UnCommonSense in Action! Informative Negations for Commonsense Knowledge Bases

Hiba Arnaout*
harnaout@mpi-inf.mpg.de
Max Planck Institute for Informatics
Saarbrücken, Germany

Tuan-Phong Nguyen*
tuanphong@mpi-inf.mpg.de
Max Planck Institute for Informatics
Saarbrücken, Germany

Simon Razniewski
srazniew@mpi-inf.mpg.de
Max Planck Institute for Informatics
Saarbrücken, Germany

Gerhard Weikum
weikum@mpi-inf.mpg.de
Max Planck Institute for Informatics
Saarbrücken, Germany

ABSTRACT

Knowledge bases about commonsense knowledge i.e., CSKBs, are crucial in applications such as search and question answering. Prominent CSKBs mostly focus on positive statements. In this paper we show that materializing important negations about everyday concepts increases the usability of CSKBs. We present UnCommonSense, a web portal to explore informative negations about everyday concepts: (i) in a trivia interface, we show how users can explore negative statements about concepts of their choice; (ii) in a query interface, where explicit negative triple-patterns can be queried for, we show how our method can produce more accurate and relevant answers than the positive-only baseline. UnCommonSense can be accessed at: https://uncommonsense.mpi-inf.mpg.de/.

KEYWORDS

Knowledge Bases, Commonsense knowledge, Negation

ACM Reference Format:
Hiba Arnaout, Tuan-Phong Nguyen, Simon Razniewski, and Gerhard Weikum. 2023. UnCommonSense in Action! Informative Negations for Commonsense Knowledge Bases. In *Proceedings of The 16th ACM International WSDM Conference (WSDM '23).* ACM, New York, NY, USA, 4 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Motivation and Problem. Commonsense knowledge is important for many applications like question answering and dialogue agents. This knowledge is often stored in triple form in commonsense knowledge bases (CSKBs), e.g., (`vinegar, HasProperty, acidic`). Recently, we have seen a rising interest in constructing, curating, and querying such CSKBs. State-of-the-art CSKBs mainly focus on storing positive information, and collect little negative information. This poses a major limitation when downstream use has to decide whether absent information is false or missing [4]. With the open-world assumption (OWA) that most large-scale CSKBs postulate, an absent statement is *unknown*, regardless of whether it is false in reality. For instance, in Ascent [8], we know that *elephants* have tusks, are found in the wild, and have the color grey. This information is expressed in triple form, namely (`elephant, HasA, tusk`), (`elephant, AtLocation, the wild`), and (`elephant, HasProperty, grey`). Due to the OWA, absent information about further behavior or properties about *elephants* is not known to be true or false for a fact. For example, "*Are elephants nocturnal?*", "*Can they jump like many other land mammals?*". To empower downstream use cases, explicit assertion of negated statements can be very useful, e.g., (`elephant, NotHasProperty, nocturnal`), (`elephant, NotCapableOf, jump`).

Approach. The system demonstrated in this paper relies on the UnCommonSense method [3]. In a nutshell: given a target concept, e.g., *elephant*, comparable concepts are computed by employing structured taxonomies and latent similarity measures, e.g., other *wild animals* like *zebra, tiger, lion.* Among these comparable concepts the local closed-world assumption (LCWA) is postulated. Therefore, any positive statement that holds for *at least one* of the comparable concepts and *not* the target concept is a candidate negative statement. Restricting the inferences to information about *comparable*, rather than *random* (e.g., *cake, newspaper*), concepts produces much more relevant candidate statements, in this case animal-related statements such as (`elephant, NotIsA, carnivore`) and (`elephant, NotHasA, claw`). Nonetheless, due to the incompleteness of large-scale CSKBs, inferred negations might be inaccurate, i.e., positive in reality but simply missing from the CSKB. For instance, (`elephant, HasA, eye`) is a missing statement from Ascent. Moreover, lightly-canonicalized CSKBs might contain multiple phrases indicating the same meaning, e.g., in Ascent, (`butterfly, CapableOf, lie their eggs`) but (`wasp, CapableOf, lay egg`). This contradiction between information about target concept *butterfly* and comparable concept *wasp* will be overlooked during the previous inference step. To overcome these issues, we scrutinize the candidates against related statements in the input CSKB using sentence embeddings [9] and against a pre-trained language model (LM) as an external source of latent knowledge [7]. Finally, the potentially large set of candidates is ranked

*Equal contribution.

