Searching for Evidence of Scientific News in Scholarly Big Data

Md Reshad Ul Hoque[1], Dash Bradley[2], Chiman Kwan[3],Agnese Chiatti[4], Jiang Li[1] and Jian Wu[2]
[1]Electrical & Computer Engineering, Old Dominion University, Norfolk, VA, USA
[2]Computer Science, Old Dominion University, Norfolk, VA, USA
[3]Applied Research LLC, Rockville, MD, USA
[4]Knowledge Media Institute, The Open University, UK
{mhoqu001,jli,j1wu}@odu.edu,chiman.kwan@arllc.net

Abstract

Public digital media can often mix factual information with fake scientific news, which is typically difficult to pinpoint, especially for non-professionals. These scientific news articles create illusions, misconceptions and ultimately influence the public opinion, with serious consequences even at a much broader, societal scale. Yet, the existing solutions aimed at automatically verifying the credibility of news articles are still unsatisfactory. We propose to verify scientific news by retrieving and analyzing its most relevant source papers from an academic digital library (DL), e.g., arXiv. Instead of querying news keywords or regular named entities, we query domain knowledge entities (DKEs) extracted from the given scientific news article. For each DKE, we retrieve a list of candidate scholarly papers. We then design a function to rank candidate papers to select the most relevant scholarly paper. After exploring various representations, we found that the term frequency-inverse document frequency (TF-IDF) representation with cosine similarity could outperform other baseline models. This result demonstrates the efficacy of using DKE to retrieve scientific papers which are relevant to a specific news article. It also indicates that word embedding may not be the best document representation for domain knowledge retrieval tasks. Our method is fully automated and can be effectively applied to detect fake and misinformed news across many scientific domains.

CCS Concepts

• **Information systems** → *World Wide Web*; *Content ranking*; *Information retrieval*; • **Computing methodologies** → *Artificial intelligence*;

Keywords

Fake news, Domain knowledge entity, Web API, Embedding

ACM Reference Format:
Md Reshad Ul Hoque[1], Dash Bradley[2], Chiman Kwan[3],Agnese Chiatti[4], Jiang Li[1] and Jian Wu[2]. 2019. Searching for Evidence of Scientific News in Scholarly Big Data. In *Marina del Rey '19: ACM Symposium on Neural Gaze Detection, June 03–05, 2018, Woodstock, NY*. ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/1122445.1122456

K-CAP '19, November 19–22, 2019, Marina del Rey, CA

ACM ISBN 978-1-4503-9999-9/18/06…$15.00
https://doi.org/10.1145/1122445.1122456

1 Introduction

The phenomenon of spreading scientific misinformation to the public is not new. The discrepancies between public opinion and the scientific consensus on topics such as vaccine safety or climate change have existed for a long time [15]. Nowadays, people receive a large amount of information through social media and news portals through subscriptions and recommendations. Statistics indicate that 62% of U.S. adults were exposed to news on social media in 2016 as opposed to only 49% in 2012[1]. Through the widespread social media and mobile devices, misleading and fabricated news becomes easier, leading to illusion, confusion and, in some extreme cases, even violence.

Different from political news, people, in general, have higher confidence in scientific news as it is *supposed* to be backed by scientific theorems, experiments, and observations. However, news audiences may not have necessary *domain knowledge* to discriminate scientific news that contains exaggerated, distorted, or misinterpreted assertions that lack scientific evidence. For example, a news web site called newswatch33.com published an article that claimed *NASA confirms earth will experience 15 days of complete darkness in November 2015*. This news was a hoax [2] but it became viral on digital media (Facebook) and made many people panic [3]. To prevent this type of scientific news from being disseminated beyond control, and mitigate its potential detriment to society, we need a mechanism to check its credibility (truthfulness).

Fact-checking services are already available at several websites, e.g., *www.snopes.com* and *www.factcheck.org*, however, these services trace provenance through laborious, fully manual web browsing, and cross-verification procedures. There are computational models developed to automatically detect fake news [16]. The majority of these models rely on news contents such as the author/publisher, headline, body text, images, and videos. Computationally oriented fact-checking methods try to solve two major issues: identifying check-worth claims and discriminating the credibility of fact claims. There is a lack of automatic fact-checking algorithms that can assess the credibility of scientific news articles using provenance scholarly papers.

