Coverage of Information Extraction from Sentences and Paragraphs

Simon Razniewski[1], **Nitisha Jain**[2], **Paramita Mirza**[1], **Gerhard Weikum**[1]

[1] Max Planck Institute for Informatics
[2] Hasso Plattner Institute

{srazniew, paramita, weikum}@mpi-inf.mpg.de
nitisha.jain@hpi.de

Abstract

Scalar implicatures are language features that imply the negation of stronger statements, e.g., *"She was married twice"* typically implicates that she was not married thrice. In this paper we discuss the importance of scalar implicatures in the context of textual information extraction. We investigate how textual features can be used to predict whether a given text segment mentions all objects standing in a certain relationship with a certain subject. Preliminary results on Wikipedia indicate that this prediction is feasible, and yields informative assessments.

1 Introduction

Following the cooperative principle, natural language utterances can implicate a range of assertions that are not explicitly stated (Grice, 1975). One specific class of implicatures are scalar implicatures, which concern the negation of stronger statements (Carston, 1998). Scalar implicatures are derived from Grice's maxim of quantity - that speakers would make stronger statements if possible, therefore, negation can be deduced if these are not made.

Yet the maxim of quantity interacts with the maxim of relevance, i.e., what is implicated depends on what is relevant in a given context. Consider the examples in Figure 1. From the first sentence, typical for biographical descriptions, most humans would draw the implicature that Obama has no other children. For the second sentence about Jolie, this implicature would typically not be drawn - she might well have other children that are too young or too old to be brought to school, but are not relevant in the school context.

The interaction between the maxims of quantity and relevance has implications for textual information extraction (IE). Textual IE usually produces (ideally canonicalized) subject-predicate-object triples, and annotates them with estimates of *correctness* (also called accuracy, precision or confidence), e.g., a 93% belief that Malia is really Obama's child. In contrast, IE usually lacks such an ability for *coverage* (also called recall or completeness). It is not able to estimate whether its extractions represent all facts pertaining to a certain topic, e.g., all children of Jolie.

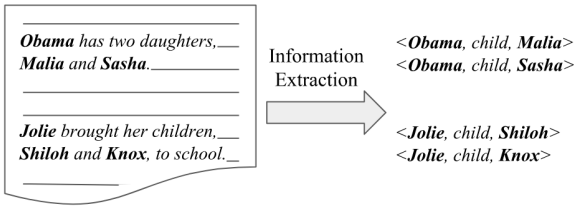

Figure 1: Two sentences with different coverage.

Importance of coverage-awareness. Coverage-awareness of IE is a crucial and highly desirable property for several downstream use cases. (i) Today's *question answering* systems are well geared for questions where exactly one answer should be returned (e.g., quiz questions or reading-comprehension tasks) (Fader et al., 2014; Yang et al., 2015). In contrast, for questions with sets of answers, QA systems often merely yield subsets, and do not inform the user about that. Similarly, they struggle with questions that have no answer, too often still returning a best-effort answer even if it is incorrect. Coverage awareness would enable a better treatment of both cases. (ii) Guiding editors in how to *prioritize curation* efforts is a key issue for collaboratively built and maintained knowledge bases such as Wikidata (Balaraman et al., 2018). Yet, methods to automatically identify incomplete parts are largely based on aggregate-level statistics, and do not consider entity-specific textual context (Razniewski et al., 2017; Galárraga et al., 2017). (iii) Coverage awareness would also be useful for *automated knowledge base construc-*

tion techniques, either to dynamically adjust confidence thresholds, i.e., lowering thresholds in case of low coverage, and increasing thresholds in case of too many extractions, or to reallocate search budgets to low-coverage regions, while stopping the exploration of complete areas (Ipeirotis et al., 2007; Jain et al., 2008).

Contribution and Approach. In this paper we analyze the viability of text coverage prediction. We use textual features to estimate whether a text segment contains all objects for a given subject-predicate pair, e.g., whether a given text mentions all children of Jolie. For an experimental study, we use fact counts for 5 Wikidata relations as ground truth. Using these counts, we train and evaluate on Wikipedia-extracted sentences and paragraphs, finding that coverage prediction is generally feasible and yields informative assessments.