WSDM '23, February 27– March 03, 2023, Singapore

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/XXXXXXX.XXXXXXX

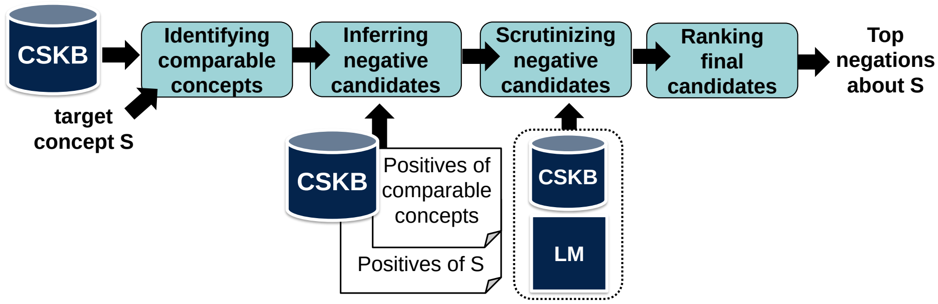

Figure 1: Architecture of UNCOMMONSENSE.

by computing informativeness using statistical scores, i.e., relative frequency within groups of comparable concepts. For example, while *elephant* cannot, 67% of its sibling concepts, i.e., comparable concepts, can jump. This procedure generates negations of significantly higher accuracy and informativeness than previous methods. Further details are in [3].

Demo. We present the UNCOMMONSENSE portal, where users can browse negative trivia about concepts of their choice, and query the Ascent KB using explicit negated relations. The method is applicable to any other commonsense KB, e.g., ConceptNet [13] (Ascent is chosen due to its higher coverage of statements in the ConceptNet schema). The demo is accessible at https://uncommonsense.mpi-inf.mpg.de/.

2 UNCOMMONSENSE

2.1 Method Description

The web portal is based on the research work published in [3] and shown in Figure 1. Given a target concept *S* (*elephant*) and a CSKB *K* (Ascent):

(1) We identify comparable concepts: To ensure highly thematic candidate negations, we begin by finding relevant context, i.e., parts of the CSKB, where LCWA can be reasonably postulated. We opt for a combination of two complementary similarity measures: (i) using hypernymy relations [5], we collect concepts that share a class with *S*, e.g., *elephant*, *squirrel*, and *lion* are all taxonomic siblings under *land mammals*; then (ii) using latent representations [17], we compute cosine similarity between embeddings of *S* and unordered taxonomic siblings collected in (i), e.g., *lion* is closer to *elephant* than *squirrel*. We consider the closest-*k* siblings for the next step.

(2) We infer negative candidates: We generate the initial candidate set by computing the difference between *positives about the comparable concepts* and *positives about S*. For example, if `(HasA, tongue)`, `(CapableOf, jump)`, and `(HasProperty, carnivore)` hold for the comparable concept *lion* and `(HasA, tongue)` holds for *elephant*, the initial candidate negations are `(elephant, NotCapableOf, jump)` and `(elephant, NotHasProperty, carnivore)`.

(3) We scrutinize candidate negations: To remove candidates that might be *falsely* identified as negative, due to the incompleteness or lack of phrase disambiguation in *K*, we perform two kinds of plausibility checks: (i) in KB-based check, we compute the semantic-similarity [9] to get rid of any remaining contradictions that were not covered by the exact matching of the inference step. For instance, candidate `(elephant, NotHasProperty, big animal)` is eliminated due to the positive statement `(elephant, HasProperty, large animal)` in *K*; (ii) in LM-based check, we probe pre-trained LMs for any possibly missing information from *K*. For instance, the probe "[MASK] eat grass." with prediction *elephant* at position 76 eliminates the candidate `(elephant, NotCapableOf, eat grass)`. Moreover, to remove candidates that are nonfactual, we eliminate any candidate that is identified as *too generic*, i.e., it holds for at least 5% of all concepts in *K*. For instance, candidate statement `(elephant, NotHasProperty, amazing)` is discarded since `(HasProperty, amazing)` holds for 16% of all concepts in *K*.

(4) We score by informativeness: With a potentially large number of remaining candidates, ranking is crucial. For instance, final candidate set for *S* = *elephant* with 40 comparable concepts in Ascent contains over 1.6k candidate negative statements. We quantify the inforamtiveness of a certain candidate negation by how *uncommon it is amongst comparable concepts*, i.e., using statistical frequency within the group of siblings. For example, 3 out of 4 siblings of elephants *can jump* (while elephants cannot) and 1 out of 4 siblings *has hoof* (while elephants do not). Therefore, the former is more noteworthy.

