Predicting Document Coverage for Relation Extraction

Sneha Singhania, Simon Razniewski, Gerhard Weikum
Max Planck Institute for Informatics, Germany
{ssinghan,srazniew,weikum}@mpi-inf.mpg.de

Abstract

This paper presents a new task of predicting the coverage of a text document for relation extraction (RE): Does the document contain many relational tuples for a given entity? Coverage predictions are useful in selecting the best documents for knowledge base construction with large input corpora. To study this problem, we present a dataset of 31,366 diverse documents for 520 entities. We analyze the correlation of document coverage with features like length, entity mention frequency, Alexa rank, language complexity, and information retrieval scores. Each of these features has only moderate predictive power. We employ methods combining features with statistical models like TF-IDF and language models like BERT. The model combining features and BERT, HERB, achieves an F1 score of up to 46%. We demonstrate the utility of coverage predictions on two use cases: KB construction and claim refutation.

1 Introduction

Motivation and Problem Relation extraction (RE) from text documents is an important NLP task with a range of downstream applications (Han et al., 2020). For these applications, it is vital to understand the quality of RE results. While extractors typically provide confidence (or precision) scores, this paper puts forward the notion of *RE coverage* (or recall). Given an input document and an RE method, coverage measures the fraction of the extracted relations compared to the complete ground-truth that holds in reality. We consider this on a per-subject and per-predicate basis—for example, all memberships of Bill Gates in organizations or all companies founded by Elon Musk.

Document coverage for RE highly varies. Consider the three text snippets about Tesla as shown in Figure 1. The first text contains all five founders of Tesla, while the second text contains only two of them, and the third has just one. Analogously, for the entity Tesla and the relation *founded-by*, we see that text 1 has coverage 1, text 2 has coverage 0.4, and text 3 has coverage 0.2.

When applying RE at scale, for example, to populate or augment a knowledge base (KB), an RE system may have to process a huge number of input documents that differ widely in their coverage. As state-of-the-art extractors are based on heavy-duty neural networks (Lin et al., 2016; Zhang et al., 2017; Soares et al., 2019; Yao et al., 2019), processing all documents in a large corpus may be prohibitive and inefficient. Instead, prioritizing the input documents by identifying the best documents with high coverage could be effective. This is why coverage prediction is crucial for large-scale RE, but the problem has not been explored so far.

This problem would be easy if we could first run a neural RE system on each document and then assess the yield, either by comparison to withheld labeled data or by sampling followed by human inspection. However, this is exactly the computational bottleneck that we must avoid. The challenge is to estimate document coverage, for a given entity and relation of interest, with inexpensive and lightweight techniques for document processing.

Approach and Contributions This paper presents the first systematic approach for analyzing and predicting *document coverage for relation extraction*. To facilitate extensive experimental study on this novel task, we introduce a large Document Coverage (DoCo) dataset of 31,366 web documents for 520 distinct entities spanning 8 relations, along with automated extractions and coverage labels. Table 1 provides samples from DoCo for each relation.

We employ a classifier architecture that we call HERB (for Heuristics with BERT), based on a document's lightweight features and additionally incorporates pretrained language models like BERT without any costly re-training and fine-tuning. The best configuration of this

Transactions of the Association for Computational Linguistics, vol. 10, pp. 207–223, 2022. https://doi.org/10.1162/tacl_a_00456
Action Editor: Hoifung Poon. Submission batch: 9/2021; Revision batch: 10/2021; Published 3/2022.

www.cnbc.com/2020/01/30/elon-musk-i-really-didnt-want-to-be-ceo-of-tesla.html

Text 1: … The company had five co-founders: Martin Eberhard and Marc Tarpenning, who started the original Tesla Motors in 2003, as well as Ian Wright, JB Straubel and Musk. …

www.rnz.co.nz/national/programmes/ninetonoon/audio/2017545 07/new-zealand-co-founder-of-tesla-motors-ian-wright

Text 2: … Ian Wright is a New Zealander engineer who co-founded Tesla Motors with Elon Musk in 2003. …

https://www.tesla.com/elon-musk

Text 3: … Elon Musk co-founded and leads Tesla, SpaceX, Neuralink and The Boring Company. As the co-founder and CEO of Tesla, Elon leads all product designs …

Figure 1: Sample documents from our Document Coverage (DoCo) dataset.

classifier achieves an F1-score of up to 46%. The classifier provides scores for its predictions and thus also supports ranking documents by their expected yield for the RE task at hand.

We evaluate our approach against a range of state-of-the-art baselines. Our results show that heuristic features like text length, entity mention frequency, language complexity, Alexa rank, or information retrieval scores have only moderate predictive power. However, in combination with pre-trained language models, the proposed classifier gives useful predictions of document coverage.

We further study the role of coverage prediction in two extrinsic use cases: *KB construction* and *claim refutation*. For KB construction, we show that coverage estimates by HERB are effective in ranking candidate documents and can substantially reduce the number of web pages one needs to process for building a reasonably complete KB. For the task of claim refutation (e.g., Tim Cook is the CEO of Microsoft), we show that coverage estimates for different documents can provide counter-evidence that can help to invalidate false statements obtained by RE systems.
The salient contributions of this work are:

1. We introduce the novel task of predicting document information coverage for RE.
2. To support experimental comparisons, we present a large dataset of annotated web documents.
3. We propose a set of heuristics for coverage estimation, analyze them in isolation and in combination with an inexpensive standard embedding-based document model.
4. We study the application of our classifier on two important use cases: KB construction and claim refutation. Experiments show that our predictor is useful in both of these tasks.

Our data, models and code is publicly available.[1]

2 Related Work

Relation Extraction (RE) RE is the task of identifying the relation types between two entities that are mentioned together in a sentence or in proximity within a document (e.g., in the same paragraph). RE has a long history in NLP research (Mintz et al., 2009; Riedel et al., 2010), with a recent overview given by Han et al. (2020). State-of-the-art methods are based on deep neural networks trained via distant supervision (Lin et al., 2016; Zhang et al., 2017; Soares et al., 2019; Yao et al., 2019). On the practical side, RE is available in several commercial APIs for information extraction from text. In our experiments, we make use of Rosette[2] and Diffbot.[3] Our approach is agnostic to the choice of extractors, though; any RE tool can be plugged in.

Knowledge Base Construction (KBC) RE plays a crucial part in the more comprehensive KBC task: identifying instances of entity pairs that stand in a given relation in order to construct a knowledge base (Mitchell et al., 2018; Weikum et al., 2021; Hogan et al., 2021).

The input is typically a set of documents, often assumed to be fixed and given upfront. This disregards the critical issue of benefit/cost trade-offs, which mandates identifying high-yield inputs for resource-bounded KBC. Identifying relevant, expressive, and preferable sources for KBC is often referred to as *source discovery*. Source discovery can be performed via IR-style ranking of documents or can be based on heuristic estimators of the yield of relation extractors (Wang et al., 2019; Razniewski et al., 2019). The former work, in

[1] www.mpi-inf.mpg.de/document-coverage-prediction.
[2] https://rosette.com/.
[3] https://www.diffbot.com/.

