Knowledge-Guided Linguistic Rewrites for Inference Rule Verification

Prachi Jain and **Mausam**
Indian Institute of Technology, Delhi
New Delhi, India
{csz148211,mausam}@cse.iitd.ac.in

Abstract

A corpus of inference rules between a pair of relation phrases is typically generated using the statistical overlap of argument-pairs associated with the relations (e.g., PATTY, CLEAN). We investigate *knowledge-guided linguistic rewrites* as a secondary source of evidence and find that they can vastly improve the quality of inference rule corpora, obtaining 27 to 33 point precision improvement while retaining substantial recall. The facts inferred using cleaned inference rules are 29-32 points more accurate.

1 Introduction

The visions of machine reading (Etzioni, 2007) and deep language understanding (Dorr, 2012) emphasize the ability to draw inferences from text to discover implicit information that may not be explicitly stated (Schubert, 2002). This has natural applications to textual entailment (Dagan et al., 2013), KB completion (Socher et al., 2013), and effective querying over Knowledge Bases (KBs).

One popular approach for fact inference is to use a set of *inference rules* along with probabilistic models such as Markov Logic Networks (Schoenmackers et al., 2008) or Bayesian Logic Programs (Raghavan et al., 2012) to produce human-interpretable proof chains. While scalable (Niu et al., 2011; Domingos and Webb, 2012), this is bound by the coverage and quality of the background knowledge – the set of inference rules that enable the inference (Clark et al., 2014).

Antecedent	Consequent	Y/N?
(X, make a note of, Y)	(X, write down, Y)	Y
(X, offer wide range of, Y)	(X, offer variety of, Y)	Y
(X, make full use of, Y)	(Y, be used by, X)	Y
(X, be wounded in, Y)	(X, be killed in, Y)	N
(X, be director of, Y)	(X, be vice president of, Y)	N
(X, be a student at, Y)	(X, be enrolled at, Y)	N

Figure 1: Sample rules verified (Y) and filtered (N) by our method. Rules #4, #5 were correctly and #6 wrongly filtered.

The paper focuses on generating a *high precision* subset of inference rules over Open Information Extraction (OpenIE) (Etzioni et al., 2011) relation phrases (see Fig 1). OpenIE systems generate a schema-free KB where entities and relations are represented via normalized but not disambiguated textual strings. Such OpenIE KBs scale to the Web.

Most existing large-scale corpora of inference rules are generated using distributional similarity, like argument-pair overlap (Schoenmackers et al., 2010; Berant et al., 2012), but often eschew any linguistic or compositional insights. Our early analysis revealed that such inference rules have very low precision, not enough to be useful for many real tasks. For human-facing applications (such as IE-based demos), high precision is critical. Inference rules have a multiplicative impact, since one poor rule could potentially generate many bad KB facts.

Contributions: We investigate the hypothesis that *"knowledge-guided linguistic rewrites can provide independent verification for statistically-generated Open IE inference rules."* Our system KGLR's rewrites exploit the compositional structure of Open IE relation phrases alongside knowledge in resources like Wordnet and thesaurus. KGLR independently verifies rules from existing inference rule

corpora (Berant et al., 2012; Pavlick et al., 2015) and can be seen as additional annotation on existing inference rules. The verified rules are 27 to 33 points more accurate than the original corpora and still retain a substantial recall. The precision of inferred knowledge also has a precision boost of over 29 points. We release our KGLR implementation, its annotations on two popular rule corpora along with gold set used for evaluation and the annotation guidelines for further use (available at *https://github.com/dair-iitd/kglr.git*).

2 Related work

Methods for inference over text include random walks over knowledge graphs (Lao et al., 2011), matrix completion (Riedel et al., 2013), deep neural networks (Socher et al., 2013; Rocktäschel et al., 2015a), natural logic inference (MacCartney and Manning, 2007) and graphical models (Schoenmackers et al., 2008; Raghavan et al., 2012). Most of these need (or benefit from) a background knowledge of inference rules, including matrix completion (Rocktäschel et al., 2015b).

