nature machine intelligence

Article

https://doi.org/10.1038/s42256-023-00726-1

Improving Wikipedia verifiability with AI

Received: 29 September 2022

Accepted: 4 September 2023

Published online: 19 October 2023

Check for updates

Fabio Petroni[1,6] ✉, Samuel Broscheit[2,6], Aleksandra Piktus[3], Patrick Lewis[3], Gautier Izacard[3,4], Lucas Hosseini[3], Jane Dwivedi-Yu[3], Maria Lomeli[3], Timo Schick[3], Michele Bevilacqua[1], Pierre-Emmanuel Mazaré[3], Armand Joulin[3], Edouard Grave[3] & Sebastian Riedel[3,5]

Verifiability is a core content policy of Wikipedia: claims need to be backed by citations. Maintaining and improving the quality of Wikipedia references is an important challenge and there is a pressing need for better tools to assist humans in this effort. We show that the process of improving references can be tackled with the help of artificial intelligence (AI) powered by an information retrieval system and a language model. This neural-network-based system, which we call SIDE, can identify Wikipedia citations that are unlikely to support their claims, and subsequently recommend better ones from the web. We train this model on existing Wikipedia references, therefore learning from the contributions and combined wisdom of thousands of Wikipedia editors. Using crowdsourcing, we observe that for the top 10% most likely citations to be tagged as unverifiable by our system, humans prefer our system's suggested alternatives compared with the originally cited reference 70% of the time. To validate the applicability of our system, we built a demo to engage with the English-speaking Wikipedia community and find that SIDE's first citation recommendation is preferred twice as often as the existing Wikipedia citation for the same top 10% most likely unverifiable claims according to SIDE. Our results indicate that an AI-based system could be used, in tandem with humans, to improve the verifiability of Wikipedia.

Wikipedia is one of the most visited websites[1], with half a trillion page views per year[2], and constitutes one of the most important knowledge sources today. As such, it is critical that any knowledge on Wikipedia is verifiable: Wikipedia users should be able to look up and confirm claims made on Wikipedia using reliable external sources[3]. To facilitate this, Wikipedia articles provide in-line citations that point to background material supporting the claim. Readers who challenge Wikipedia claims can follow these pointers and verify the information themselves[4–6]. However, in practice, this process can fail: a citation might not entail the challenged claim or its source might be questionable. Such claims may still be true, but a careful reader cannot easily verify them with the information in the cited source. Under the assumption that a Wikipedia claim is true, its verification is a two-stage process: (1) check the consistency of the existing source and (2) if that fails, search for new evidence.

As defined above, verification of Wikipedia claims requires deep understanding of language and mastery of online search. To what extent can machines learn this behaviour? This question is important from the perspective of progress in fundamental AI. For example, verification requires the ability to detect logical entailment in natural language and to convert claims and their context to the best search term for finding evidence—two long-standing problems that have been primarily investigated in somewhat synthetic settings[7–13]. It is equally important from a practical perspective. A machine verifier can assist Wikipedia editors by both flagging what citations might trigger failed verifications and suggesting what to replace citations with in case they currently do not support their respective claim. This can be significant: searching potential evidence and carefully reading the search results requires time and

[1]Samaya AI, London, UK. [2]Amazon Alexa AI, Tübingen, Germany. [3]FAIR, Meta, London, UK. [4]Inria and ENS, PSL University, Paris, France. [5]University College London, London, UK. [6]These authors contributed equally: Fabio Petroni, Samuel Broscheit. ✉e-mail: fabiopetroni@samaya.ai