Our conceptual contributions are:

- We introduce and define the novel problem of textual coverage prediction, and we discuss its key features.
- We present a method, along with experimental results that demonstrate its practical value.

Our experiments confirm that coverage estimation is possible, and yield the following technical insights:

- Features: Unigrams and bigrams provide informative cues towards coverage estimation.
- Scope: Coverage estimation is feasible for diverse domains ranging from family relations to organizational membership, and both on the level of sentences and paragraphs.

2 Background

Information extraction from text sources has been greatly advanced over the past two decades; see e.g., (Agichtein and Gravano, 2000; Etzioni et al., 2004; Suchanek et al., 2009; Mintz et al., 2009; Riedel et al., 2013; Dong et al., 2014; Shin et al., 2015; Mausam, 2016; Chiticariu et al., 2018; Stanovsky et al., 2018). The underlying methodologies span regular-expression matching, rule-based extraction, conditional random fields, constraint reasoning, all the way to deep learning. Depending on the task at hand, IE often achieves high correctness (sometimes above 90%). However, evaluating its coverage is inherently hard, as this would require exhaustively annotated corpora as gold standard. As a consequence, assessing and optimizing coverage has typically been an afterthought at best, and is usually completely disregarded.

In contrast, coverage (recall) is one of the key metrics in information retrieval (IR), i.e., in search applications. Here, recall is measured in terms of retrieving a large fraction of the relevant documents or passages, where relevance is stated by gold-standard annotations. In the context of entire IE workflows (e.g., for text analytics over business news), the prior works of Ipeirotis et al. (2007) and Jain et al. (2008) have considered optimizations for recall. However, this solely refers to the search-centric parts of such workflows, that is, the document or passage sets that are then fed into IE steps.

Grice's maxims of cooperative communication (Grice, 1975) introduce the concept of implicatures, which are conclusions that humans draw even though texts do not literally support them. The implicatures of interest here are scalar implicatures, i.e., the conclusion that no more facts are true than those explicitly stated (Carston, 1998). Scalar implicatures are closely connected to the closed-world assumption in logics, where statements are assumed to be false, unless explicitly stated. Yet, as exemplified in in Figure 1, due to the maxim of relevance, the scope of scalar implicatures may vary significantly.

Closest to coverage-awareness is recent work on counting quantifier extraction (Mirza et al., 2018). There, relation counts are extracted from phrases such as *"Jolie has six children"*, which, in a second step, are compared against fact counts in an existing KB. In contrast, the present work aims to directly predict the coverage of text segments.

3 Problem and Approach

While information extraction is a noisy process with both false positives and false negatives, our focus here is on whether, in principle, a text segment allows the extraction of all facts that hold in reality. For this purpose, we assume we have perfect knowledge of *all real-world facts* for the objects that are connected to a specific subject s and property p; we denote this object set as $RW\{o \mid sp\}$. Now assume an educated and linguistically versed human is presented with a text segment t and the task of telling which objects o she would assign to a fixed subject s and property p given solely the text t. We denote this ground-

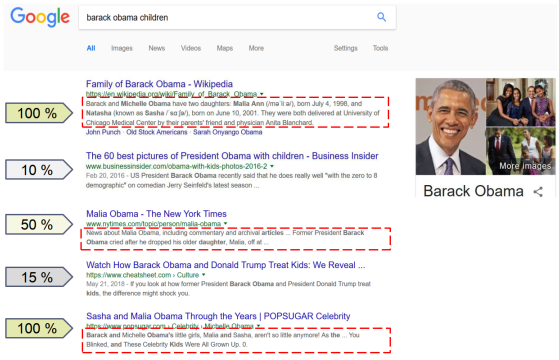

Figure 2: Possible application of a coverage classifier on web search snippets.

truth extraction as $GTE\{o \mid sp, t\}$.

Text Coverage Prediction Problem. Given a text segment t for a fixed subject s and property p, predict whether the human ground-truth from reading t matches the real-world facts:

$$GTE\{o \mid sp, t\} = RW\{o \mid sp\}.$$

This problem is different from assessing the quality of specific IE methods and tools. Since there are no perfect IE tools, considering the extractions from an IE tool would confound two distinct issues: 1) whether a text segment contains all information of interest (our present problem), and 2) what the recall of the specific IE tool is (a standard evaluation criterion for IE methods). Although our automated evaluation below necessarily builds on concrete choices for IE methods, our emphasis is on the fundamental problem of recall assessment given solely a text segment, as described above.