Inference rules are predominantly generated via extended distributional similarity – two phrases having a high degree of argument overlap are similar, and thus candidates for a unidirectional or a bidirectional inference rule. Methods vary on the base representation, e.g., KB relations (Galárraga et al., 2013; Grycner et al., 2015), Open IE relation phrases (Schoenmackers et al., 2010), syntactic-ontological-lexical (SOL) patterns (Nakashole et al., 2012), and dependency paths (Lin and Pantel, 2001). An enhancement is global transitivity (TNCF algorithm) for improving recall (Berant et al., 2012). The highest precision setting of TNCF ($\lambda = 0.1$) was released as a corpus (informally called CLEAN) of Open IE inference rules.[1]

Distributional similarity approaches have two fundamental limitations. First, they miss obvious commonsense facts, e.g., ⟨(X, married, Y) → (X, knows, Y)⟩ – text will rarely say that a couple know each other. Second, they are consistently affected by statistical noise and end up generating a wide variety of inaccurate rules (see rules #4, and #5 in Figure 1). Our early experiments with CLEAN revealed its precision to be about 0.49, not enough to be useful in practice, especially for human-facing applications.

Similar to our paper, some past works have used alternative sources of knowledge. Weisman *et al.* (2012) study inference between verbs (e.g., ⟨startle → surprise⟩), but they get low (0.4) precision. Wordnet corpus to generate inference rules for natural logic (Angeli and Manning, 2014) improved noun-based inference. But, they recognize relation entailments as a key missing piece. Recently, natural logic semantics is added to a paraphrase corpus (PPDB2.0). Many of their features, e.g., lexical/orthographic, multilingual translation based, are complimentary to our method.

We test our KGLR algorithm on CLEAN and entailment/paraphrase subset of PPDB2.0 (which we call PPDB_e).

3 Knowledge-Guided Linguistic Rewrites (KGLR)

Given a rule ⟨(X, r_1, Y) → (X, r_2, Y)⟩ or ⟨(X, r_1, Y) → (Y, r_2, X)⟩ we present KGLR, a series of rewrites of relation phrase r_1 to prove r_2 (egs in Fig 1). The last two rewrites deal with reversal of argument order in r_2; others are for the first case.

Thesaurus Synonyms: Thesauri typically provide an expansive set of potential synonyms, encompassing near-synonyms and contextually synonymous words. Thesaurus synonyms are not that helpful for *generating* inference rules (or else we will generate rules like ⟨produce → percolate⟩). However, they are excellent in rule verification as they provide evidence independent from statistical overlap metrics.

We allow any word/phrase w_1 in r_1 to be replaced by any word/phrase w_2 from its thesaurus synsets as long as (1) w_2 has same part-of-speech as w_1 and (2) w_2 is seen in r_2 at the same distance from left of the phrase as w_1 in phrase r_1, but ignoring words dropped due to other rules whose details follows next. To define a thesaurus synset, we tag w_1 with its POS and look for all thesaurus synsets of that POS containing w_1. We allow this rewrite if PMI(w_1, w_2) $> \lambda$ (=-2.5 based on a devset). We calculate PMI as $log \frac{(\#w_1 \text{ occurs in synsets of } w_2 + \#w_2 \text{ occurs in synsets of } w_1)}{(\#\text{ of synsets of } w_1 \times \#\text{ of synsets of } w_2)}$. Some words can be both synonyms and antonyms in different situations. For example, thesaurus lists

[1] *http://u.cs.biu.ac.il/~nlp/resources/downloads/predicative-entailment-rules-learned-using-local-and-global-algorithms*

'bad' as both a synonym and antonym of 'good'. We don't allow such antonyms in these rewrites.

Thesarus synonyms can verify ⟨offer a vast range of → provide a wide range of⟩, since offer-provide, and vast-wide are thesaurus synonyms. We use Roget's 21^{st} Century Thesaurus in KGLR implementation.

Negating rules: We reject rules where r_2 explicitly negates r_1 or vice versa. We reject a rule if r_2 is same as r_1 if we drop 'not' from one of them. For example, the rule ⟨be the president of → be not the president of⟩, will be rejected.

Wordnet Hypernyms: We replace word/phrase w in r_1 by its Wordnet hypernym if it is in r_2. We prove ⟨be highlight of → be component of⟩, as Wordnet lists 'component' as a hypernym of 'highlight'.

Dropping Modifiers: We drop any adjective, adverb, superlatives or comparatives (e.g., 'more', 'most') from r_1. This lets us verify ⟨be most important part of → be part of⟩.

Gerund-Infinitive Equivalence: We convert infinitive constructions into gerunds and vice versa. For example, ⟨starts to drink ↔ starts drinking⟩.