https://doi.org/10.1038/s42256-023-00726-1

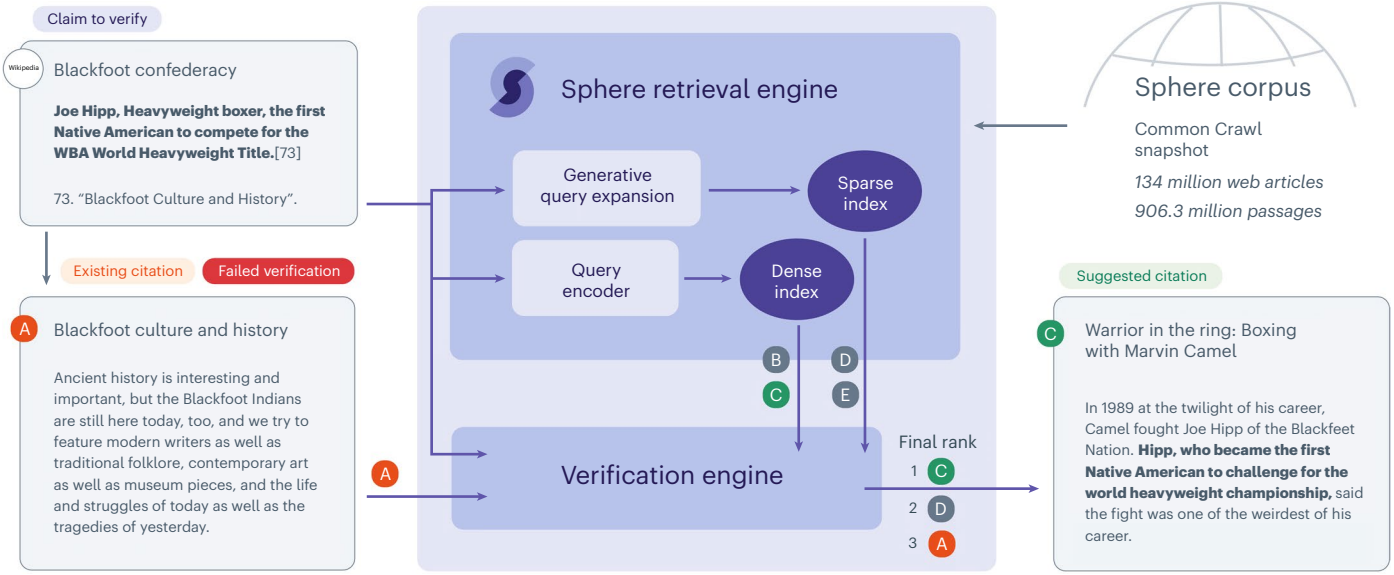

Fig. 1 | Overview of SIDE. The decision flow of SIDE from a claim on Wikipedia to a suggestion for a new citation is as follows: (1) the claim is sent to the Sphere retrieval engine, which produces a list of potential candidate documents from the Sphere corpus; (2) the verification engine ranks the candidate documents and the original citation with respect to the claim; (3) if the original citation is not ranked above the candidate documents, then a new citation from the retrieved candidates is suggested. Note that the score of the verification engine can be indicative of a potential failed verification, as the one reported in the example.

high cognitive effort. Integrating an AI assistant into this process could help to reduce both.

In this work we develop SIDE, an AI-based Wikipedia citation verifier. SIDE finds claims on Wikipedia that likely cannot be verified given the current citation, and for such, scans a web snapshot for an alternative. Its behaviour is learnt using Wikipedia itself: using a carefully curated corpus of English Wikipedia claims and their current citations, we train (1) a retriever component that converts claims and contexts into symbolic and neural search queries optimized to find candidate citations in a web-scale corpus; and (2) a verification model that ranks existing and retrieved citations according to how likely they might verify a given claim.

We evaluate our model using both automatic metrics and human annotations. To measure the accuracy of our system automatically, we check how well SIDE recovers existing Wikipedia citations in high-quality articles as defined by the Wikipedia featured article class. We find that in nearly 50% of the cases, SIDE returns exactly the source that is used in Wikipedia as its top solution. Notably, this does not mean the other 50% are wrong but that they are not the current Wikipedia source.

We also test SIDE's ability to be a citation assistant. In a user study we present existing Wikipedia citations next to the ones that SIDE produces. Users then assess the extent to which the presented citations support the claim, and which citation—from SIDE or Wikipedia—would be better for verification. Overall, more than 60% of the time users prefer SIDE's citations over Wikipedia's, which increases above 80% when SIDE associates a very low verification score to the Wikipedia citation.

System architecture

In Fig. 1, we provide a high-level overview of SIDE that shows an example of the decision flow given a Wikipedia claim. In the following, we briefly describe all major components of the system and how they interact with one another. We use the term 'claim' to refer to the sentence preceding a Wikipedia citation. The cited documents are represented as a list of passages.

The retrieval engine

Given a claim tagged as 'failed verification' by a human editor, or flagged by our verification engine, SIDE needs to retrieve a list of documents that support it. A human verifier would do so by (1) synthesizing a search query based on the claim's context; and (2) executing this query against a search engine. Fundamentally, SIDE 'learns' to do the same, using both sparse and dense retrieval sub-systems that we explain in more detail below. The claim's context is represented using the sentences preceding the citation, as well as the section title and the title of the enclosing Wikipedia article. We use Sphere[14], a web-scale corpus and search infrastructure for web-scale data, as a source of candidate web pages. Classic sparse and neural dense approaches are known to have complementary strengths[15] and hence we merge their results to produce the final list of recommended evidence.