By casting the problem into a binary classification task, we look only at two cases: a) $GTE\{o \mid sp, t\}$ contains all real-world facts (*complete*), and b) it does not (*incomplete*). This formulation disregards complex graded cases, such as $GTE\{o \mid sp, t\}$ containing at least 70% of the real-world facts. Nevertheless, the problem naturally invites the use of scores that are confidences/probabilities. For example, for the first sentence in Figure 1, the probability to contain all $\langle Obama, child, *\rangle$-facts might be 0.9, while for the second sentence, the probability to contain all $\langle Jolie, child, *\rangle$ facts might be 0.4. An illustration of how such confidence scores could be applied to Web search snippets is shown in Figure 2.

Methods and baselines. We approach the problem as classification task for t. As state-of-the-art text classification methods, we use interpretable feature-based Support Vector Machines (SVMs) with unigrams and bigrams from t as input, and Long Short-Term Memory networks (LSTMs) (Tai et al., 2015). LSTMs are used to encode sentences/paragraphs, using word representations (d=100) that are learned from scratch (initialized uniformly). One hidden layer of size 256 with ReLU activation and an output layer with sigmoid activation are used for binary classification. We employ the Adam optimizer with default parameter values. The models were trained for 20 epochs. We also experimented with using pretrained (word2vec) embeddings, but found no improvement, possibly due to the unorthodox nature of the problem, where typical word semantics as relevant for classic NLP tasks like QA, translation etc. do not help.

In web extraction scenarios one could additionally also consider features such as subject popularity, the relative position of a text segment, or web-source reliability scores.

We employ three baselines. Two natural baselines are *length* and *#pnames*, which classify the longest text segments (by character length) or the text segments containing most proper names as complete, i.e., assume that the more information, the better. A lower bound is given by a third baseline, *random*, which simply tosses a coin to decide whether a text segment is classified as complete, or not. For all baselines, the classification thresholds/coin bias is chosen so that input class distributions are maintained, therefore their precision and recall coincide.

4 Experimental Setup

Predicates. We perform experiments for 5 Wikidata predicates that span three different domains: (i) family relations: *child* (P40), *spouse* (P26), (ii) education and work: *educatedAt* (P69), *employer* (P108), (iii) band compositions: via *hasPart* (P527) for instances of the *musical ensemble* (Q2088357) class.

Approximating ground-truth extractions. Due to the intellectual complexity of fact extraction, crowdsourcing annotations faces scalability challenges, particularly at the paragraph level. We therefore opt for approximating the ideal human extractions $GTE\{o \mid sp, t\}$ via the combination of open information extraction, predicate paraphrase matching and object label matching. Note that this specific choice is not decisive for our approach

and merely serves as a concrete scalable instantiation of our framework, human labels or other automated IE methods could be plugged in as well.

To evaluate whether a text segment contains an $\langle s, p, o \rangle$-fact, we rely on the open information extraction system OpenIE 4 (Pal and Mausam, 2016), the PATTY predicate paraphrase dictionary (Nakashole et al., 2012), and Wikidata entity alias names.

For example, suppose we are interested in extracting facts for the *child* property for the subject *Angelina Jolie*. OpenIE extracts the triple $\langle \underline{Maddy}, is\ first\ adopted\ \underline{son}\ [of], Jolie \rangle$ from the text segment *"Jolie's first adopted son is Maddy."* As *(i) Maddy* is one of the aliases of *Angelina Jolie's* child *Maddox Chivan* in IMDb, and *(ii) son* appears in the list of paraphrases for the *child* predicate, we consider that the text segment contains the fact $\langle Angelina\ Jolie, child, Maddox\ Chivan \rangle$.