Entity	Relation	Content	Coverage
George W. Bush	*family*	President Bush grew up in Midland, Texas, as the eldest son of **Barbara and George H.W. Bush** . . . and met **Laura Welch**. They were married in 1977 . . . twin daughters: **Barbara**, married to **Craig Coyne**, and **Jenna**, married to **Henry Hager**. The Bushes also are the proud grandparents of **Margaret Laura ''Mila''**, **Poppy Louise**, and **Henry Harold ''Hal'' Hager** . . .	1
FedEx	*partner-org*	FedEx Corp. . . . to acquire **ShopRunner**, the e-commerce . . . acquires the International Express business of **Flying Cargo Group** . . . acquires **Manton Air-Sea Pty Ltd**, a leading provider . . . acquires **P2P Mailing Limited**, a leading . . . acquires **Northwest Research**, a leader in inventory . . . acquires **TNT Express** . . . acquires **GENCO** . . . acquires **Bongo International** . . . acquires the **Supaswift businesses** in South Africa . . . acquires **Rapidão Cometa** . . .	1
Warren Buffett	*member-of*	He formed **Buffett Partnership Ltd.** in 1956, and by 1965 he had assumed control of **Berkshire Hathaway** . . . Following Berkshire Hathaway's significant investment in **Coca-Cola**, Buffett became . . . director of **Citigroup Global Markets Holdings**, **Graham Holdings Company** and **The Gillette Company** . . .	0.8
Indra Nooyi	*edu-at*	Nooyi was born in Chennai, India, and moved to the US in 1978 when she entered the **Yale School of Management** . . . secured her B.S. from **Madras Christian College** and her M.B.A. from **Indian Institute of Management Calcutta** ...	0.75
J. K. Rowling	*position-held*	Rowling is one of the best-selling **authors** today . . . job of a **researcher** and bilingual **secretary** for Amnesty International . . . position of a **teacher** led to her relocating to Portugal ...	0.67
Apple Inc.	*founded-by*	**Steve Jobs**, the co-founder of Apple Computers . . . switched over to managing the Apple ''Macintosh'' project that was started . . .	0.33
Intel	*board-member*	**Andy D. Bryant** stepped down as chairman . . . **Dr. Omar Ishrak** to succeed . . . **Alyssa Henry** was elected to Intel's board. Her election marks the seventh new independent director . . .	0.125
3M	*ceo*	The American multinational conglomerate corporation 3M was formerly known as Minnesota Mining and Manufacturing Company. It's based in the suburbs . . .	0

Table 1: Sample entity-relation-document triples for all eight relations present in our DoCo dataset.

particular, approaches yield optimization as a set coverage maximization problem through shared properties of extracted entities. The latter uses textual features in a supervised SVM or LSTM model, a baseline with which we also compare in our experiments.

Document Ranking in IR Information retrieval (IR) ranks documents by relevance to a query with keywords or telegraphic phrases. Relevance judgments are based on the perception of informativeness concerning the query and its underlying user intent. Standard metrics for assessment, like precision, recall, and nDCG (Järvelin and Kekäläinen, 2002), are not applicable to our setting. The notion of coverage pursued in this paper refers to the yield of structured outputs by RE systems rather than document relevance. For example, a query-topic-wise highly relevant document that contains few extractable facts about named entities would still have low RE coverage.

Relevance of Coverage Estimates Understanding and incorporating document coverage prediction into NLP-based information extraction is essential for several reasons. For *resource-bounded KB construction*, it is crucial to know which documents are most promising for extraction with limited budgets for crawling and RE processing and/or human annotation (Ipeirotis et al., 2007; Wang et al., 2019). For *claim refutation*, coverage estimates can help to assess statements as questionable if documents with high coverage do not support them. So far, claim evaluation systems mostly rely on textual cues about factuality or source credibility (Nakashole and Mitchell, 2014; Rashkin et al., 2017; Thorne et al., 2018; Chen et al., 2019).

For *question answering* over knowledge bases, it is important to know whether a KB can be relied upon in terms of complete answer sets (Darari et al., 2013; Hopkinson et al., 2018; Arnaout et al., 2021). Current coverage estimation techniques for KBs do this analysis only post-hoc after the KB is fully constructed (Galárraga et al., 2017; Luggen et al., 2019), losing access to valuable information from extraction time.

3 Coverage Prediction

We take an entity-centric perspective, and view RE methods as functions mapping document-entity-relation tuples onto the set of objects found in the document. Formally, given a document d,

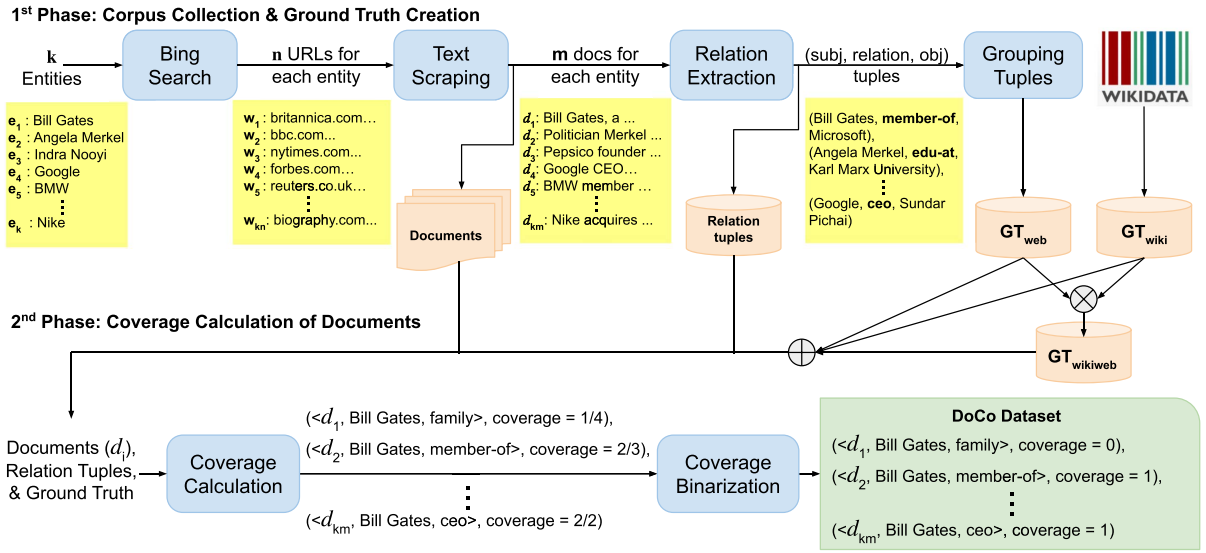

Figure 2: **Dataset Construction Pipeline.** There are two main phases: 1) corpus collection to create $\texttt{GT}_{\texttt{web}}$, and 2) coverage calculation. Phase 1 involves: i) for each entity e_i, n websites are collected using the Bing search API, ii) text is scraped from each website, iii) RE tuples from documents are extracted via Rosette/Diffbot, and iv) RE tuples are deduplicated and consolidated to form $\texttt{GT}_{\texttt{web}}$. The scraped documents are stored as inputs for phase 2 which consists of: i) for each document d_i, previously extracted relations are collected, and ii) based on the choice of GT, coverage is calculated to create the final DoCo dataset.

an entity e, a relation r, a ground truth GT of objects that stand in relation r with e, and a relation extraction method $extr$, the document coverage of d for (e, r) applying $extr$ is defined as:

$$\text{coverage}_{extr}(d, e, r) = \frac{\mid extr(d, e, r) \cap GT \mid}{\mid GT \mid} \quad (1)$$

The task thus takes the form of a classical prediction problem, either as a numerical coverage value or binarized class label. In Section 5, we propose several heuristics and methods that can be used to predict coverage for a given document. To study this novel problem, we require evaluation data. The following section thus deals with the generation of a large and diverse document coverage dataset.