Sparse retriever with generative query expansion. The sparse retrieval sub-system uses a sequence-to-sequence (seq2seq) model[15,16] to translate the citation context into query text, and then matches the resulting query—a sparse bag-of-words vector—on a BM25 index[17–21] of Sphere. We train the seq2seq model using data from Wikipedia itself: the target queries are web page titles of existing Wikipedia citations. The title of a web page or source often contains a summary or a condensed representation of the key information within the content. By using the titles to train a seq2seq query expansion model, we leverage this concise and meaningful information to generate better query expansions. In practice, we construct a query by concatenating the sentence preceding the citation, the Wikipedia title containing the claim and the generated web page title to be sent to BM25. Sparse retrieval methods rank documents by weighted lexical overlap and represent queries and documents as high-dimensional sparse vectors with dimensions corresponding to vocabulary terms. BM25 is by nature very successful in retrieving passages that require high lexical overlap, also for long tail names and words. The disadvantage for BM25 in this setting is that we do not know where the claim in the text in front of the citation is located, it could be just a short span of text, or the claim could be fragmented over multiple sentences and require references to the context of the Wikipedia article. Indeed, in a manual evaluation of a small sample (30 instances) we found that roughly one-third of the sentences had some kind of co-reference, which was crucial for understanding the claim. Empirically we found that only using the first sentence in front

https://doi.org/10.1038/s42256-023-00726-1

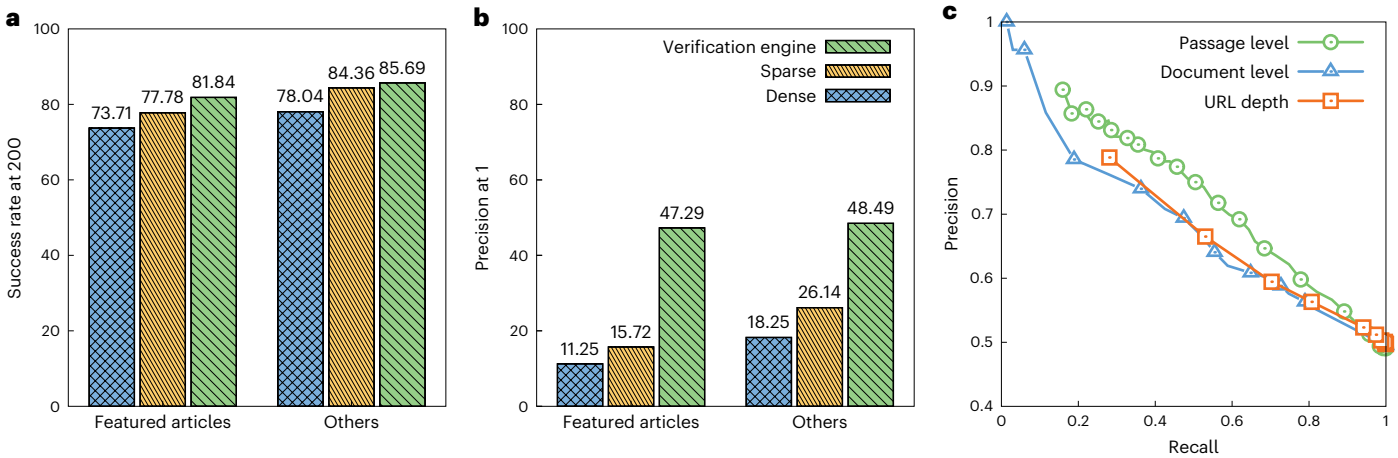

Fig. 2 | Automatic evaluation of SIDE components on the WAFER test set. **a**, Proportion of times our retrievers can surface the gold source among the top 200 results, for citations in featured and other Wikipedia articles. The verification engine bar (green) combines sparse and dense retrievers, 100 passages each. **b**, Accuracy in surfacing the gold source in first position, for citations in featured and other Wikipedia articles. The verification engine (green) takes as input a combination of 100 passages from the sparse and 100 from the dense retriever and reranks those. **c**, Precision versus recall in detecting citations marked as failed verification against citations in featured articles. We compare a passage versus a document-level approach for the verification engine and a baseline that simply uses the depth of the cited URL.

of the claim and also adding the Wikipedia article's title to the query did yield the best BM25 results.

Dense passage retriever. The dense retrieval sub-system is a neural network that learns from Wikipedia data to encode the citation context into a dense query vector[22–26]. This vector is then matched against the vector encodings of all passages in Sphere and the closest ones are returned. The context and passage encoders are trained such that the context and passage vectors of existing Wikipedia citation and evidence pairs are maximally similar[23]. Dense passage retrieval is a method that learns to embed queries and documents as low-dimensional dense vectors. The basic building block of dense passage retriever (DPR) is a BERT-like neural encoder, that consumes a sequence of tokens and predicts one dense vector. DPR consists of two such neural encoders, one for the query and one for a document's passage. DPR is then trained on a dataset with instances consisting of (query, correct document) tuples. The training objective is to maximize the similarity between the query vector and the passage vectors of a correct document using the inner product metric, and to minimize the similarity for incorrect documents. In contrast to BM25, DPR can learn which parts of the text are likely the important elements. Another advantage is that DPR is typically stronger in retrieving passages with rephrased versions of the claim.