Labelled data. We use distant supervision to automatically label data. Assuming that Wikidata's coverage is near-perfect for popular entities, for each of the predicates we collect the 1000-8000 most popular subjects in Wikidata, along with their facts for the respective property (see Table 1).[1] As many properties have a skew towards low frequencies, which may make completeness prediction trivial, we only considered subjects having at least two objects in Wikidata (#subj w/ $\geq$2 obj). We collected two granularities of text units, *sentences* and *paragraphs* that contain at least one object, as found on the Wikipedia pages of the respective subjects. To ensure that general features are learned, we mask proper names and specific numbers with generic placeholders. A text segment is labelled *complete*, if it contains, for each object listed on Wikidata, at least one first names or alias. It is labelled as *incomplete* otherwise. In total, we obtain about 300 complete and 2000 incomplete sentences and paragraphs per relation, which we split into 80% for training and 20% for testing.

5 Results and Discussion

Table 2 shows the precision, recall and F1-score in terms of identifying complete text segments. Both

[1]While using Wikidata as source for labels for distant supervision of Wikipedia texts may seem circuitous, we note that unlike Wikipedia, Wikidata is language-independent, thus, has potential for much higher coverage especially for entities more famous outside English-speaking countries.

	child	*spouse*	*member*	*employer*	*educatedAt*
#subj	40,145	45,261	8,901	58,731	273,128
#subj w/ $\geq$2 obj	15,022	4,055	1,022	12,885	72,847
seeds	1,000	1,000	1,000	1,000	8,000
Sentences +/-	135/2,050	119/2,444	672/10,358	47/1,499	447/ 2,603
Paragraphs +/-	217/1,595	385/2,044	930/ 5,362	108/1,248	339/ 2,384

Table 1: Number of Wikidata subjects and derived labelled text segments. +/- signifies *complete/incomplete* w.r.t. Wikidata.

Text unit	Model	*child* P	R	F	*spouse* P	R	F	*hasPart* P	R	F	*employer* P	R	F	*educatedAt* P	R	F
Sentence	*random*			.06			.04			.06			0			.14
	length			.05			.28			.13			0			.24
	#pnames			.05			.22			.17			0			.28
	SVM	.50	**.42**	**.46**	.35	.33	.34	.39	.23	.28	**.17**	**.13**	**.14**	**.55**	.39	.46
	LSTM	**.64**	.39	.45	**.58**	**.39**	**.47**	**.54**	**.69**	**.60**	0	0	0	.47	**.98**	**.64**
Paragraph	*random*			.12			.16			.15			.08			.12
	length			.17			.37			.26			.10			.21
	#pnames			.19			.40			.31			.20			.29
	SVM	**.50**	.44	.47	**.74**	.73	**.73**	**.59**	.53	**.56**	**.50**	.05	.09	**.70**	.57	**.63**
	LSTM	.41	**.83**	**.55**	.54	**1**	.70	.50	**.59**	.54	.34	**.75**	**.47**	.42	**.96**	.58

Table 2: Performance of coverage prediction.

SVMs and LSTMs outperform the baselines by a considerable margin, performing slightly better at paragraph than at sentence level. They perform best there on the *spouse* property (.73/.70 F1), followed by *educated at* (.63-.58 F1). They comparably fail at sentence level on *employer* (.14/0 F1), presumably because it is rather rare that all employments are listed in the same sentence. For instance, it is more common to find *"He served as a professor at* [*University A*]*, and also held an appointment at* [*University B*]*. In July 2007, he left* [*University A*] *and joined the faculty of* [*University C*]*. He was also a visiting professor at the* [*University D*]*."* as a complete paragraph.

In Table 3 we show the most informative bigrams for the n-gram-based SVMs on the paragraph level, for predicates having reasonably good F1 scores. Most of the highly weighted bigrams signal the beginning of name listing, such as *daughters* $\langle pname \rangle$, *married twice*, *featuring lineup* and *attended* $\langle pname \rangle$. Some bigrams convey temporal information, such as *later married*, *briefly attended* and *left graduating*, which indicate that the paragraph contains a narrative that lists object names for different time periods. This was particularly true for the *spouse*, *employer* and *educatedAt* predicates, for which usually only one object is valid at each timepoint.