4 Dataset Construction

A thorough study of document coverage prediction requires a corpus with two characteristics: (i) relation diversity (i.e., documents containing enough automatically extractable relations) and (ii) content diversity (i.e., multiple documents with varying content per entity). Existing text corpora, like the popular NYT (Sandhaus, 2008) and Newsroom dataset (Grusky et al., 2018), contain ample numbers of articles that mention newsworthy entities; however, the articles are primarily short, mentioning only very few relations. On the other end, machine-translated multilingual versions of Wikipedia articles (Roy et al., 2020) allow extraction of many relations but lack diversity.

For the novel task of predicting document information coverage, we thus built the DoCo (Document Coverage) dataset, consisting of 31,366 web documents for 520 distinct entities, each with its coverage value. Figure 2 illustrates the dataset construction.

Entity Selection First, well-known entities of two types, person (PER) and organization (ORG), were selected from popular ranking lists by Time 100[4] and Forbes[5,6] (‘‘Influential people around the globe’’, ‘‘Most valuable tech companies’’). These entities covered 12 diverse sub-domains, including politicians, entrepreneurs, singers, sports figures, writers, and actors, for PER, and technology, automobile, retail, conglomerate, pharmaceuticals, and financial corporations, for ORG. Popular and long-tail entities for PER, companies

[4] https://time.com/collection/100-most-influential-people-2020/.
[5] https://forbes.com/forbes-400/.
[6] https://forbes.com/lists/global2000/#9a993675ac04.

across demographics and with differing net worth for ORG, were chosen to further obtain documents with varying content.

Websites and Content We aimed to collect diverse 100 URLs per entity by issuing a set of search engine queries per entity, for example, ''about PER'', ''PER biography'', ''ORG history''. A total of 6 set of queries for PER and 10 for ORG was designed. Since the URLs returned over the set of queries were not always unique, we retained the duplicated URL only once.

Extracting textual content without noisy headers, menus, and comments required a labor-intensive scraping step. We handled the multi-domain content scraping task through a combination of libraries like Newspaper3k,[7] Readability,[8] and online scraping services like Import.io[9] and ParseHub.[10] We ensured high-quality scraped content by applying rule-based filters to remove noisy elements like embedded ADs and reference links. The scraped documents covered a range of website domains, including biographical sites, news articles, official company profiles, newsletters, and so on.

Relation Tuples Each document in DoCo was processed by two relation extraction APIs, Rosette and Diffbot. To annotate each document with coverage, we focused only on the entity queried initially to obtain the document. For our experimental study, we selected the following frequently occurring relations: *member-of*, *family*, *edu-at*, and *position-held*, for PER, and *partner-org*, *founded-by*, *ceo*, and *board-member*, for ORG. For more accurate coverage calculation, the RE tuples were deduplicated, for example, (Gates, *member-of*, Microsoft Corp.) would become (Bill Gates, *member-of*, Microsoft), via alignment to Wikidata identifiers returned by the APIs.

The relations extracted by the APIs are fine-grained like person-member-of, person-employee-of, org-acquired-by, and org-subsidiary-of. We combined the first two as member-of for PER and the last two as partner-org for ORG as coarse-grained relations.

[7] https://newspaper.readthedocs.io/en/latest/.

[8] https://pypi.org/project/readability-lxml/.

[9] https://www.import.io/.

[10] https://www.parsehub.com/.

Relation	Wikidata Property
member-of	member of (P463), member of political party (P102), part of (P361), employer (P108), owner of (P1830), record label (P264), member of sports team (P54)
family	father (P22), mother (P25), spouse (P26), child (P40), stepparent (P3448), sibling (P3373)
edu-at	educated at (P69)
position-held	position held (P39), occupation (P106)
partner-org	owner of (P1830), owned by (P127), member of (P463), parent organization (P749), subsidiary (P355)
founded-by	founded by (P112)
ceo	chief executive officer (P169)
board-member	board member (P3320)

Table 2: Wikidata property names and identifiers used to create $\texttt{GT}_{\texttt{wiki}}$.

Ground Truth We considered three ground-truth labels to calculate coverage for each document:

1. *Wikidata* ($\texttt{GT}_{\texttt{wiki}}$): A popular KB providing data for most relations yet having coverage limitations (Galárraga et al., 2017; Luggen et al., 2019). For example, for Bill Gates, Microsoft and other popularly associated companies for the *member-of* relation are present, but niche entities like Honeywell are missing. Depending on the entity type and sub-domain, we created the ground-truth labels by choosing those Wikidata properties that best matched the semantics of the 8 selected relations. Table 2 provides the complete information.

2. *Web Extractions* ($\texttt{GT}_{\texttt{web}}$): We used the set of frequent extractions across all the documents in DoCo as web-aggregated ground truth. For a given entity-relation (e, r), an extraction was determined frequent if it appeared in at least 5% of total documents corresponding to e, or if its count was no less than 5 times the highest counted tuple for (e, r). Deciding frequent extractions relative to total document count and other tuples' frequencies for an entity resulted in noise-free ground-truth labels.

3. *Wikidata and Web Extractions* ($\texttt{GT}_{\texttt{wikiweb}}$): We merged both previous variants using set

# PER entities	250
# ORG entities	270
# Relations	8
# Documents	31,366
Doc. length range (words)	[20, 10906]
# Unique website domains	600
# Doc. with non-zero RE tuples	26956
# Doc. with non-zero coverage	14086
# Doc. in class informative	7103 (22.6 %)

Table 3: Characteristics of the DoCo dataset.

union operation and phrase embeddings with cosine similarity for higher recall.

Coverage Calculation Coverage was computed on a per entity-relation-document basis using Equation (1). Even though real-valued coverage values are computed while constructing the dataset, it is often not possible to give nuanced predictions at test time. Consider the text ''. . . Musk is a co-founder of Tesla ...''. The term *co-founder* clearly indicates the presence of multiple founders; however, the context does not provide any clue on the total number of co-founders. For example, there could be one other co-founder (coverage 0.5) or 9 other co-founders (coverage 0.1).

Coverage Binarization We binarized the coverage values to circumvent this problem, splitting documents into two classes: *informative* and *uninformative*. The binarization method comprised an absolute and a relative threshold: A document was labeled as informative or 1 if its coverage was greater than 0.5, or greater than the coverage of at least 85% of documents for the same (e, r); otherwise, it was labeled as uninformative or 0.

Dataset Characteristics After filtering duplicates, irrelevant URLs like social media handles, and video-content websites, we obtained a total of 31,366 documents for 520 entities. Table 3 provides an overview of the DoCo dataset. We can see that DoCo's labels are imbalanced, as only 22.6% of the documents are informative and 77.4% are uninformative. The count of documents with non-zero RE tuples is higher than those with non-zero coverage since the RE tuples were not always related to the subject entity, hence irrelevant towards coverage calculation.

Table 4 gives the average number of objects present in each ground truth variant. On average

Relation	$\texttt{GT}_{\texttt{wiki}}$	$\texttt{GT}_{\texttt{web}}$	$\texttt{GT}_{\texttt{wikiweb}}$
member-of	3.61	6.51	7.12
family	2.21	4.0	4.41
edu-at	2.26	2.07	2.58
position-held	5.86	7.76	10.37
partner-org	6.16	4.26	3.12
founded-by	1.07	1.06	1.66
ceo	1.03	2.77	2.86
board-member	0.47	1.44	1.75

Table 4: Average number of objects per entity.