The verification engine

Given a claim and possible evidence document, either on Wikipedia or proposed by the retrieval engine, a human would carefully evaluate to what extent the claim is supported by the provided evidence. This is the role played by our verification engine, a neural network taking the claim and a document as input, and predicting how well it supports the claim. Because of efficiency reasons, it operates on a per passage level and calculates the verification score of a document as the maximum over its per-passage scores. The verification scores are calculated by a fine-tuned BERT[27] transformer that uses the concatenated claim and passage as input. This architecture is akin to prior work for textual entailment in natural language inference[28].

The verification engine is optimized to rank claim–document pairs in order of verifiability rather than making verification versus failed-verification binary decisions. This is motivated by the way we envision SIDE's usage: we want to prioritize existing claims for humans to check by starting with those that are less likely to be supported by their current evidence, and to highlight recommended evidence for a given claim by starting with documents that are more likely to support it. To train the verification engine, we use an objective that rewards models when they rank existing Wikipedia evidence higher than evidence returned by our retrieval engine. Even though these training data could be noisy, given Wikipedia evidence might be of poor quality (a core motivation behind this work) and claims may have varying levels of veracity, we find that it still provides a meaningful signal on average. We test this claim empirically in the next section.

Data and training

Many components of our system, such as the dense retriever and the verification engine, are based on neural networks requiring examples to be trained and evaluated. We propose to leverage the scale of Wikipedia, and its millions of existing citations, to build WAFER, a training and evaluation dataset for our models (see an example citation from the dataset in the Supplementary Information). It should be noted that the obtained data are noisy, as existing citations could fail verification, and how to determine if it could be used to train our system is an interesting research question. Moreover, our system processes references at the passage level, while our training data corresponds to pairs of claims and documents. A passage is a chunk of text containing approximately one hundred words, a long document is represented as multiple passages. During cross-validation we split our data as claims at the article level to avoid potential test leakage into the training data—all claims in a single Wikipedia article are either in the training or evaluation split. With this strategy, we find that only approximately 2% of the claims in the evaluation set are repeated verbatim in the training set (exactly 83 claims).

We train the retriever and the verification engine using an expectation-maximization (EM) algorithm, modelling the passage containing the evidence as a latent variable. The verification engine employs a cross-encoder architecture based on a fine-tuned RoBERTa transformer that takes the concatenated claim and passage as input. It calculates verification scores for each claim-document pair on a per-passage level. The strategy used during the expectation (E) step accounts for the lack of supervision of a gold passage (only the gold URL is provided, which can contain multiple passages). The EM strategy identifies the highest-scoring positive passage from the n passages contained in a given source for each claim, after artificially creating some

https://doi.org/10.1038/s42256-023-00726-1

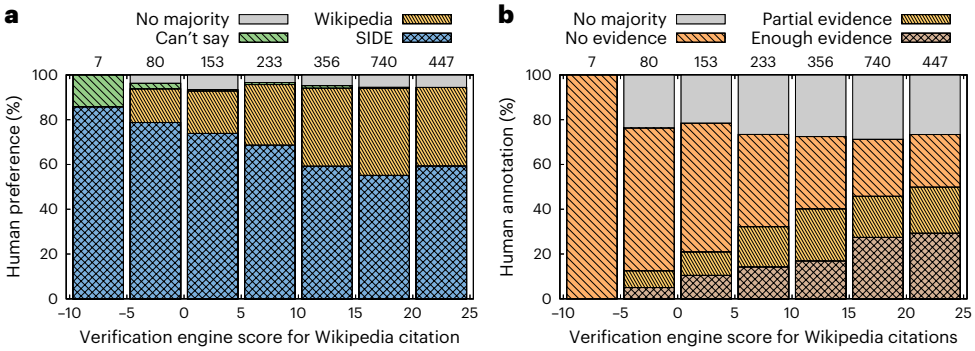

Fig. 3 | Crowd annotators' preference judgements. Crowd annotator evaluation for 2016 claims in the WAFER test set for which SIDE produces a citation with higher evidence score than the existing Wikipedia citation. We collect five annotations per claim and report majority vote results, bucketed according to the verification engine score associated with the existing Wikipedia citation (bucket size reported on top). **a**, Crowd annotators' preference for citations suggested by SIDE versus those on Wikipedia for a given claim, without knowing their identity. Fleiss's κ inter-annotator agreement = 0.2. **b**, Evidence annotations for Wikipedia citations: (1) enough to verify the claim; (2) the claim is only partially verified; (3) no evidence. Fleiss's κ inter-annotator agreement = 0.11.