The above task can be accomplished in two steps. First, given a vast amount of scholarly papers, how can one find the *most relevant* ones pertaining to a scientific news article?. Second, we need to examine and compare the statements in the research paper and the news article to judge whether the latter are supported or not supported by the former. In this work, we focus on the first task.

[1]http://www.journalism.org/2016/05/26/news-use-across-social-media-platforms-2016/
[2]https://www.snopes.com/fact-check/15-days-darkness-november
[3]https://bit.ly/2Tfmp0I

Several challenges are identified. (1) There is not a digital library search engine API that can provide search service of a *complete* dataset of research papers, especially the most updated ones. (2) For a particular search API such as the *arXiv.org*, the ranking mechanism is proprietary. The results returned can be query dependent. Therefore, the key challenge is to find the *signature terms* that coexist in the scientific news article and certain research papers. We propose to use domain knowledge entities (DKEs) extracted from a news article as search engine queries. (3) The results returned by search engines are only candidate papers. Thus, it is necessary to design a function to rank candidate papers depending on their similarity to the news article.

2 Related Work

Fake news detection can be formalized into a classification problem, typically tackled through two major approaches [5]. In *linguistic approaches*, deceptive messages are extracted and analyzed with the associate language patterns. Bag of words, deep syntax, semantic analysis are instances of models for these approaches. *Network-based approaches* use network information such as message metadata, structured knowledge queries, and users reactions to detect fake news [5]. News content and social context are the two major sources of features. For example, a ranking approach was proposed to detect fake tweets with multimedia contents [1]. The proposed system calculates the legitimacy score of a tweet with respect to another tweet in the same topic. An automatic rumor verification algorithm was proposed using a new set of features that capture the semantic similarity between the rumors and the external information. In [17], these results were further improved, by applying transfer learning. A framework for misleading articles verification on Facebook was proposed [12], which uses regular expressions to group sentences with an intransitive verb and sentences ending with question marks. However, none of them verified news articles with scholarly papers.

In a recent article closely related to our work, a method was proposed to recommend research articles in PubMed to consumers of online vaccine information [7]. Articles are ranked using an approach called canonical correlation analysis (CCA). However, this approach failed to beat the baseline, which is a simple, TF-IDF based method. The best CCA-based approach ranks the matching source articles first for 14% and in the top 50 for only 38%.

Our method is different from existing works in the following aspects. First, instead of using regular keyphrase extractors, such as TextRank, we will use the DKE extractor that extracts entities reflecting domain knowledge , and, therefore, representing better signature phrases to query relevant research papers. Second, our approach capitalizes on a search API for news articles spanning across multiple domains, instead of relying just on a local digital repository.

3 Methods

3.1 Domain Knowledge Entity (DKE)

Before describing the system, we introduce a special type of entity called domain knowledge entity or DKE. DKEs are noun phrases that deliver domain knowledge. They are different from regular named entities such as people, organization, and locations, extracted by commonly used Name Entity Recognition (NER) tools [8]. Those named entities deliver *general* knowledge that usually does not require domain knowledge to understand. DKEs are also different from keyphrases. Keyphrases provide a *top-level* understanding of the entire article. DKEs provide a *fine-grained* description of the main relevant objects in an article. Statistically, a research article may contain 3-10 keyphrases, but one paragraph may contain 10 or more DKEs. For example, in *To spot these two new wanderers, Warsaw University astronomers used a technique called gravitational microlensing.*, *Warsaw University* is a regular named entity referring to an organization while *gravitational microlensing* is a DKE.

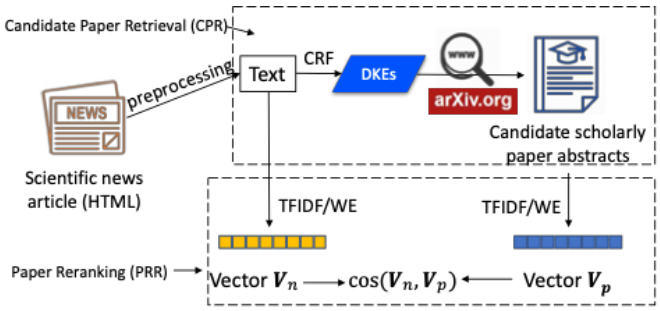

Figure 1: The top level architecture of our system using DKEs to query a search engine API, retrieve and rerank candidate papers. The ranking scores are calculated by cosine similarities between vector representations of paper abstracts V_p and news article V_n.