Relation	Human	Diffbot	Rosette	$\texttt{GT}_{\texttt{wiki}}$	$\texttt{GT}_{\texttt{web}}$	$\texttt{GT}_{\texttt{wikiweb}}$
member-of	4.36	3.66	**5.04**	4.54	7.12	9.22
family	4.74	**3.82**	0.66	1.76	5.78	6.64
edu-at	1.72	2.5	**2.52**	2.94	3.08	2.18
position-held	2.9	**4.26**	–	6.7	6.14	9.52
partner-org	3.7	0.72	**2.26**	0.8	5.04	5.92
founded-by	1.34	0.58	**1.8**	0.78	2.84	2.96
ceo	2.02	**1.96**	–	1.68	4.32	4.2
board-member	2.62	**1.54**	–	2.82	3.48	2.64

Table 5: Average tuple count per relation. The RE tool with higher tuple count (boldfaced) is chosen for each relation.

across relations, the number of objects in $\texttt{GT}_{\texttt{web}}$ is higher than those in $\texttt{GT}_{\texttt{wiki}}$ by 23.7%, and $\texttt{GT}_{\texttt{wikiweb}}$ is higher than those in $\texttt{GT}_{\texttt{wiki}}$ by 28.8%. This implies that $\texttt{GT}_{\texttt{web}}$ and $\texttt{GT}_{\texttt{wiki}}$ can have overlapping objects, and $\texttt{GT}_{\texttt{web}}$ might contain extra objects towards $\texttt{GT}_{\texttt{wikiweb}}$ creation.

Dataset Quality We analyzed the quality of the DoCo dataset by comparing automatic relation extractions to extractions given by human annotators. A sample of 400 documents was selected, 50 per relation, with half from the high-coverage range and the rest from the low-coverage range. Each document was annotated with all correct tuples for the document's main subject entity.

Table 5 shows the observed averaged counts. We note that the human annotators extracted a substantial number of tuples for all 8 relations, indicating the richness and breadth of the DoCo documents. The two automatic extractors mostly yielded smaller numbers of tuples, with a few exceptions. These exceptions include spurious tuples, though. The ground-truth variants consistently suggest higher numbers, but except for the conservative $\texttt{GT}_{\texttt{wiki}}$, these are usually overestimates due to spurious tuples. The GT variants

should thus be seen as upper bounds for the true RE coverage.

We analyzed how well the automatic annotations reflect human annotations' coverage by computing Pearson correlation coefficients for the entire set of 400 sample documents. For a relation, the RE tool with higher averaged count was chosen for our experiments, and the correlation for (Human, RE) is 0.68. This shows that optimizing for coverage by automatic RE tools is highly correlated with the overarching goal of approximating human-quality outputs.

5 Approach

We aim to model coverage prediction by processing unstructured document text by inexpensive lightweight techniques. This is crucial for identifying promising documents before embarking on heavy-duty RE.

Heuristics We devise several simple heuristics involving textual features for document coverage.

1. *Document Length*: The length of a document is a proxy for the amount of information contained. Longer documents may express more relations.

2. *NER Frequency*: Length can be misleading when a document is verbose, yet uninformative. The count of named-entity mentions matching the relation domain (e.g., persons for the relation *family*, or organizations for the relation *member-of*) could correlate with coverage.

3. *Entity Saliency*: The more frequently an entity is mentioned in a document, the more likely the document expressed relations for that entity.

4. *IR-Relevance Signals*: The surface similarity of the entire document with the input query is another cue. We adopt BM25 (Robertson et al., 1995), a classical and still powerful IR model for ranking documents, using $\langle e \rangle + \langle r \rangle$ as query, where e and r are the target entity and relation, respectively. Recent advances on neural rankers are considered as well (Nogueira and Cho, 2020). We follow Nogueira et al. (2020) and use the T5 sequence-to-sequence model (Raffel et al., 2020) to rank documents.

5. *Website Popularity*: Popular websites may be visited often because they are more informative. We use the standard Alexa rank[11] as a measure of popularity.

6. *Text Complexity*: RE methods are effective on simpler text, and may not be able to effectively extract relations from documents written in complex prose. We use the Flesch score (Flesch and Gould, 1949), a popular text readability measure.

7. *Random*: We contrast the predictive power of our proposed methods with two random baselines: A fair coin, and a biased coin maintaining the label imbalance in our test set.

Methods We use several inexpensive statistical models for document representation and feed them to a logistic regression classifier.

8. *Latent Topic Modeling*: Topics in a document could be a useful indicator of coverage. For example, for relation *family*, latent topics like *ancestry* or *personal life* are relevant. We use Latent Dirichlet Allocation (LDA) (Blei et al., 2003) to model documents as distributional vectors.

9. *BOW+TFIDF*: A simple yet effective statistic to measure word importance given a document in a corpus is the product of term frequency and inverse document frequency (TF-IDF). We vectorize a document into a Bag-of-Words (BOW) representation with TF-IDF weights.

10. *Ngrams+TFIDF*: A document is vectorized using frequent n-grams ($n \leq 3$) with TF-IDF weights.

We employ two neural baselines including LSTM and pre-trained language model (BERT).

11. *LSTM*: Previous work by Razniewski et al. (2019) used textual features to estimate the presence of a complete set of objects in a text segment. We adopt their architecture, representing documents using 100 dimensional

[11] `www.alexa.com`.

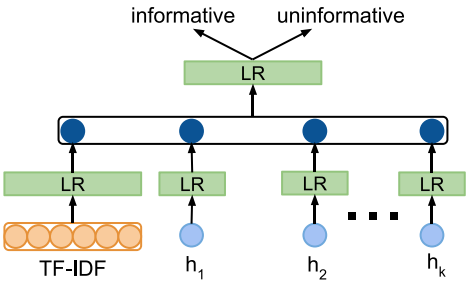

Figure 3: Architecture for Heuristics combined with TF-IDF (Heu+TFIDF).

GloVe embeddings (Pennington et al., 2014), and processing them in LSTM (Hochreiter and Schmidhuber, 1997), followed by a feed-forward layer with ReLU activation before the classifier.

12. *Language Model (BERT)*: Without costly re-training or fine-tuning, we utilize pre-trained BERT embeddings (Devlin et al., 2019) in a feature-based approach by extracting activations from the last four hidden layers. As in the original work, these contextual embeddings are fed to a two-layer 768-dimensional BiLSTM before the classifier.

Our experiments (Sec. 6.2) reveal that each of our proposed heuristics has only a moderate predictive power. We therefore formulate a lightweight classifier to combine heuristics with the best performing statistical model (TF-IDF), or language model (BERT).

13. *Heuristics with BOW+TFIDF* (Heu+TFIDF): We combine TF-IDF with heuristics (one to six) using stacked Logistic Regression (LR) (Figure 3). In level 1, the TF-IDF vector and each individual heuristic are fed to separate LR classifiers. In level 2, all the outputs of level 1 LRs are concatenated and fed to a final LR classifier for coverage prediction. The entire model is jointly trained.

