nmod:to effect/N
cause/N -nsubj	opened/VBD	+dobj effect/N
cause/N -nsubj	produce/VB	+dobj effect/N
cause/N -nsubj	produced/VBD	+dobj effect/N
cause/N -nsubj	reduces/VBZ	+dobj effect/N
cause/N -nsubj	result/VB	+nmod:in effect/N
cause/N -nsubj	result/VBP	+nmod:in effect/N
cause/N -nsubj	resulted/VBD	+nmod:in effect/N
cause/N -nsubj	resulted/VBN	+nmod:in effect/N
cause/N -nsubj	results/VBZ	+nmod:in effect/N
cause/N -nsubj	trigger/VB	+dobj effect/N
cause/N -nsubj:xsubj	cause/VB	+dobj effect/N
cause/N -nsubj:xsubj	prevent/VB	+dobj effect/N
cause/N -nsubj	contribute/VBP	+nmod:to effect/N

Tabelle A.4: High Frequency, High Precision (2/2)

Cause Dependency	Token/POS	Effect Dependency
cause/N -nsubj	produce/VBP	+dobj effect/N
cause/N -nmod:to	attributed/VBN	+nsubjpass effect/N
cause/N -nsubj	break/VB	+dobj effect/N
cause/N -nsubj	contributed/VBD	+nmod:to effect/N
cause/N -nmod:by	brought/VBN	-acl effect/N
cause/N -nsubj	bring/VBP	+dobj effect/N
cause/N -nsubj	contribute/VB	+nmod:to effect/N
cause/N -nsubj	contributed/VBN	+nmod:to effect/N
cause/N -nsubj	leads/VBZ	+nmod:to effect/N
cause/N -nsubj	closed/VBD	+dobj effect/N
cause/N -nsubj	cut/VBD	+dobj effect/N
cause/N -nsubj	move/VB	+dobj effect/N
cause/N -nmod:on	based/VBN	-acl effect/N
cause/N -nmod:with	associated/VBN	-acl effect/N
cause/N -nsubj	contributes/VBZ	+nmod:to effect/N
cause/N -nsubj	responsible/JJ	+nmod:for effect/N
cause/N -nsubj	showed/VBD	+dobj effect/N

Literaturverzeichnis

[1] E. Agichtein and L. Gravano. *Snowball*: extracting relations from large plain-text collections. In *Proceedings of the fifth ACM conference on Digital libraries - DL '00*, pages 85–94, San Antonio, Texas, United States, 2000. ACM Press. ISBN 978-1-58113-231-1. doi: 10.1145/336597.336644. URL `http://portal.acm.org/citation.cfm?doid=336597.336644`.

[2] S. Bethard and J. Martin. Learning Semantic Links from a Corpus of Parallel Temporal and Causal Relations. pages 177–180, Jan. 2008. doi: 10.3115/1557690.1557740.

[3] C. Buck, K. Heafield, and B. van Ooyen. N-gram Counts and Language Models from the Common Crawl. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 3579–3584, Reykjavik, Iceland, May 2014. European Language Resources Association (ELRA). URL `http://www.lrec-conf.org/proceedings/lrec2014/pdf/1097_Paper.pdf`.

[4] D. Garcia. COATIS, an NLP system to locate expressions of actions connected by causality links. In E. Plaza and R. Benjamins, editors, *Knowledge Acquisition, Modeling and Management*, Lecture Notes in Computer Science, pages 347–352, Berlin, Heidelberg, 1997. Springer. ISBN 978-3-540-69606-3. doi: 10.1007/BFb0026799.

[5] R. Garcia-Retamero and U. Hoffrage. How causal knowledge simplifies decision-making. *Minds and Machines*, 16(3):365–380, Aug. 2006. ISSN 1572-8641. doi: 10.1007/s11023-006-9035-1. URL `https://doi.org/10.1007/s11023-006-9035-1`.

[6] R. Girju. Automatic detection of causal relations for Question Answering. In *Proceedings of the ACL 2003 workshop on Multilingual summarization and question answering - Volume 12*, MultiSumQA '03, pages 76–83, USA, July 2003. Association for Computational Linguistics. doi: 10.3115/1119312.1119322. URL `https://doi.org/10.3115/1119312.1119322`.

[7] R. Girju and D. Moldovan. Text Mining for Causal Relations. page 5, May 2002.

[8] R. Green, C. A. Bean, and S. H. Myaeng. *The Semantics of Relationships*. URL `https://link.springer.com/book/10.1007/978-94-017-0073-3`.