2.2 Web Portal

Implementation. The web portal is implemented in Python using the Django framework[1]. We use nginx[2] as web sever and store our datasets in a PostgreSQL database. The demo is deployed on a Debian virtual machine at the Max Planck Institute for Informatics that has 8GB of RAM and 50GB storage.

Data and Method Hyperparameters. This demo covers all 8029 *primary* concepts in Ascent. The data follows the established ConceptNet schema i.e., canonicalized concepts and relations (both positive and negative). We initially produce 6.7 billion[3] negations from assuming CWA, which are reduced to 46 million negations after LCWA is postulated, and finally to 13.8 million final negations

[1] https://www.djangoproject.com/
[2] https://www.nginx.com/
[3] For this dataset, we only need to store samples for displaying

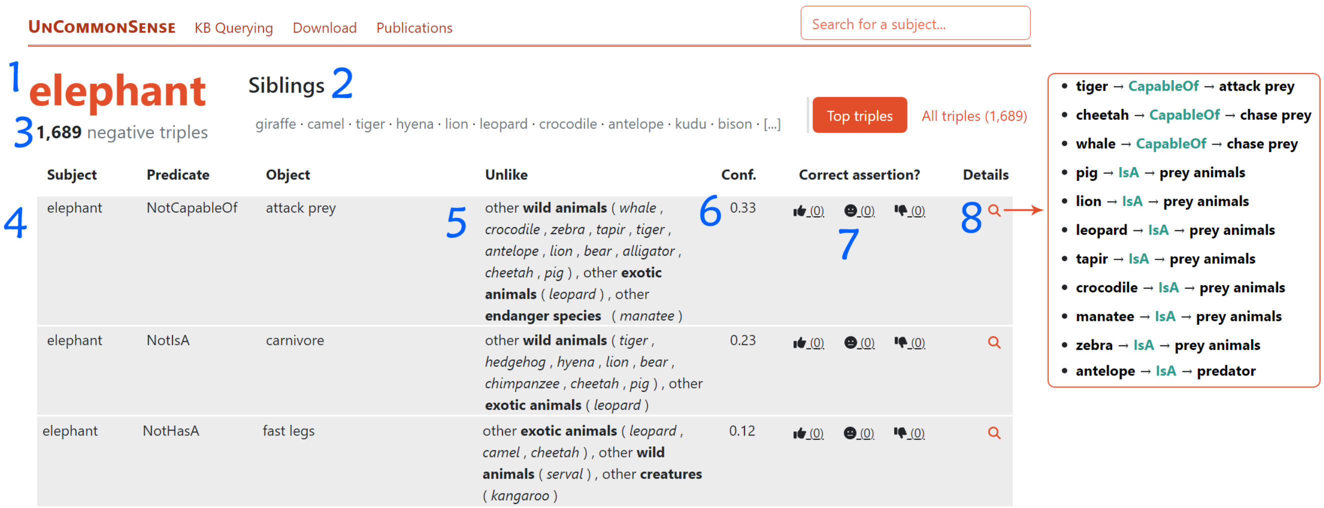

Figure 2: Negative statements about *elephant*.

after the scrutinizing step. We set the hyperparameters to their best-performing values as reported in [3], namely we set the number of comparable concepts to 30, quality check threshold to 0.05, semantic similarity threshold to 0.7, and rank threshold of LM to 50.

Per-concept Statements. The main function of UNCOMMONSENSE allows users to inspect informative negations about concepts of their choice (see Figure 2). This interface has an target concept field **(1)** which takes an Ascent primary concept as input (i.e., "search for a subject" auto-completion field at top-right side). The 10 most relevant siblings (comparable concepts) are also displayed **(2)**. To give the user a feel of the full size of the final negations set, the total number of results is displayed **(3)**. Each result, i.e., negative statement, is presented in a Subject-Predicate-Object format **(4)** where the *Subject* is the target concept chosen by the user, the *Predicate* is a canonicalized relation (following ConceptNet's negative relations by prefixing a positive pre-defined relation with "Not"), and *Object* is a phrase that holds for at least one of the siblings in Ascent. To give the user a glimpse into which of the siblings a certain negation *holds* for and potentially explain its particular salience to the target concept, we extend every result with a provenance **(5)**. This negation-explanation can be read as "*Subject* is *Predicate Object* unlike other *class1-shared-with-target* (*sibling1, sibling2,..*), other *class2-shared-with-target* (*sibling6, sibling7, ..*), ..". **(6)** shows the confidence score of a certain negation, which is computed according to the relaxed informativeness quantification proposed in [3]. Users can give feedback on the accuracy of these *inferred* negations **(7)**. Finally, users can click on details **(8)** to see how else was this result statement phrased for the siblings.