14. ***Heuristics with BERT*** (HERB): We combine BERT with heuristics (one to six) in a two-step process (Figure 4). In the first step, we reuse the BERT model above (with no additional training or fine-tuning) for coverage prediction. This prediction is then concatenated with heuristics to form a single vector, which is fed to a LR classifier.

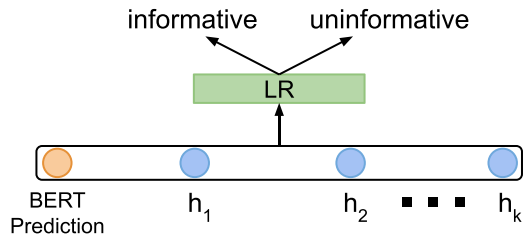

Figure 4: Architecture for Heuristics combined with BERT Prediction (HERB).

6 Experiments

6.1 Setup

Dataset We considered two automatic RE tools, Rosette and Diffbot, *extr* (Rosette, Diffbot), and three ground truth variants: $\texttt{GT}_{\texttt{wiki}}$, $\texttt{GT}_{\texttt{web}}$, $\texttt{GT}_{\texttt{wikiweb}}$. For each relation, we report on the combination of RE tool and GT variant that achieves the highest count of documents classified as high-coverage.

Each relation had a separate labeled set of documents, split into 70% train, 10% validation and 20% test. Information leakage was prevented by splitting along entities, i.e., all documents on the same entity would exclusively be in one of train, validation or test set. The number of training samples per relation varies from 664 (*board-member*) to 3604 (*position-held*). Since the label distribution in DoCo is imbalanced, the uninformative (or 0) class in all train datasets were undersampled to obtain a 50:50 distribution, while the validation and test datasets were kept unchanged to reflect the real-world imbalance. Named entities and numbers were masked.

Models Each proposed heuristic was turned into a classifier by first ranking documents according to the heuristic, and then labeling the top 50% documents as class 1 or informative. We used the Okapi BM25[12] and monoT5[13] open-source implementations for IR ranking. The monoT5 model is generally used for passage ranking, and as DoCo documents are much longer with multiple passages, we used the MaxP algorithm (Dai and Callan, 2019) to compute the document ranking. Since the difference in performance between T5 and BM25 models is negligible, we chose the simpler yet equally effective BM25 model as IR-relevance signal for HERB.

[12] https://pypi.org/project/rank-bm25/.
[13] https://github.com/castorini/pygaggle.

Method	PER				ORG				Avg.
	member-of	*family*	*edu-at*	*position-held*	*partner-org*	*founded-by*	*ceo*	*board-member*	
Random (biased)	5.7	6.8	4.9	10.0	7.5	1.2	13.5	3.7	6.6
Random (fair)	15.7	11.1	12.6	15.4	15.2	8.9	21.3	7.2	13.4
Text Complexity	9.6	5.4	6.1	10.3	3.5	3.3	15	5.4	7.3
Alexa Ranking	12.6	9.8	8.1	12.4	16.7	11.3	24.8	7.3	12.9
Entity Saliency	17.8	14.3	11.9	18.2	14.7	8.4	24.6	7.1	14.6
Document Length	20.5	19.0	15.5	21.9	23.9	12.8	28.8	8.5	18.9
NER Count	24.3	19.8	18.2	–	21.1	13.7	34.5	11.8	20.5
BM25 IR	27.1	21.1	18.8	26.3	21.8	12.9	36.6	12.1	22.1
T5 IR	26.9	23.2	20.3	29.6	19.5	15.4	41.1	13.1	23.6
LDA Topic Model	19.3	19.0	14.5	21.1	15.7	8.6	25.2	11.5	16.9
GloVe+LSTM	16.5	28.6	19.8	32.9	24.2	19.5	24.4	4.9	21.3
Ngrams+TFIDF	36.2	40.0	25.6	40.2	18.6	25.5	41.8	30.2	32.3
BOW+TFIDF	36.0	41.0	29.2	42.1	17.2	28.3	40.6	32.1	33.3
BERT	40.4	39.7	35.7	44.4	22.0	30.8	43.0	33.8	36.2
Heu+TFIDF	41.9	**43.5**	31.3	36.5	**35.1**	28.2	41.4	22.0	35.0
HERB	**44.2**	41.7	**40.5**	**45.6**	28.8	**32.5**	**46.2**	**34.8**	**39.3**

Table 6: F1-scores (%) obtained on the coverage prediction task by various heuristics and methods.

Feature based methods including topic modeling with LDA, TF-IDF, and n-grams, were fed to a Logistic Regression classifier. In the LSTM architecture, we used 100 dimensional GloVe embeddings with a vocabulary size of 100,000, and a 100 dimensional hidden state for LSTM.

For pre-trained language models, we used the BERT-base-uncased[14] model (without additional retraining or fine-tuning) to encode sentences, by summing the [CLS] token's representation from the last four hidden layers. Input documents were padded or truncated to 650 sentences, and represented through sentence encodings. Coverage classification was performed using the feature-based approach outlined in Devlin et al. (2019).

We constructed mini-batches of size 32, used the Adam optimizer initialized with a constant learning rate of $1e$-05 and $1e$-09 epsilon value, and trained for 200 epochs. Because our dataset is imbalanced, we monitored validation precision to save the best model, and report optimal F1-scores (Lipton et al., 2014) to compare results.

6.2 Results

Our results are shown in Table 6. Each heuristic gives a mediocre performance, with T5 IR achieving the highest average F1 of 23.6 among the heuristics. In the trained group of models, LDA has the lowest average F1 of 16.9, while BERT performs the best with an average F1 of 36.2.

[14] https://huggingface.co/bert-base-uncased.

Although each heuristic has moderate predictive power, combining them with statistical models like TF-IDF, or pre-trained language models like BERT, gives the best performance. Among the combination models, HERB outperforms Heu+TFIDF in a clear majority of relations.

Model Analysis Statistical models like BOW+TFIDF and Ngrams+TFIDF performed comparably to BERT for a minority of relations. To better understand these models, we analyzed highly positive and negative features. Table 7 provides noteworthy examples. We observe the presence of semantically relevant phrases. We also inspect the weights of the trained LR classifier of HERB. Across relations, BERT had the highest average weight (5.05), followed by BM25 (2.56), while NER Count had the lowest weight (0.07).

Feature Ablations We further perform an ablation analysis, with Table 8 showing the average F1-scores when individual heuristics are removed from HERB. Removing either BM25 or Text Complexity leads to a significant drop in performance, indicating that other heuristics or BERT do not capture these features well.

Human Performance Finally, we compare the results against human performance on identifying high-coverage documents. For each relation, 10 randomly sampled test documents were labeled as informative or uninformative for RE solely by reading the document. Averaged over all relations,

Relation	Important Phrases
member-of	[org], is part of, ambassador, is associated with, [org] partner
family	[person], married, father, wife, children, daughter, parents, [number]
edu-at	[org], graduated, degree, studied, [org] in [number], is part of
position-held	[person], leader, president, actor, professor, writer, founder, police, portman
partner-org	[org], [number] [org], subsidiary, merger, the company, member of
founded-by	[person], founder, director, executive, chairman, co founder, head of, chief executive
ceo	ceo, [person] director, chief, officer, founders, chief executive officer, president
board-member	[org], [person], chairman, executive, board of directors, [number] senior executive, officer in charge, representative director

Table 7: Highly weighted phrases given by the trained LR classifier of Ngrams+TFIDF and BOW+TFIDF.