We also find various terms indicating incompleteness, for instance *surviving* ("Had 5 children, but only Mary and Bob were surviving to adulthood"), *succeeded* ("She was later succeeded by her son James in her role as ..."), *addition* ("In addition, a daughter, Susan, was born in ...") among

child	*spouse*	*hasPart*	*educatedAt*
⟨*pname*⟩ *sons*	*married twice*	*featuring lineup*	*educated* ⟨*pname*⟩
grandsons ⟨*pname*⟩	*children* ⟨*pname*⟩	*lineup* ⟨*pname*⟩	*briefly attended*
⟨*num*⟩ *grandsons*	*second marriage*	*consists* ⟨*pname*⟩	*attended* ⟨*pname*⟩
daughters ⟨*pname*⟩	⟨*num*⟩ *son*	⟨*num*⟩ *tour*	*left graduating*
sons ⟨*pname*⟩	*later married*	*vocals* ⟨*propname*⟩	⟨*pname*⟩ *left*

Table 3: Selected important paragraph-level bigrams indicating completeness for SVMs.

Sentence	LSTM score
He was the father of actor Pierre Renoir (1885-1952), filmmaker Jean Renoir (1894-1979) and ceramic artist Claude Renoir (1901-1969).	0.54
His daughter Julie Gavras and his son Romain Gavras are also filmmakers.	0.46
Genghis Khan was aware of the friction between his sons (particularly between Chagatai and Jochi) and worried of possible conflict between them if he died.	0.42
"From this moment I am no longer the king; the king is Victor my son."	0.17

Table 4: Example LSTM predictions at sentence-level.

unigrams indicating incompleteness of *child* at sentence level, or *leaving* and *previously* for *employer*.

While at paragraph level, there is no clear winner, at sentence level LSTMs outperform SVMs. Possibly, this is due to latent representations being more important when features are sparse (i.e., text segments are short). Anecdotal LSTM sentence-level predictions are shown in Table 4. The full input and resulting predictions will be made available on Github.

6 Discussion

Task Difficulty. The prediction results, ranging in F1-score from 0 to .64 for the sentence level and .09 to .73 for the paragraph level, are significantly lower than typical scores in information extraction (e.g., up to .83 F1 in the KBP TAC 2017 challenge (Getman et al., 2017)). Several aspects contribute to the problem's hardness.

- *Training data quality*. We find that distantly supervised training data for recall is much noisier than for classical IE tasks, because knowledge bases such as Wikidata, despite having low error rates, have many gaps where they are incomplete (rather than incorrect). This mirrors a similar problem as found in (Mirza et al., 2018).
- *Low NED recall*. The task requires to match text mentions against KB entities. Yet even famous subjects frequently have obscure objects, e.g., none of Bill Gates' children has a Wikipedia page. NED tools consequently often failed to correctly resolve related mentions. In the present work we thus opted for lexical matching, trading a higher recall against a lower precision.
- *Time-variance*. While some KB relations are quite stable (e.g., children), others are more volatile, and may both grow or shrink over time (e.g., band membership) (Wijaya et al., 2015). Such dynamicity adds complexity to the recall assessment, as recall may then be specific to certain time points.

Relative recall. Our work has focused on estimating the recall w.r.t. reality, as judged by gold-standard annotators. An equally important question, close to previous work on species-count estimation (Salloum et al., 2013), is to estimate the recall relative to what can be maximally achieved by using the union of all possible sources. For instance, for many long-tail subjects, no source would hold complete information, but specific sources could still hold maximal information.

Modelling and reasoning. While we have shown that textual information can be useful in inferring the recall of extractions, recall estimation might benefit from more explicit modelling and reasoning. One relevant aspect could be temporal reasoning, for instance, a professional career without temporal gaps (e.g., high school till 1993, BSc. 1994-1997, then launch of a startup in 1998) is a helpful indicator towards complete education extraction. Such reasoning could be applied on top of temporal information extraction (Ling and Weld, 2010). Another aspect are statistical priors and typicality information. Information that rock bands often consist of one bassist, 1-2 guitarists, one vocalist and one drummer could be helpful in assessing extraction recall at extraction time, similar as done post-hoc in (Galárraga et al., 2017).

7 Conclusion

This paper presented a first investigation on IE coverage estimation. Our results support linguistic theories about scalar implicatures, and show that coverage estimation is generally feasible. The next challenge is to incorporate this into actual noisy IE extraction pipelines.