[9] S. Heindorf, Y. Scholten, H. Wachsmuth, A.-C. Ngonga Ngomo, and M. Potthast. CauseNet: Towards a Causality Graph Extracted from the Web. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 3023–3030, Virtual Event Ireland, Oct. 2020. ACM. ISBN 978-1-4503-6859-9. doi: 10.1145/3340531.3412763. URL `https://dl.acm.org/doi/10.1145/3340531.3412763`.

[10] X. Jin, X. Wang, X. Luo, S. Huang, and S. Gu. Inter-sentence and Implicit Causality Extraction from Chinese Corpus. In H. W. Lauw, R. C.-W. Wong, A. Ntoulas, E.-P. Lim, S.-K. Ng, and S. J. Pan, editors, *Advances in Knowledge Discovery and Data Mining*, Lecture Notes in Computer Science, pages 739–751, Cham, 2020. Springer International Publishing. ISBN 978-3-030-47426-3. doi: 10.1007/978-3-030-47426-3_57.

[11] J. R. Landis and G. G. Koch. The Measurement of Observer Agreement for Categorical Data. *Biometrics*, 33(1):159–174, 1977. ISSN 0006-341X. doi: 10.2307/2529310. URL `https://www.jstor.org/stable/2529310`. Publisher: [Wiley, International Biometric Society].

[12] B. Levin and M. R. Hovav. A preliminary analysis of causative verbs in English. *Lingua*, 92:35–77, Apr. 1994. ISSN 0024-3841. doi: 10.1016/0024-3841(94)90337-9. URL `https://www.sciencedirect.com/science/article/pii/0024384194903379`.

[13] H. Li, D. Bollegala, Y. Matsuo, and M. Ishizuka. Using Graph Based Method to Improve Bootstrapping Relation Extraction. In A. Gelbukh, editor, *Computational Linguistics and Intelligent Text Processing*, Lecture Notes in Computer Science, pages 127–138, Berlin, Heidelberg, 2011. Springer. ISBN 978-3-642-19437-5. doi: 10.1007/978-3-642-19437-5_10.

[14] Z. Li, X. Ding, T. Liu, J. E. Hu, and B. Van Durme. Guided Generation of Cause and Effect. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, pages 3629–3636, Yokohama, Japan, July 2020. International Joint Conferences on Artificial Intelligence Organization. ISBN 978-0-9992411-6-5. doi: 10.24963/ijcai.2020/502. URL `https://www.ijcai.org/proceedings/2020/502`.

[15] Z. Luo, Y. Sha, K. Q. Zhu, S.-w. Hwang, and Z. Wang. Commonsense causal reasoning between short texts. In *Proceedings of the Fifteenth International Conference on Principles of Knowledge Representation and Reasoning*, KR'16, pages 421–430, Cape Town, South Africa, Apr. 2016. AAAI Press.

[16] G. A. Miller. WordNet: a lexical database for English. *Communications of the ACM*, 38(11):39–41, Nov. 1995. ISSN 0001-0782. doi: 10.1145/219717.219748. URL `https://doi.org/10.1145/219717.219748`.

[17] B. Rink, C. Bejan, and S. Harabagiu. Learning Textual Graph Patterns to Detect Causal Event Relations. Jan. 2010.

[18] S. Saha, H. Pal, and Mausam. Bootstrapping for Numerical Open IE. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 317–323, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-2050. URL `https://aclanthology.org/P17-2050`.

[19] S. Schuster and C. D. Manning. Enhanced English Universal Dependencies: An Improved Representation for Natural Language Understanding Tasks. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2371–2378, Portorož, Slovenia, May 2016. European Language Resources Association (ELRA). URL `https://aclanthology.org/L16-1376`.

[20] W. Stegmüller. Das Problem der Kausalität. In E. Topitsch, editor, *Probleme der Wissenschaftstheorie: Festschrift für Victor Kraft*, pages 171–190. Springer, Berlin, Heidelberg, 1960. ISBN 978-3-662-25138-6. doi: 10.1007/978-3-662-25138-6_6. URL `https://doi.org/10.1007/978-3-662-25138-6_6`.

[21] J. Yang, S. C. Han, and J. Poon. A Survey on Extraction of Causal Relations from Natural Language Text, Oct. 2021. URL `http://arxiv.org/abs/2101.06426`. Number: arXiv:2101.06426 arXiv:2101.06426 [cs].