HERB	39.3%
- Doc. Length	36.8% (–2.44)
- Entity Saliency	36.4% (–2.85)
- Alexa Ranking	36.3% (–3.03)
- NER Count	36.2% (–3.11)
- BM25	36.0% (–3.29)
- Text Complexity	35.7% (–3.62)

Table 8: Average F1 performance with feature ablations. Text Complexity and BM25 are most important.

humans obtained an F1 score of 70.42%, compared with HERB predictions reaching an average F1 of 39.3%, and all baselines were significantly inferior. The large gap between humans and learned predictors shows the hardness of the coverage prediction task and underlines the need for the presented research.

7 Analysis and Discussion

Domain Dependency To investigate how strongly prediction depends on in-domain training data, we performed a stress test, where the train, validation, and test sets were split along domains (e.g., singers vs. entrepreneurs vs. politicians). Table 9 shows the resulting F1-scores (%). For HERB, the average F1-score on the in-domain test set is 34.3%, while on the out-of-domain test set is 34.2%—that is, there is no notable drop for the challenging domain-transfer case. We observe a minor drop for larger relations, while even increases are visible for the smallest two relations. This suggests that HERB learned generalizable features that are beneficial across domains.

Evaluation of Document Ranking So far, we have evaluated our methods on a binary prediction problem. However, use cases frequently require a ranking capability (see also Sec. 8). We additionally evaluate our methods on a ranking task, where documents are ranked by the score of positive predictions.

We use the mean Normalized Discounted Cumulative Gain (mean nDCG) (Järvelin and Kekäläinen, 2002) as the evaluation metric. A similar performance trend to the F1 metric is observed among our methods. HERB performs the best with an average nDCG score of 0.45 across relations, while BERT and Heu+TFIDF have 0.44 and 0.43, respectively.

RE Limitations The performance of RE methods significantly impacts the quality of $\texttt{GT}_{\texttt{Web}}$ as well as the RE coverage of documents. Although we used state-of-the-art commercial APIs, these nonetheless struggle on open web documents. To illustrate this, we randomly sampled 40 documents from DoCo and compared the count of RE tuples returned by Diffbot/Rosette against the count by a human relation extractor. Diffbot returned 60.6% fewer relational tuples, and Rosette returned 72.3% fewer, suggesting the need for further improvement of RE methods.

Error Analysis We analyzed the incorrect predictions by HERB and categorized the errors. For each relation, we randomly sampled 10 incorrectly predicted documents, 5 false positives and 5 false negatives. Out of the total 80 samples, 63.75% of documents contained partial information for the chosen relation; on 15% of documents the IE methods failed to extract all the necessary RE

Setting	*member-of*	*family*	*edu-at*	*position-held*	*partner-org*	*founded-by*	*ceo*	*board-member*	Avg.
HERB (in-domain)	40.8	41.8	34.9	42.8	28.4	17.1	45.4	23.3	34.3
HERB (out-of-domain)	35.7	39.7	32.5	39.1	29.4	23.8	42.3	31.1	34.2
Training Data Size	2194	1650	1458	2940	1124	828	2058	608	

Table 9: Comparison of F1-scores (%) of HERB on the in-domain and out-of-domain test set.

tuples; the ground truth for 3.75% of documents had an incomplete set of objects; 3.75% documents had noisy content; and 2.5% documents had incomplete information due to failure of scraping methods on complex website layouts.

Multiple documents in the low-information category contained speculative content—for example, considerations about candidates for a new appointment as a board member or CEO. In other cases, the document would mention the increased count of board members, but not their names. A few documents also had partial information leading to false positives—for example, a document partially talking about the footballer Sergio Agüero for the *family* relation was incorrectly classified as informative; as it also contained a complete family history about another footballer, Diego Maradona (Sergio's father-in-law).

Conversely, documents may contain information relevant to a relation without actual mention of the relation, which leads to false negatives. For example, a document on the LinkedIn Corporation stating '' . . . Weiner stepped down from LinkedIn . . . He named Ryan Roslansky as his replacement.'' was labeled uninformative for the *ceo* relation. Although Ryan Roslansky and LinkedIn are related through the *ceo* relation, the implicit statement was not noticed by HERB.

We specifically inspected the IR baselines' performance to understand better why these are mediocre predictors at best. The IR signals about entire documents merely reflect that a document is on the proper topic given by the query entity, but that does not necessarily imply that the document contains *many* relational facts about the target entity. For RE coverage, IR-style document-query relevance is a necessary cue but not a sufficient criterion.

Efficiency and Scalability We measured the run-time of HERB against a state-of-the-art neural model for document-level RE (DocRED) (Yao et al., 2019). Based on the DocRED leaderboard,[15] we selected the currently best open-source method: the Transformer-based Structured Self-Attention Network (SSAN) (Xu et al., 2021).

A sample of 100 documents from DoCo was given to both HERB and SSAN and processed as follows. For HERB, features are computed utilizing BERT, followed by coverage prediction. For SSAN, documents first need to be pre-processed to construct the necessary DocRED representation. This includes named entity recognition and pair-wise co-reference resolution, using Stanza[16] to properly group same-entity occurrences.

The measurements show the following. HERB takes about 2 seconds, on average, to process one document, whereas SSAN requires 13.6 seconds—a factor of 6.8 higher in speed and resource consumption. The difference becomes even more prominent for very long documents with many named entity mentions. HERB's run-time grows linearly with document length, while SSAN's run-time exhibits quadratic growth with the number of entity mentions.

This quadratic complexity of full-fledged neural RE has inherent reasons (as stated in Yao et al., 2019). Document-level relation extraction generally requires computations for all possible pairs of entity mentions. The neural RE methods need to have the positions of candidate entity pairs as input, which necessitates considering all pairs of mentions.

8 Applications

To demonstrate the importance of coverage prediction, we evaluated its utility in two use cases, knowledge base construction and claim refutation. For the former, we discuss the importance of ranking documents by RE coverage (Section 8.1)

[15] https://competitions.codalab.org/competitions/20717#results.

[16] https://stanfordnlp.github.io/stanza/.

and a practically relevant setting where RE is constrained by resource budgets (Section 8.2).

8.1 Document Ranking for Relation Extraction

Relation extraction plays a pivotal role in KB construction. We show the relevance of coverage estimates for prioritizing among documents. Entities from our test dataset serve as subjects for RE. We select top k documents from the test dataset corpus by four different techniques. We compare the performance of each method by the total number of extracted RE tuples per subject and compute recall w.r.t. the Wikidata ground-truth.

1. *Random*: A random sample of documents.
2. *IR-Relevance*: Using BM25 to identify the most relevant documents.
3. *Coverage Prediction*: HERB's predictions to rank documents.
4. *Coverage Oracle*: Selecting documents by their ground-truth labels from DoCo. This ranking gives an upper bound on what an ideal method could achieve.

Setup The document coverage calculation is on a per (e, r) pair basis. In a single iteration, all the proposed methods are given a set of documents partitioned by (e, r) pairs. Each method uses its technique to rank the documents, and the top k ranked documents are given to the RE API (Rosette or Diffbot) for obtaining the set of relational tuples.

Results Figure 5 (*top*) compares the total RE tuples obtained by the proposed methods, averaged across test dataset entities and 8 chosen relations. Notably, BM25 doesn't perform much better than random, while coverage prediction is not far behind the perfect ranking defined by the coverage oracle. Ordering documents by coverage prediction instead of IR-relevance gives 50% more extractions from the top-10 documents.