3 DEMONSTRATION EXPERIENCE

We show how users can interact with the web portal through two scenarios: 1) Exploring negative statements about a concept of choice, and 2) Querying the CSKB using explicit negative relations.

Scenario 1 - Knowledge Exploration. The user is an elementary school student who is fascinated by the animal kingdom. She has explored many positive statements about them in Ascent [4], namely about their properties and what they are capable of doing. Next, she would like to explore more on the things she might not be aware of. By querying *elephant* in UNCOMMONSENSE, shown in Figure 2, she learns that, unlike other *exotic animals* such as *leopard* and *cheetah*, *elephants* do not have fast legs. She also learns that they do not attack preys. This made perfect sense to her after also learning that they are not carnivorous, unlike other *wild animals* such as *tiger*, *lion*, and *chimpanzee*.

Scenario 2 - Querying CSKB. The user is preparing for a meal and is looking for food items or meal ideas that do not require using the oven, since he does not own one. He queries Ascent CSKB using UNCOMMONSENSE, i.e., Ascent plus explicit negations, using the explicit negative triple-pattern `<?x NotAtLocation oven>` and is satisfied with top results (see Figure 3) such as *cheeseburger*, *sandwich*, and *salad*, all of which not requiring an oven. One can also see that if the user were to query positive-only Ascent (baseline following the Closed-world Assumption), 84% (6798) of all Ascent's concepts would be returned as plausible answers. The set is also unranked, hence the score=0, with many irrelevant answers, such as *newsroom* and *mathematics*. A second user is interested in sports that are *not* part of the Olympic games, they translate their information need by entering `<?x NotIsA olympic sport>`. UNCOMMONSENSE returns 96 sports, top ones including *croquet*, *rodeo*, and *kayaking*. On the other hand, positive-only Ascent returns 7320 unranked concepts, including *dialect*, *bread*, and *accountant*. UNCOMMONSENSE shows that even a simple negative triple-pattern with no positive conjunction or restriction of result-concept type, e.g., `<?x IsA sport. ?x NotIsA olympic sport>` return highly relevant concepts, unlike the baseline with mostly off-topic answers. This is especially helpful for users who are not familiar with the wording of object phrases a certain CSKB accepts, e.g., should they

[4] https://ascentpp.mpi-inf.mpg.de

Results from UNCOMMONSENSE & ASCENT++

{ X | <X ; NotAtLocation ; the oven> }

Found **784** results for x, showing the first 20 results:

#	Subject	Score
1	cheeseburger	0.75
2	sandwich	0.70
3	yorkshire pudding	0.70
4	wheat	0.69
5	turnip	0.68
6	salad	0.67
7	babka	0.66
8	appetizer	0.65
9	strudel	0.65
10	tortilla chip	0.62

Results from ASCENT++

{ X } \ { X | <X ; AtLocation ; the oven> }

Found **6798** results for x, showing the first 20 results:

#	Subject	Score
1	newsroom	0.00
2	erosion	0.00
3	mink	0.00
4	candidiasis	0.00
5	fishing rod	0.00
6	fix	0.00
7	fundamentalist	0.00
8	mathematics	0.00
9	balsam	0.00
10	pipe	0.00

Figure 3: Food that *doesn't* require an oven.

augment the query with `<?x IsA sport>`, `<?x IsA game>`, `<?x IsA activity>`, ..

4 RELATED WORK

Commonsense knowledge acquisition includes several large-scale projects including ConceptNet [13], ATOMIC [12], WebChild [15], Quasimodo [10], and Ascent [8]. Eventhough their main focus is positive knowledge, some of these projects allow the addition of negative statements. For example, ConceptNet [13] contains 6 pre-defined negative relations (NotIsA, NotCapableOf, NotDesires, NotHasA, NotHasProperty, and NotMadeOf), which we adapt in our demo. The portion of negative statements in its latest version is less than 2%. Quasimodo [10] contains 350k statements with negative *polarity*, yet many have quality issues, e.g., (elephant, NotCapableOf, quit smoking). On actively collecting informative commonsense negations, NegatER [11] proposes using graph-based triple corruption and fine-tuned LMs to discover meaningful negations. A detailed comparison to this work is in [3]. Other approaches that target informative negations in encyclopedic KBs, such as Wikidata [16] and Yago [14], include statistical inferences [1, 2] and text extractions [1, 6].