Deverbal Nouns: We use Wordnet's derivationally related forms to compute a verb-noun pair list. We allow back and forth conversions from "be noun of" to related verb. So, we verify ⟨be cause of → cause⟩.

Light Verbs and Serial Verbs: If a light verb precede a word with derivationally related noun sense, we delete it. Similarly, if a serial verb precede a word with derivationally related verb sense, we delete it. We identify light verbs via the verbs that frequently precede a ⟨(a|an) (verb|deverbal noun)⟩ pair in Wikipedia. Serial verbs are identified as the verbs that frequently precede another verb in Wikipedia. Thus we can convert ⟨take a look at → look at⟩.

Preposition Synonyms: We manually create a list of preposition near-synonyms such as into-to, in-at, at-near. We replace a preposition by its near-synonym. This proves ⟨translated into → translated to⟩.

Be-Words & Determiners: We drop be-words ('is', 'was', 'be', etc.) and determiners from r_1 and r_2.

Active-Passive: We allow (X, $verb$, Y) to be rewritten as (Y, $be\ verb\ by$, X).

Redundant Prepositions: We find that often prepositions other than 'by' can be alternatively used with passive forms of some verbs. Moreover, some prepositions can be redundantly used in active forms too. For example, ⟨(X, absorb, Y) ↔ (Y, be absorbed *in*, X)⟩, or similarly, ⟨(X, attack, Y) ↔ (X, attack *on*, Y)⟩. To create such a list of verb-preposition pairs, we simply trust the argument-overlap statistics. Statistics here does not make that many errors since the base verb in both relations is the same.

3.1 Implementation

KGLR allows repeated application of these rewrites to modify r_1 and r_2. If it achieves $r_1 = r_2$ it verifies the inference rule. For tractable implementation KGLR uses a depth first search approach where a search node maintains both r_1 and r_2. Search does not allow rewrites that introduce any new lexical (lemmatized) entries not in original words(r_1) ∪ words(r_2). If it can't apply any rewrite to get a new node, it returns failure.

Many rules are proved by a sequence of rewrites. E.g., to prove ⟨(X, be a major cause of, Y) → (Y, be caused by, X)⟩, the proof proceeds as: (X, be a major cause of, Y) → (X, be major cause of, Y) → (X, be cause of, Y) → (X, cause, Y) → (Y, be caused by, X) by dropping determiner, dropping adjective, deverbal noun, and active-passive transformation respectively. Similarly, ⟨(X, helps to protect, Y) → (X, look after, Y)⟩ follows from gerund-infinitive conversion (helps protect), dropping support from serial verbs (protect), and thesaurus synonym (look after).

4 Experiments

KGLR verifies a subset of rules from CLEAN and PPDB_e to produce, VCLEAN and VPPDB_e. Our experiments answer these research questions: (1) What is the precision and size of the verified subsets compared to original corpora?, (2) How does additional knowledge generated after performing inference using these rules compare with each other? and (3) Which rewrites are critical to KGLR performance?

Comparison of CLEAN and VCLEAN: The original CLEAN corpus has about 102K rules. KGLR verifies about 36K rules and filter 66K rules out. To estimate the precisions of CLEAN and VCLEAN we independently sampled a random subset of 200 inference rules from each and asked two annotators (graduate level NLP students) to label the rules as

correct or incorrect. Rules were mixed together and the annotators were blind to the system that generated a rule. Our initial annotation guideline was similar to that of textual entailment – label a rule as correct if the consequent can usually be inferred given the antecedent, for most naturally occurring argument-pairs for the antecedent.

Our annotators faced one issue with the guideline – some inference rules were valid if (X,Y) were bound to specific types, but not for others. For example, $\langle$(X, be born in, Y) $\rightarrow$ (X, be birthplace of, Y)$\rangle$ is valid if Y is a location, not if Y is a year. Even seemingly correct inference rules, e.g., $\langle$(X, is the father of, Y) $\rightarrow$ (Y, is the child of, X)$\rangle$, can make unusual incorrect inferences: (Gandhi, is the father of, India) does not imply (India, is the child of, Gandhi). Unfortunately, these corpora don't associate argument-type information with their inference rules.