negative examples and setting their corresponding scores to a very small negative number. The negative examples consist of references with incorrect claims. Our original data only contain positive examples of claims and references but mining negative examples is a standard solution. The selected positive examples are then used to create a mask that is utilized in the maximization (M) step. As a consequence, the existing Wikipedia evidence is ranked higher than the evidence returned by our retrieval engine. Even though training retrievers by mining negative examples is a standard solution, we introduce negative examples to train the verification engine and determine whether an existing reference is failing verification for a particular claim. Indeed, the problem of ranking a set of candidate documents for a particular claim is different from ranking existing pairs of documents and claims.

Evaluation and results

Evaluating the performance of our system is challenging because we cannot be certain that existing citations are always accurate and because of the lack of annotations for citations that fail verification. Therefore, we first evaluate the components of our system in isolation by addressing the following two questions: (1) given a Wikipedia claim, can our retrieval solutions surface the existing citation source from more than 100 million web articles? And (2) is our verification engine able to assign low scores to citations marked as failing verification in Wikipedia? After investigating these two questions, we conduct a large-scale human annotation campaign to evaluate the overall system. Additional details on experimental data and setting are provided in the Supplementary Information.

Retrieval evaluation

We report our results in Fig. 2 (additional results are found in the Supplementary Information). We note that the sparse retrieval solution outperforms the dense approach for retrieval from the web, which is consistent with previous observations[14]. However, we obtain our best overall 'success rate at 200' by combining 100 results from each given they are highly complementary[15] (see Fig. 2a)—this ensemble is what we use to retrieve passages to feed into the verification engine component. Notably, the verification engine component surfaces the original citation document in the highest ranked position nearly 50% of the time (see Fig. 2b).

In general, retrieving evidence for claims in featured articles is more challenging than for other claims in Wikipedia, for example, we observe a large difference of −7.0% (dense) versus −10.4% (sparse) 'precision at 1' between featured and non-featured articles. One hypothesis for this is that there exists an intrinsic popularity bias associated with featured content. Featured content might often correlate with popular topics, which in turn means that more sources on the web contain relevant information. By contrast, claims in more niche articles have much less coverage on the web and therefore are easier to find.

The verification engine model considerably boosts the accuracy of the retrieval component and almost levels the gap for featured articles, suggesting greater ability to identify evidence. This performance can be explained by its ability to leverage fine-grained language comprehension, since the model can directly compare the two texts using a cross-attention mechanism[29].

Detecting failed verification

Our goal in this analysis is to measure the degree to which the verification engine score can be used to detect whether a citation fails verification. To this aim, we rank the union of test citations in featured articles and test failed verification citations. An ideal system would place all failed verification at the bottom end of the ranked list and featured citations at the top. To compute the rank, we consider two different instantiations of the verification engine, which operate either at a passage or document level. As many failed citations include a link to an over-generic URL, we include a simple baseline related to the depth of a source URL. In the passage-level solution, we independently compute a score for each passage in a document with the verification engine and rank citations according to the maximum score. For the document-level approach, we feed as much text as possible (on average the first two or three passages) for the source document as input to a seq2seq model[16].

The resulting precision-recall curve is shown in Fig. 2c. Overall, the passage-level verification engine performs very well; if we only consider a conservative recall of 15%, for instance, approximately 90% are failed verification citations. Notably, these results are achieved without any explicit supervision on failed verification instances, given that the verification engine is trained only on positive examples. A document-level approach leads to worse results, mainly due to the impossibility of considering the whole document (given architectural constraints on maximum input size). Considering URL depth turns out to be a solid baseline. To further investigate this aspect, we study the distribution of depths for URLs in our data (Supplementary Information) and find that citations in featured articles tend to be deep (that is, specific) while citations marked as failed verification are usually shallow (that is, generic).

Evaluation of the final system

To test the performance of our final system, we perform a two-stage human assessment: (1) a large-scale crowd annotation campaign and

https://doi.org/10.1038/s42256-023-00726-1

Table 1 | Fine-grained human annotations for Wikipedia citations for which crowd annotators indicate no evidence

Supporting evidence availability	Proportion (%, n)
No evidence	41.3% (52)
Partial evidence	18.2% (23)
Full evidence in one passage	16.7% (21)
Full evidence in multiple passages	13.5% (17)
Evidence not in crawled text (for example, multimedia)	7.1% (9)
Paywall access	3.2% (4)

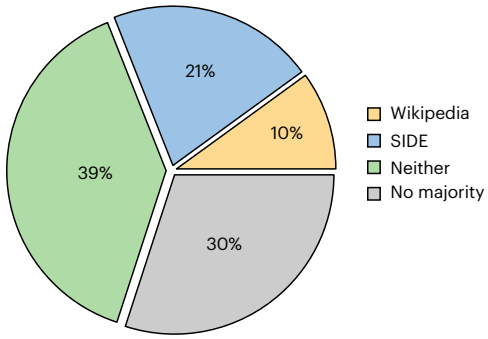

Fig. 4 | Wikipedia users' preference judgements. Wikipedia users annotations via our demo.