5 RESOURCES

For researchers interested in this topic, the web portal provides a JSON-formatted data dump (4.9 GB) of all 13.8 million negative statements inferred by UNCOMMONSENSE about Ascent's primary concepts. It can be downloaded from: https://uncommonsense.mpi-inf.mpg.de/download/.

6 CONCLUSION

In this demo, we present UNCOMMONSENSE, a web portal for inspecting informative negative statements about everyday concepts in a large-scale commonsense knowledge base.

ACKNOWLEDGMENTS

Funded by the German Research Foundation (DFG: Deutsche Forschungsgemeinschaft) - Project 453095897 - "Negative Knowledge at Web Scale".

REFERENCES

[1] Hiba Arnaout, Simon Razniewski, and Gerhard Weikum. 2020. Enriching Knowledge Bases with Interesting Negative Statements. In *AKBC*.
[2] Hiba Arnaout, Simon Razniewski, Gerhard Weikum, and Jeff Z. Pan. 2021. Negative Knowledge for Open-World Wikidata. In *WWW Companion*.
[3] Hiba Arnaout, Simon Razniewski, Gerhard Weikum, and Jeff Z Pan. 2022. UnCommonSense: Informative Negative Knowledge about Everyday Concepts. In *CIKM*.
[4] Giorgos Flouris, Zhisheng Huang, Jeff Z. Pan, Dimitris Plexousakis, and Holger Wache. 2006. Inconsistencies, Negations and Changes in Ontologies. In *AAAI*.
[5] Sven Hertling and Heiko Paulheim. 2017. WebIsALOD: Providing Hypernymy Relations Extracted from the Web as Linked Open Data. In *ISWC*.
[6] G. Karagiannis, I. Trummer, S. Jo, S. Khandelwal, X. Wang, and C. Yu. 2019. Mining an "anti-knowledge base" from Wikipedia updates with applications to fact checking and beyond. In *PVLDB*.
[7] Nayeon Lee, Belinda Z. Li, Sinong Wang, Wen-tau Yih, Hao Ma, and Madian Khabsa. 2020. Language Models as Fact Checkers?. In *ACL, FEVER workshop*.
[8] Tuan-Phong Nguyen, Simon Razniewski, Julien Romero, and Gerhard Weikum. 2022. Refined Commonsense Knowledge from Large-Scale Web Contents. *TKDE* (2022).
[9] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *EMNLP*.
[10] Julien Romero, Simon Razniewski, Koninika Pal, Jeff Z. Pan, Archit Sakhadeo, and Gerhard Weikum. 2019. Commonsense Properties from Query Logs and Question Answering Forums. In *CIKM*.
[11] Tara Safavi, Jing Zhu, and Danai Koutra. 2021. NegatER: Unsupervised Discovery of Negatives in Commonsense Knowledge Bases. In *EMNLP*.
[12] Maarten Sap, Ronan Le Bras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A. Smith, and Yejin Choi. 2019. ATOMIC: An Atlas of Machine Commonsense for If-Then Reasoning. In *AAAI*.
[13] Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. ConceptNet 5.5: An Open Multilingual Graph of General Knowledge. In *AAAI*.
[14] F. Suchanek, G Kasneci, and G. Weikum. 2007. Yago: A Core of Semantic Knowledge. In *WWW*.
[15] Niket Tandon, Gerard de Melo, Fabian Suchanek, and Gerhard Weikum. 2014. WebChild: Harvesting and Organizing Commonsense Knowledge from the Web. In *WSDM*.
[16] D. Vrandečić and M. Krötzsch. 2014. Wikidata: a Free Collaborative Knowledge base. *CACM* (2014).
[17] Ikuya Yamada, Akari Asai, Jin Sakuma, Hiroyuki Shindo, Hideaki Takeda, Yoshiyasu Takefuji, and Yuji Matsumoto. 2020. Wikipedia2Vec: An Efficient Toolkit for Learning and Visualizing the Embeddings of Words and Entities from Wikipedia. In *EMNLP*.