Figure 5 (*bottom*) shows the number of RE tuples that match the Wikidata KB, thus comparing the methods on precision. As was foreseeable, the coverage oracle method wins due to the usage of correct coverage values for ranking. HERB's coverage prediction performance is considerably

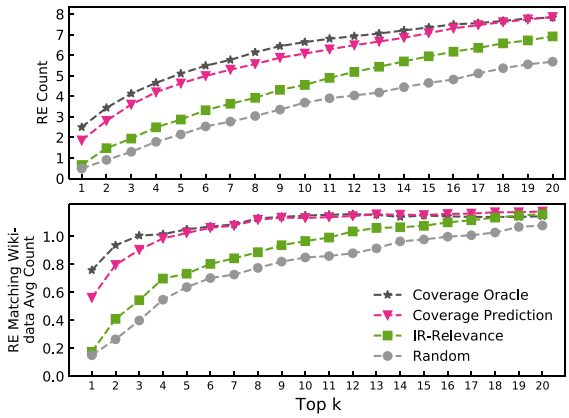

Figure 5: Total yield (top) and precision (bottom) of KBC based on different ranking methods for documents.

Method	RE Count	#Docs Processed
SSAN	59	410
HERB+SSAN	96	318

Table 10: Relation extraction under run-time constraint.

higher than IR-relevance and other methods, while it matches the coverage oracle for $K \geq 4$. Beyond $K > 15$, all methods yield nearly the same sets of tuples, hence similar precision.

8.2 Budget-constrained Relation Extraction

Document coverage predictions are particularly important for massive-scale RE tasks targeted at long-tail entities, such as populating or augmenting a domain-specific knowledge base (e.g., about diabetes or jazz music). Such tasks may require screening a huge number of documents. Therefore, practically viable RE methods need to operate under budget constraints, regarding the monetary cost of computational resources (e.g., using and paying for cloud servers) as well as the cost of energy consumption and environmental impact.

In the experiment described here, we simulate this setting, comparing standard RE by SSAN against HERB-enhanced RE where HERB prioritizes documents for RE by SSAN. We assume a budget of 10 minutes of processing time and give both methods 100 candidate documents. SSAN selects documents randomly and processes them until it runs out of time. HERB+SSAN sorts documents by HERB scores for high coverage and then lets SSAN process them in this order. The time

Subject	Relation	Object	Document Snippet
Alphabet Inc.	*ceo*	Susan Wojcicki	Susan Wojcicki is CEO of Alphabet subsidiary YouTube, which has 2 billion monthly users.
Oracle Corporation	*founded-by*	David Agus	Oracle Co-founder Larry Ellison and acclaimed physician and scientist Dr. David Agus formed Sensei Holdings, Inc.
PepsiCo	*board-member*	Joan Crawford	Film actress Joan Crawford, after marrying Pepsi-Cola president Alfred N. Steele became a spokesperson for Pepsi.

Table 11: Incorrect claims extracted by Diffbot RE API from documents predicted as low coverage.

for HERB itself is part of the 10-minute budget for the HERB+SSAN method.

As a proof-of-concept, we ran this experiment for a sample of 10 different entities (each with a pool of 100 documents).

Table 10 shows the results. Due to the upfront cost of HERB, HERB+SSAN processes fewer documents within the 10-minute budget, but its yield is substantially higher than that of SSAN alone, by a factor of 1.63. This demonstrates the need for document-coverage prediction towards realistic usage.

8.3 Claim Refutation

Our second use case is fact-checking, specifically the case of refuting false claims by providing counter-evidence via RE.

Reasoning *Extraction confidence* and *document coverage* are conceptually independent notions. However, when looking at sets of documents, an interesting relation emerges. Consider two documents, d_1 with high coverage, and d_2 with low coverage, along with two claims c_1 and c_2 from the respective documents, extracted with the same confidence. *Can we use coverage information to make claims about extraction correctness?*

We propose the following hypothesis: Given that d_1 is asserted to have high coverage, we can conclude that any statement not mentioned in d_1 (like c_2) is more likely false. In contrast, the low coverage of d_2 implies that d_2 is unlikely to contain all factual statements. Thus, c_1 not being found in d_2 is no indication that it could not be true.

Validation We experimentally validated the correctness of the above reasoning as follows. From the collection of relation extractions from the test dataset documents, we randomly sampled 69 pairs of claims for the same entity and relation, which had low support (i.e., extraction found only in one website). We then ordered the pairs by the coverage of the documents that did not express them, obtaining 69 claims with relatively higher coverage in non-expressing documents and 69 claims with relatively lower coverage.

We manually verified the correctness of each claim on the Internet, verifying annotator agreement on a sub-sample, where we found a high Fleiss' Kappa (Fleiss, 1971) inter-annotator agreement of 0.82.

Using these annotations, we found that from the 69 claims absent from lower-coverage documents, 58% (40) were correct, while from those absent from higher-coverage documents, only 36% (25) were correct. In other words, the fraction of correct claims absent from low-coverage documents is 1.6 times higher; so coverage can be used as a feature for claim refutation.

Table 11 shows examples of claims absent from high-coverage documents.

9 Conclusion

This paper introduces the new task of document coverage prediction and a large dataset for experimental study of the task. Our methods show that heuristic features can boost the performance of pre-trained language models without costly fine-tuning. Moreover, we demonstrate the value of coverage estimates for the use cases of knowledge base construction and claim refutation. Our future research includes developing a user-friendly tool to support knowledge engineers.

Acknowledgments

We thank Andrew Yates for his suggestions. Further thanks to the anonymous reviewers, action editor, and fellow researchers at MPI, for their comments towards improving our paper. This work is supported by the German Science Foundation (DFG: Deutsche Forschungsgemeinschaft) by grant 4530095897: ''Negative Knowledge at Web Scale''.

References

Hiba Arnaout, Simon Razniewski, Gerhard Weikum, and Jeff Z. Pan. 2021. Negative statements considered useful. *Journal of Web Semantics*, 71:100661. https://doi.org/10.1016/j.websem.2021.100661

David M. Blei, Andrew Ng, and Michael I. Jordan. 2003. Latent dirichl et allocation. *Journal of Machine Learning Research*, 3:993–1022.

Sihao Chen, Daniel Khashabi, Wenpeng Yin, Chris Callison-Burch, and Dan Roth. 2019. Seeing things from a different angle: Discovering diverse perspectives about claims. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 542–557, Minneapolis, Minnesota. Association for Computational Linguistics.

Zhuyun Dai and Jamie Callan. 2019. Deeper text understanding for IR with contextual neural language modeling. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 985–988, New York, NY, USA. Association for Computing Machinery. https://doi.org/10.1145/3331184.3331303

Fariz Darari, Werner Nutt, Giuseppe Pirrò, and Simon Razniewski. 2013. Completeness statements about RDF data sources and their use for query answering. In *The Semantic Web – ISWC 2013*, pages 66–83. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-41335-3_5

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Joseph L. Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378–382.