To mitigate this we refined the annotation guidelines to accept inference rules as correct as long as they are valid for *some* type-pair. The inter-annotator agreement with this modification was 94% (κ = 0.88). On the subset of the tags where the two annotators agreed we find the precision of CLEAN to be 48.9%, whereas VCLEAN was evaluated to be 82.5% precise – much more useful for real-world applications. Multiplying the precision with their sizes, we find the effective yield[2] of CLEAN to be 50K compared to 30K for VCLEAN. Overall, we find that VCLEAN obtains a 33 point precision improvement with an effective yield of about 60%.

Error Analysis: Most of VCLEAN errors are due to erroneous (or unusual) thesaurus synonyms. For missed recall, we analyzed CLEAN's sample missed by VCLEAN. We find that only about 13% of those are world knowledge rules (e.g., rule #6 in Figure 1). Other missed recall is because of some missing rewrites, missing thesaurus synonyms, spelling mistakes. These can potentially be captured by using other resources and adding rewrite rules.

Comparison of PPDB$_e$ and VPPDB$_e$: Unlike CLEAN, PPDB2.0 associates a confidence value for each rule, which can be varied to obtain different levels of precision and yield. We control for yield so that we can compare precisions directly.

We operate on PPDB$_e$ subset that has an Open IE-like relation phrase on both sides; this was identified by matching to ReVerb syntactic patterns (Etzioni et al., 2011). This subset is of size 402K. KGLR on this produces 85K verified rules (VPPDB$_e$). We find the threshold for confidence values in PPDB$_e$ that achieves the same yield (confidence > 0.342).

We perform annotation on PPDB$_e$(0.342) and VPPDB$_e$ using same annotation guidelines as before. The inter-annotator agreement was 91% (κ = 0.82). On the subset of the tags where the two annotators agreed we find the precision of PPDB$_e$ to be low – 44.2%, whereas VPPDB$_e$ was evaluated to be 71.4% precise. We notice that about 4 in 5 PPDB relation phrases are of length 1 or 2 (whereas 50% of CLEAN relation phrases are of length $\geq$ 3). This contributes to a slightly lower precision of VPPDB$_e$, as most rules are proved by *thesaurus synonymy* and the power of KGLR to handle compositionality of longer relation phrases does not get exploited.

Comparison of Inferred Facts: A typical use case of inference rules is in generating new facts by applying inference rules to a KB. We independently apply VCLEAN's, CLEAN's, PPDB$_e$'s and VPPDB$_e$'s inference rules on a public corpus of 4.2 million ReVerb triples.[3] Since ReVerb itself has significant extraction errors (our estimate is 20% errors) and our goal is to evaluate the quality of inference, we restrict this evaluation to only the subset of accurate ReVerb extractions.

VCLEAN *and* CLEAN *facts:* We sampled about 200 facts inferred by VCLEAN rules and CLEAN rules each (applied over accurate ReVerb extractions) and gave the original sentence as well as inferred facts to two annotators. We obtained a high inter-annotator agreement of 96.3%(κ = 0.92) and we discarded disagreements from final analysis. Overall, facts inferred by CLEAN achieved a precision of about 49.1% and those inferred by VCLEAN obtained a 81.6% precision. The estimated yields of fact corpora (precision$\times$size) are 7 and 4.5 million for CLEAN and VCLEAN respectively. This yield estimate does not include the initial 4.2 million facts.

PPDB$_e$ *and* VPPDB$_e$ *facts:* As done previously, we sampled 200 facts inferred by PPDB$_e$ and VPPDB$_e$ rules, which were annotated by two annotators. We obtained a good inter annotator agree-

[2] Yield is proportional to recall

[3] Available at *http://reverb.cs.washington.edu*

System	CLEAN	VCLEAN
Size	102,565	36,229
Rule Precision	48.9%	82.5%
Rule Yield	50,154	29,889
Fact Precision	49.1%	81.6%
Fact Yield	7 million	4.5 million
System	**PPDB$_e$(0.342)**	**VPPDB$_e$**
Size	85,272	85,261
Rule Precision	44.2%	71.4%
Fact Precision	22.16%	51.30%
Fact Yield	41 million	35 million

Figure 2: The precision and yield of inference rules after KGLR validation, and that of KB generated by inference using these rule-sets. Comparison with PPDB$_e$ is yield-controlled.