(2) a smaller scale fine-grained evaluation. First, we select claims in the test set for which SIDE outputs a citation source with a higher score than that on Wikipedia. We then ask crowd annotators to express their preference on which citation (SIDE's suggestion or Wikipedia's) better supports a given claim. Additionally, we ask them to assess whether a source contains enough evidence to support the claim, partial evidence (meaning that only parts of the claim are supported), or no evidence whatsoever. To keep the annotation load tractable, we use our verification engine component to select a single passage from each source and consider overlapping passages for Wikipedia sources to avoid cutting evidentiary sentences (exact instructions for the crowd annotators are included in the Supplementary Information). We conducted the annotation campaign on Amazon Mechanical Turk, paid $1.2 per annotation and collected five annotations per claim. A total of 192 crowd annotators participated in our campaign, their personal information is confidential. Note that the decision-making process of crowd annotators in our large-scale evaluation depends heavily on the content of a single passage from the cited sources, which may lead to a preference for secondary sources or simpler language over primary and high-quality sources. Regardless of these limitations, we were still interested in understanding whether the retrieved passages were of good syntactic quality, coherence and relevance to the claims, providing at least a basic measure of the system's ability to identify pertinent evidence.

Both preferences for SIDE-suggested sources (Fig. 3a) and Wikipedia evidence annotations (Fig. 3b) are proportional to the ranker score for the existing Wikipedia citation—the lower the score, the more preferences for SIDE and the less evidence found within Wikipedia. These results suggest that the ranker score might be a valid proxy for the presence of evidence in a citation, and might help in surfacing cases that require human attention. To verify the noise introduced by automatically selecting a single passage for each source, we conduct a control study on more than 500 sources where we ask annotators if they prefer the selected passage (that is, the top scored) with respect to a random one within the source. We find that for over 80% of the cases annotators prefer the selected passage, with an inter-annotator agreement of 0.27 (Fleiss's κ). Finally, to validate the crowd annotators' accuracy, we annotate more than 100 cases where evidence was not found in the Wikipedia citations. In Table 1, we find that sometimes the evidence is in the source but not within the crawled text (for example, multimedia content); other times, it is spread across multiple passages (which the system cannot detect, but that we plan to tackle in future work). Overall, more than 40% of the time, no evidence can be found in the reference to verify a claim.

To conduct an evaluation using more realistic conditions and gain a deeper understanding of the system's performance, we designed the smaller scale, fine-grained evaluation involving the Wikipedia community. This approach allowed us to closely assess the system with entire documents and real Wikipedia users, offering a more comprehensive and authentic analysis of the system's capabilities in finding the most appropriate sources to support the given claims. To this end, we build a demo of SIDE and engaged with the English-speaking Wikipedia community, asking users if they would use the citation on Wikipedia, the top-1 citation suggested by SIDE or neither to verify a given claim. We do not reveal the source of a citation in the user interface (that is, Wikipedia or SIDE), select claim-citation pairs on Wikipedia that are likely to fail verification (verifier score below 0) and allow access to the full text for each citation (instead of a single passage). Results (Fig. 4) reveal that SIDE can indeed select claim-citation pairs that fail verification—users selected the Wikipedia citation in only 10% of cases, compared with the 60% of citations for which either SIDE's recommendation or neither of the two were preferred. The observation that no majority was found in 30% of claims highlights the inherent difficulty of the task. Factors contributing to this difficulty include varying interpretations and preferences among users, ambiguity in claims, the diverse expertise, and levels of familiarity with citation quality and relevance. Note that 21% of the time SIDE provides a top-1 recommendation that is judged appropriate by Wikipedia users. We additionally conduct a sign test between SIDE and Wikipedia preferences resulting in a P value of 0.018. In total, 101 anonymous, authenticated Wikipedia users participated to our study, recruited over a set of channels including the WikiProject Reliability page (https://en.wikipedia.org/wiki/Wikipedia:WikiProject_Reliability) and the Wiki Research mailing list (wiki-research-l@lists.wikimedia.org). All users expressed their consent in using the data collected as part of a scientific publication. We collected a total of 220 annotations, with 3 annotations per claim on average and a Fleiss's κ inter-annotator agreement of 0.18.