Rudolf Flesch and Alan J. Gould. 1949. *The Art of Readable Writing*, volume 8. Harper New York. https://doi.org/10.1037/h0031619

Luis Galárraga, Simon Razniewski, Antoine Amarilli, and Fabian M. Suchanek. 2017. Predicting completeness in knowledge bases. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, pages 375–383, New York, NY, USA. Association for Computing Machinery. https://doi.org/10.1145/3018661.3018739

Max Grusky, Mor Naaman, and Yoav Artzi. 2018. Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 708–719, New Orleans, Louisiana. Association for Computational Linguistics. https://doi.org/10.18653/v1/N18-1065

Xu Han, Tianyu Gao, Yankai Lin, H. Peng, Yaoliang Yang, Chaojun Xiao, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. 2020. More data, more relations, more context and more openness: A review and outlook for relation extraction. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 745–758, Suzhou, China. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735, PubMed: 9377276

Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia d'Amato, Gerard de Melo, Claudia Gutiérrez, Sabrina Kirrane, José Emilio Labra Gayo, Roberto Navigli, Sebastian Neumaier, Axel-Cyrille N. Ngomo, Axel Polleres, Sabbir M. Rashid, Anisa Rula, Lukas Schmelzeisen, Juan Sequeda, Steffen Staab, and Antoine Zimmermann. 2021. Knowledge graphs. *ACM Computing Surveys*, 64(4):96–104. https://doi.org/10.1145/3447772

Andrew Hopkinson, Amit Gurdasani, Dave Palfrey, and Arpit Mittal. 2018. Demand-weighted completeness prediction for a knowledge base. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 3 (Industry Papers)*, pages 200–207, New Orleans, Louisiana. Association for Computational Linguistics. https://doi.org/10.18653/v1/N18-3025

Panagiotis G. Ipeirotis, Eugene Agichtein, Pranay Jain, and Luis Gravano. 2007. Towards a query optimizer for text-centric tasks. *ACM Transactions on Database Systems*, 32(4):21–es.

Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of IR techniques. *ACM Transactions of Information Systems*, 20(4):422–446. https://doi.org/10.1145/582415.582418

Yankai Lin, Shiqi Shen, Zhiyuan Liu, Huanbo Luan, and Maosong Sun. 2016. Neural relation extraction with selective attention over instances. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2124–2133, Berlin, Germany. Association for Computational Linguistics.

Zachary C. Lipton, Charles Elkan, and Balakrishnan Narayanaswamy. 2014. Optimal thresholding of classifiers to maximize F1 measure. In *Machine Learning and Knowledge Discovery in Databases*, pages 225–239, Berlin, Heidelberg. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-662-44851-9_15, PubMed: 26023687

Michael Luggen, Djellel Difallah, Cristina Sarasua, Gianluca Demartini, and Philippe Cudré-Mauroux. 2019. Non-parametric class completeness estimators for collaborative knowledge graphs—the case of wikidata. In *The Semantic Web – ISWC 2019*, pages 453–469, Cham. Springer International Publishing. https://doi.org/10.1007/978-3-030-30793-6_26

Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011, Suntec, Singapore. Association for Computational Linguistics. https://doi.org/10.3115/1690219.1690287

Tom M. Mitchell, William W. Cohen, Estevam R. Hruschka, Partha P. Talukdar, Bo Yang, J. Betteridge, Andrew Carlson, Bhavana Dalvi, Matt Gardner, Bryan Kisiel, Jayant Krishnamurthy, Ni Lao, Kathryn Mazaitis, Thahir Mohamed, Ndapandula Nakashole, Emmanouil Antonios Platanios, Alan Ritter, Mehdi Samadi, Burr Settles, Richard C. Wang, Derry T. Wijaya, Abhinav Gupta, Xinlei Chen, Abulhair Saparov, Malcolm Greaves, and Joel Welling. 2018. Never-ending learning. *Communications of the ACM*, 61(5):103–115. https://doi.org/10.1145/3191513

Ndapandula Nakashole and Tom M. Mitchell. 2014. Language-aware truth assessment of fact candidates. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1009–1019, Baltimore, Maryland. Association for Computational Linguistics. https://doi.org/10.3115/v1/P14-1095

Rodrigo Nogueira and Kyunghyun Cho. 2020. Passage re-ranking with BERT. *ArXiv*, abs/1901.04085.

Rodrigo Nogueira, Zhiying Jiang, Ronak Pradeep, and Jimmy Lin. 2020. Document ranking with a pretrained sequence-to-sequence model. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 708–718, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.findings-emnlp.63

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics. https://doi.org/10.3115/v1/D14-1162

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Hannah Rashkin, Eunsol Choi, Jin Y. Jang, Svitlana Volkova, and Yejin Choi. 2017. Truth of varying shades: Analyzing language in fake news and political fact-checking. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2931–2937, Copenhagen, Denmark. Association for Computational Linguistics. https://doi.org/10.18653/v1/D17-1317

Simon Razniewski, Nitisha Jain, Paramita Mirza, and Gerhard Weikum. 2019. Coverage of information extraction from sentences and paragraphs. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5771–5776, Hong Kong, China. Association for Computational Linguistics. https://doi.org/10.18653/v1/D19-1583

Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling relations and their mentions without labeled text. In *Machine Learning and Knowledge Discovery in Databases*, pages 148–163, Berlin, Heidelberg. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-15939-8_10

Stephen E. Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford. 1995. Okapi at TREC-3. In *Overview of the Third Text REtrieval Conference (TREC-3)*, pages 109–126. Gaithersburg, MD: NIST.

Dwaipayan Roy, Sumit Bhatia, and Prateek Jain. 2020. A topic-aligned multilingual corpus of wikipedia articles for studying information asymmetry in low resource languages. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 2373–2380, Marseille, France. European Language Resources Association.

Evan Sandhaus, Philadelphia. 2008. The New York Times Annotated Corpus. *Linguistic Data Consortium*, 6(12):e26752.

Livio B. Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. 2019. Matching the blanks: Distributional similarity for relation learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2895–2905, Florence, Italy. Association for Computational Linguistics.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: A large-scale dataset for Fact Extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 809–819, New Orleans, Louisiana. Association for Computational Linguistics. https://doi.org/10.18653/v1/N18-1074

Xiaolan Wang, Xin L. Dong, Yang Li, and Alexandra Meliou. 2019. MIDAS: Finding the right web sources to fill knowledge gaps. In *2019 IEEE 35th International Conference on Data Engineering (ICDE)*, pages 578–589. https://doi.org/10.1109/ICDE.2019.00058

Gerhard Weikum, Luna Dong, Simon Razniewski, and Fabian M. Suchanek. 2021. Machine knowledge: Creation and curation of comprehensive knowledge bases. *Foundations and Trends in Databases*, 10(2–4):108–490. https://doi.org/10.1561/1900000064

Benfeng Xu, Quan Wang, Yajuan Lyu, Yong Zhu, and Zhendong Mao. 2021. Entity structure within and throughout: Modeling mention dependencies for document-level relation extraction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35(16), pages 14149–14157.

Yuan Yao, Deming Ye, Peng Li, Xu Han, Yankai Lin, Zhenghao Liu, Zhiyuan Liu, Lixin Huang, Jie Zhou, and Maosong Sun. 2019. DocRED: A large-scale document-level relation extraction dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 764–777, Florence, Italy. Association for Computational

Linguistics. https://doi.org/10.18653/v1/P19-1074

Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning. 2017. Position-aware attention and supervised data improve slot filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 35–45, Copenhagen, Denmark. Association for Computational Linguistics. https://doi.org/10.18653/v1/D17-1004