System	Precision	Recall
KGLR (all rules)	85.4%	62.0%
w/o Negating Rules	85.4%	62.0%
w/o Antonyms	84.2%	62.0%
w/o Wordnet Hypernyms	86.1%	59.3%
w/o Dropping Modifiers	84.9%	59.6%
w/o Gerund-Infinitive Equivalence	85.2%	61.0%
w/o Light and Serial Verbs	85.0%	59.9%
w/o Deverbal Nouns	85.4%	62.0%
w/o Preposition Synonyms	86.9%	56.9%
w/o Active-Passive	85.0%	54.5%
w/o Redundant Prepositions	86.1%	61.6%

Figure 3: Ablation study of rule verification using KGLR rewrites on our devset of 600 CLEAN rules

ment of 90.0%(κ = 0.8) and we discarded disagreements from final analysis. Overall, facts inferred by PPDB$_e$ achieved a really poor precision - 22.2% and those inferred by VPPDB$_e$ obtained an improvement of about 29 points (51.3% precision). Short relation phrases (mostly of length 1 or 2, which forms 80% of PPDB$_e$) contribute to low precision of VPPDB$_e$. Example low precision VPPDB$_e$ rules include $\langle$ (X, be, Y) $\rightarrow$ (X, obtain, Y)$\rangle$, $\langle$ (X, include, Y) $\rightarrow$ (X, come, Y)$\rangle$, which were inaccurately verified due to thesaurus errors. The estimated yields of fact corpora are 41 million and 35 million for PPDB$_e$ and VPPDB$_e$ respectively.

Ablation Study of KGLR rewrites: We evaluate the efficacy of different rewrites in KGLR by performing an ablation study (see Table 3). We ran KGLR by turning of one rewrite on a sample of 600 CLEAN rules (our development set) and calculate its precision and recall. The ablation study highlights that most rewrites add some value to the performance of KGLR, however *Antonyms* and *Dropping modifiers* are particularly important for precision and *Active-Passive* and *Redundant Preposition* add substantial recall.

5 Discussion

KGLR's value is in precision-sensitive tasks such as a human-facing demo, or downstream NLP application (like question answering) where error multiplication is highly undesirable. Along with high precision, it still obtains acceptably good yield.

Our annotators observe the importance of type-restriction of arguments for inference rules (similar to rules in (Schoenmackers et al., 2010)). Type annotation of existing inference rule corpora is an important step for obtaining high precision and clarity.

Inference rules are typically of two types – linguistic/synonym rewrites, which are captured by our work, and world knowledge rules (see rule #6 in Fig 1), which are not. We were surprised to estimate that about 87% of CLEAN, which is a statistically-generated corpus, is just linguistic rewrites! Obtaining world knowledge or common-sense rules at high precision and scale continues to be the key NLP challenge in this area.

6 Conclusions

We present Knowledge-guided Linguistic Rewrites (KGLR) which exploits the compositionality of relation phrases, guided by existing knowledge sources, such as Wordnet and thesaurus to identify a high precision subset of an inference rule corpus. Validated CLEAN has a high precision of 83% (vs 49%) at a yield of 60%. Validated PPDB$_e$ has a precision of 71% (vs 44%) at same yield. The precision of inferred facts has about 29-32 pt precision gain. We expect KGLR to be effective for precision-sensitive applications of inference. The complete code and data has been released for the research community.

Acknowledgments: We thank Ashwini Vaidya and the anonymous reviewers for their helpful suggestions and feedback. We thank Abhishek, Aditya, Ankit, Jatin, Kabir, and Shikhar for helping with the data annotation. This work was supported by Google language understanding and knowledge discovery focused research grants to Mausam, a KISTI grant and a Bloomberg grant also to Mausam. Prachi was supported by a TCS fellowship.

References

Gabor Angeli and Christopher D Manning. 2014. Naturalli: Natural logic inference for common sense reasoning. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Jonathan Berant, Ido Dagan, Meni Adler, and Jacob Goldberger. 2012. Efficient tree-based approximation for entailment graph learning. In *The 50th Annual Meeting of the Association for Computational Linguistics, Proceedings of the System Demonstrations, July 10, 2012, Jeju Island, Korea*.

Peter Clark, Niranjan Balasubramanian, Sumithra Bhakthavatsalam, Kevin Humphreys, Jesse Kinkead, Ashish Sabharwal, and Oyvind Tafjord. 2014. Automatic construction of inference-supporting knowledge bases.