Discussion

We introduce SIDE, an AI-based system for improving the verifiability of Wikipedia citations. Building on recent advances in natural language processing, we demonstrate that machines can help humans to find better citations, a task which requires understanding of language and mastery of online search. Although previous works[7–13] have shown the ability of neural networks to perform well on natural language understanding tasks, these results were mostly obtained for well specified tasks and on synthetic datasets. Here, we show similar results in a real-world scenario, implying noisier data and a more loosely defined task.

Our primary goal is not to surpass the state of the art, but rather to demonstrate that existing technologies have reached a stage where they can effectively and pragmatically support Wikipedia users in verifying claims. Although our results are promising, and our system can already be used to improve Wikipedia, alternative system architectures may outperform our current design, particularly in light of the tremendous advances made in the field over recent years in terms of both quality and speed. Furthermore, there exist a variety of future research directions worth pursuing. For instance, we only considered references corresponding to web pages, but Wikipedia also cites books, scientific articles and other kind of documents. These include other modalities than just text, such as images and videos. To fully assess the quality of Wikipedia references, SIDE needs to become multimodal.

Second, our system currently only supports the English language, whereas Wikipedia exists for more than two hundred languages. Utilizing SIDE in the monolingual setting for languages other than English poses some challenges due to varying degrees of data availability. Wikipedia corpora for low-resource languages tends to be sparser and noisier than the corresponding corpora for medium or high resource languages. Furthermore, the Wikipedia communities could be more or less active depending on the trustworthiness they assign to the resource as well as the difference in reference quality. Third, making SIDE multilingual raises interesting research questions, such as the capabilities of performing cross-lingual citation improvements: given a claim in one language, if the system cannot find good evidence in that particular language, can it find references in other languages?

Finally, our work currently assumes that Wikipedia claims are verifiable, and only improves the quality of the references for existing claims. A natural extension of our work would be to detect claims that are not verifiable, and flag them for review by human editors. This extension comes with challenges, since demonstrating that a claim is unverifiable usually requires finding contradicting evidence. Unfortunately, Wikipedia currently does not contain such information, and thus, training AI-based systems to perform this task is not straightforward. However, we believe that SIDE could be a first step towards surfacing unverifiable claims: if SIDE cannot find good evidence for a claim, it might be impossible to verify. We report one such example in the Supplementary Information, showing that a lack of good evidence from SIDE could be an indication of unverifiability.

We release all data, code and models. And we hope that this work could be used in a broader context, for example, helping humans to check facts. More generally, we believe that this work could lead to more trustworthy information online.

Data availability

The data used to train and evaluate our models are available at https://github.com/facebookresearch/side. In particular, the whole WAFER dataset can be downloaded at https://github.com/facebookresearch/side/blob/main/datasets/WAFER.md. Statistics for the WAFER dataset are available in the Supplementary Information.

Code availability

The code to reproduce our experiments is available at https://github.com/facebookresearch/sideunder MIT License and Zenodo (https://doi.org/10.5281/zenodo.8252866)[30].