Acknowledgment. This research has been supported by an NVIDIA GPU hardware grant.

References

Eugene Agichtein and Luis Gravano. 2000. Snowball: Extracting relations from large plain-text collections. In *JCDL*.

Vevake Balaraman, Simon Razniewski, and Werner Nutt. 2018. Recoin: Relative completeness in Wikidata. In *Wiki workshop at WWW*.

Robyn Carston. 1998. Informativeness, relevance and scalar implicature. *Pragmatics And Beyond New Series*.

Laura Chiticariu, Marina Danilevsky, Yunyao Li, Frederick Reiss, and Huaiyu Zhu. 2018. SystemT: Declarative text understanding for enterprise. In *NAACL*.

Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. 2014. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *KDD*.

Oren Etzioni, Michael Cafarella, Doug Downey, Stanley Kok, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S Weld, and Alexander Yates. 2004. Web-scale information extraction in KnowItAll. In *WWW*.

Anthony Fader, Luke Zettlemoyer, and Oren Etzioni. 2014. Open question answering over curated and extracted knowledge bases. In *KDD*.

Luis Galárraga, Simon Razniewski, Antoine Amarilli, and Fabian M Suchanek. 2017. Predicting completeness in knowledge bases. In *WSDM*.

Jeremy Getman, Joe Ellis, Zhiyi Song, Jennifer Tracey, and Stephanie M Strassel. 2017. Overview of linguistic resources for the TAC KBP 2017 evaluations: Methodologies and results. In *TAC*.

Herbert Paul Grice. 1975. Logic and conversation.

Panagiotis G. Ipeirotis, Eugene Agichtein, Pranay Jain, and Luis Gravano. 2007. Towards a query optimizer for text-centric tasks. *ACM TODS*.

Alpa Jain, Panagiotis G. Ipeirotis, and Luis Gravano. 2008. Building query optimizers for information extraction: the SQoUT project. *SIGMOD Record*.

Xiao Ling and Daniel S Weld. 2010. Temporal information extraction. In *AAAI*.

Mausam. 2016. Open information extraction systems and downstream applications. In *AAAI*.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *ACL*.

Paramita Mirza, Simon Razniewski, Fariz Darari, and Gerhard Weikum. 2018. Enriching knowledge bases with counting quantifiers. In *ISWC*.

Ndapandula Nakashole, Gerhard Weikum, and Fabian Suchanek. 2012. PATTY: a taxonomy of relational patterns with semantic types. In *EMNLP*.

Harinder Pal and Mausam. 2016. Demonyms and compound relational nouns in nominal open IE. In *AKBC*.

Simon Razniewski, Vevake Balaraman, and Werner Nutt. 2017. Doctoral advisor or medical condition: Towards entity-specific rankings of knowledge base properties. In *ADMA*.

Sebastian Riedel, Limin Yao, Andrew McCallum, and Benjamin M Marlin. 2013. Relation extraction with matrix factorization and universal schemas. In *NAACL*.

Mariam Salloum, Xin Luna Dong, Divesh Srivastava, and Vassilis J Tsotras. 2013. Online ordering of overlapping data sources. *VLDB*.

Jaeho Shin, Sen Wu, Feiran Wang, Christopher De Sa, Ce Zhang, and Christopher Ré. 2015. Incremental knowledge base construction using DeepDive. *VLDB*.

Gabriel Stanovsky, Julian Michael, Luke Zettlemoyer, and Ido Dagan. 2018. Supervised open information extraction. In *NAACL*.

Fabian M Suchanek, Mauro Sozio, and Gerhard Weikum. 2009. SOFIE: a self-organizing framework for information extraction. In *WWW*.

Kai Sheng Tai, Richard Socher, and Christopher D. Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. In *ACL*.

Derry Tanti Wijaya, Ndapandula Nakashole, and Tom Mitchell. 2015. "a spousal relation begins with a deletion of engage and ends with an addition of divorce": Learning state changing verbs from wikipedia revision history. In *EMNLP*.

Yi Yang, Wen-tau Yih, and Christopher Meek. 2015. WikiQA: A challenge dataset for open-domain question answering. In *EMNLP*.