Ido Dagan, Dan Roth, Mark Sammons, and Fabio Massimo Zanzotto. 2013. *Recognizing Textual Entailment: Models and Applications*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

Pedro M. Domingos and William Austin Webb. 2012. A tractable first-order probabilistic logic. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, July 22-26, 2012, Toronto, Ontario, Canada.*

Bonnie Dorr. 2012. Language programs at Darpa. AKBC-WEKEX 2012 Invited Talk.

Oren Etzioni, Anthony Fader, Janara Christensen, Stephen Soderland, and Mausam. 2011. Open information extraction: The second generation. In *IJCAI*, volume 11, pages 3–10.

Oren Etzioni. 2007. Machine reading of web text. In *Proceedings of the 4th International Conference on Knowledge Capture (K-CAP 2007), October 28-31, 2007, Whistler, BC, Canada*, pages 1–4.

Luis Antonio Galárraga, Christina Teflioudi, Katja Hose, and Fabian Suchanek. 2013. Amie: association rule mining under incomplete evidence in ontological knowledge bases. In *Proceedings of the 22nd international conference on World Wide Web*, pages 413–422. International World Wide Web Conferences Steering Committee.

Adam Grycner, Gerhard Weikum, Jay Pujara, James Foulds, and Lise Getoor. 2015. Relly: Inferring hypernym relationships between relational phrases. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 971–981, Lisbon, Portugal, September. Association for Computational Linguistics.

Ni Lao, Tom Mitchell, and William W Cohen. 2011. Random walk inference and learning in a large scale knowledge base. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 529–539. Association for Computational Linguistics.

Dekang Lin and Patrick Pantel. 2001. Dirt@ sbt@ discovery of inference rules from text. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 323–328. ACM.

Bill MacCartney and Christopher D Manning. 2007. Natural logic for textual inference. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 193–200. Association for Computational Linguistics.

Ndapandula Nakashole, Gerhard Weikum, and Fabian Suchanek. 2012. Patty: a taxonomy of relational patterns with semantic types. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1135–1145. Association for Computational Linguistics.

Feng Niu, Christopher Ré, AnHai Doan, and Jude W. Shavlik. 2011. Tuffy: Scaling up statistical inference in markov logic networks using an RDBMS. *PVLDB*, 4(6):373–384.

Ellie Pavlick, Johan Bos, Malvina Nissim, Charley Beller, Benjamin Van Durme, and Chris Callison-Burch. 2015. Adding semantics to data-driven paraphrasing. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics (ACL 2015)*.

Sindhu Raghavan, Raymond J. Mooney, and Hyeonseo Ku. 2012. Learning to "read between the lines" using bayesian logic programs. pages 349–358, July.

Sebastian Riedel, Limin Yao, Andrew McCallum, and Benjamin M. Marlin. 2013. Relation extraction with matrix factorization and universal schemas. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA*, pages 74–84.

Tim Rocktäschel, Edward Grefenstette, Karl Moritz Hermann, Tomás Kociský, and Phil Blunsom. 2015a. Reasoning about entailment with neural attention. *CoRR*, abs/1509.06664.

Tim Rocktäschel, Sameer Singh, and Sebastian Riedel. 2015b. Injecting Logical Background Knowledge into Embeddings for Relation Extraction. In *Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*.

Stefan Schoenmackers, Oren Etzioni, and Daniel S Weld. 2008. Scaling textual inference to the web. In *Proceedings of the Conference on Empirical Methods in*

Natural Language Processing, pages 79–88. Association for Computational Linguistics.

Stefan Schoenmackers, Oren Etzioni, Daniel S Weld, and Jesse Davis. 2010. Learning first-order horn clauses from web text. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1088–1098. AssociaFrition for Computational Linguistics.

Lenhart Schubert. 2002. Can we derive general world knowledge from texts? In *Proceedings of the second international conference on Human Language Technology Research*, pages 94–97. Morgan Kaufmann Publishers Inc.

Richard Socher, Danqi Chen, Christopher D Manning, and Andrew Ng. 2013. Reasoning with neural tensor networks for knowledge base completion. In *Advances in Neural Information Processing Systems*, pages 926–934.

Hila Weisman, Jonathan Berant, Idan Szpektor, and Ido Dagan. 2012. Learning verb inference rules from linguistically-motivated evidence. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, EMNLP-CoNLL 2012*, pages 194–204.