References

1. Top websites ranking. *similarweb* https://www.similarweb.com/top-websites/ (2023). Accessed 28 September 2023.
2. Statistics. *Wikimedia* https://stats.wikimedia.org/#/all-projects/reading/total-page-views/normal|bar|2-year|~total|monthly (2023). Accessed 28 September 2023.
3. Verifiability. *Wikipedia* https://en.wikipedia.org/wiki/Wikipedia:Verifiability (2023). Accessed 28 September 2023.
4. Piccardi, T., Redi, M., Colavizza, G. & West, R. Quantifying engagement with citations on Wikipedia. In *Proc. Web Conference 2020* 2365–2376 (2020).
5. Lewoniewski, W., Węcel, K. & Abramowicz, W. Modeling popularity and reliability of sources in multilingual Wikipedia. *Information* **11**, 263 (2020).
6. Kaffee, L.-A. & Elsahar, H. References in Wikipedia: the editors' perspective. In *Companion Proc. Web Conference 2021* 535–538 (2021).
7. Bowman, S. R., Angeli, G., Potts, C. & Manning, C. D. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* 632–642 (Association for Computational Linguistics, 2015).
8. Wang, A. et al. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP* 353–355 (Association for Computational Linguistics, 2018).
9. Camburu, O. M., Rocktäschel, T., Lukasiewicz, T. & Blunsom, P. e-snli: Natural language inference with natural language explanations. *Adv. Neural Inf. Process. Syst.* **31** (2018).
10. Nie, Y. et al. Adversarial NLI: A New Benchmark for Natural Language Understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* 4885–4901 (Association for Computational Linguistics, 2020).
11. Pérez-Rosas, V., Kleinberg, B., Lefevre, A. & Mihalcea, R. Automatic detection of fake news. In *Proceedings of the 27th International Conference on Computational Linguistics* 3391–3401 (Association for Computational Linguistics, 2018).
12. Thorne, J., Vlachos, A., Christodoulopoulos, C. & Mittal, A. 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* Vol. 1, 809–819 (Association for Computational Linguistics, 2018).
13. Thorne J. & Vlachos, A. Automated fact checking: Task formulations, methods and future directions. In *Proceedings of the 27th International Conference on Computational Linguistics* 3346–3359 (Association for Computational Linguistics, 2018).
14. Piktus, A. et al. The web is your oyster - knowledge-intensive NLP against a very large web corpus. Preprint at https://doi.org/10.48550/arXiv.2112.09924 (2021).
15. Mao, Y. et al. Generation-augmented retrieval for open-domain question answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing* Vol. 1, 4089–4100 (Association for Computational Linguistics, 2021).
16. Lewis, M. et al. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* 7871–7880 (Association for Computational Linguistics, 2020).
17. Robertson, S. E. et al. *Okapi at TREC-3* (National Institute of Standards and Technology, 1995).
18. Baeza-Yates, R. et al. *Modern Information Retrieval* (Association for Computing Machinary, 1999).
19. Manning, C. D., Raghavan, P. & Schütze, H. *Introduction to Information Retrieval* Vol. 39 (Cambridge Univ. Press, 2008).
20. Robertson, S. & Zaragoza, H. *The Probabilistic Relevance Framework: BM25 and Beyond* (Now Publishers, 2009).
21. Lin, J. et al. Pyserini: A Python toolkit for reproducible information retrieval research with sparse and dense representations. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '21)* 2356–2362 (Association for Computing Machinery, 2021).
22. Wu, L., Petroni, F., Josifoski, M., Riedel, S. & Zettlemoyer, L. Scalable zero-shot entity linking with dense entity retrieval. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* 6397–6407 (Association for Computational Linguistics, 2020).
23. Karpukhin, V. et al. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* 6769–6781 (Association for Computational Linguistics, 2020).

24. Maillard, J. et al. Multi-task retrieval for knowledge-intensive tasks. In *Proc. 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing* 1098–1111 (Association for Computational Linguistics, 2021).
25. Oğuz, B. et al. Domain-matched pre-training tasks for dense retrieval. In *Findings of the Association for Computational Linguistics: NAACL 2022* 1524–1534 (Association for Computational Linguistics, 2022).
26. Luan, Y., Eisenstein, J., Toutanova, K. & Collins, M. Sparse, dense, and attentional representations for text retrieval. *Trans. Assoc. Comput. Ling.* **9**, 329–345 (2021).
27. Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proc. 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* 4171–4186 (Association for Computational Linguistics, 2019).
28. MacCartney, B. & Manning, C. D. Modeling semantic containment and exclusion in natural language inference. In *Proc. 22nd International Conference on Computational Linguistics (Coling 2008)* 521–528 (Coling 2008 Organizing Committee, 2008).
29. Seo, M. et al. Real-time open-domain question answering with dense-sparse phrase index. In *Proc. 57th Annual Meeting of the Association for Computational Linguistics* 4430–4441 (Association for Computational Linguistics, 2019).
30. Petroni, F. et al. Improving Wikipedia verifiability with AI. *Zenodo* https://doi.org/10.5281/zenodo.8252866 (2022).

Acknowledgements

The authors would like to greatly thank M. Redi and D. Saez-Trumper from the Wikimedia Foundation for their invaluable help and support throughout the project; R. Nkama for helping set up our annotation interface; and all Wikipedia users who helped us evaluate SIDE. The work of F.P. and S.B. was conducted while they were affiliated with FAIR, Meta.

Author contributions

F.P. and S.R. conceived of the presented idea. F.P. collected the data for the experiments. F.P., S.B. and S.R. conceived and planned the experiments. S.B. trained the models. F.P. and S.B. carried out the experiments. F.P. carried out the crowd annotation campaign. L.H. led the development of the demo, with help from A.P., P.L., J.D.-Y., M.L., T.S. and S.R.; F.P. engaged with the Wikipedia community. All authors contributed to the interpretation of the results and reviewing the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42256-023-00726-1.

Correspondence and requests for materials should be addressed to Fabio Petroni.

Peer review information *Nature Machine Intelligence* thanks Lucie-Aimée Kaffee and the other, anonymous, reviewer(s) for their contribution to the peer review of this work. Primary Handling Editor: Jacob Huth, in collaboration with the *Nature Machine Intelligence* team.