3.2 System Architecture

The system depicted in Figure 1 consists of 3 modules: preprocessing, candidate paper retrieval (CPR), and paper reranking (PRR).

In the **preprocessing** module, the HTML page of a news article is first downloaded and the body text is extracted from the HTML file. After that, the news article is tokenized and part-of-speech (POS)-tagged using the Stanford POS tagger [8].

In the **CPR module**, we extract DKEs from the news article body text using the method described below. The motivation to use DKEs instead of regular named entities or keyphrases to search relevant scholarly papers is that they are better at representing the research article and thus can quickly narrow down search results to a relatively small space. We query extracted DKEs for scholarly papers against an online search engine API. In this work, we query the search API on arXiv.org, a popular platform for researchers to submit pre-printed papers before/after they are officially published. It has a vast database of more than 1 millions research papers in a variety of fields focusing on physics, mathematics, computer science, biology, etc. The digital library offers a free API, through which we can get scholarly paper metadata such as titles, abstracts, authors, submission dates, and arXiv URLs. To maximize the recall, each query contains only 1 DKE and we retrieve the top 10 results for each query. Because search results of multiple queries may overlap, we remove duplicate papers by titles and authors to merge all results to the final candidate list of papers.

In the **PRR module**, abstracts of candidate papers and the given news report are represented by vectors. Candidate papers obtained in the CPR module are sorted by their cosine similarities to the news article. The candidate paper with the highest score is taken as the most relevant paper to the news article. The way to construct the vector representation can obviously have a huge impact

on the ranking results. We investigate the basic TF-IDF and the pre-trained word embedding (WE) models. WE is a vector representation method that converts texts to a lower dimension dense vector [11]. WE has already shown promising result in information retrieval system [14] in learning-to-rank tasks [19].

3.3 Data

We collect 50 scientific news articles mostly from ScienceAlert, ScienceNews, EurekAlert and forbes[4]. Each article includes a hyperlink to the supportive scholarly papers. For example, a scientific news article in ScienceAlert titled *A Physicist Has Proposed a Pretty Depressing Explanation For Why We Never See Aliens* talked about aliens. The claims of this article were backed by a scholarly paper titled *"First in, last out" solution to the Fermi Paradox*. The average length of news articles is about 900–1000 words. The collected articles are from a variety of domains such as astronomy, biology, environment, computer science, and medicine.

The HTML files were preprocessed. Many HTML files were from the same website and thus followed the same templates, so it is relatively straightforward to build a custom parser to extract the desired fields from those files, such as the authors, dates published, and news titles. All of the information was stored in JSON format. Then we remove stop words, white space, special symbols, and special characters (#,[],$, etc.) using regular expressions. As a prerequisite for DKE extraction, regular text normalization is performed such as sentence segmentation, tokenization, and POS-tagging.

We used the SemEval2017 competition Task 10 dataset to train our DKE extractor [2]. This dataset contains about 350 paragraphs extracted from ScienceDirect scholarly articles in Computer Science, Material Science , Physics and used for training our DKEs.

3.4 Domain Knowledge Entity Extraction

The DKE extractor used in this paper was adopted partially from the domain entity extractor described in [18], which proposed a hybrid architecture using non-sequential and sequential classifiers. As a preliminary study, we adopt the sequential component in which a conditional random field (CRF) model was trained based on labeled paragraphs in our training dataset. There are 9 features such as the current token, tokens within a boundary of 2 tokens, POS tags, and the word suffices.

Because the CRF model above was trained on a corpus of scientific papers, when applied on news articles, the model extracts regular named entities. For example, "Neptune","UK" are both extracted as DKEs by the CRF model but UK, a country name, is a regular named entity. We use the NLTK-NER package to identify these named entities and excluded them from the list. This effectively increases the precision of DKE extraction. On the other hand, we also notice that the CRF model misses a fraction of DKEs. For example, a news article titled *Scientists Have Connected The Brains of 3 People, Enabling Them to Share Thoughts* claimed that "using BrainNet algorithm 3 people can share their thoughts through EEG". Here "BrainNet" is the pivotal DKE but is failed to be extracted, effectively decrease the recall. To cope with this issue, we added the top 25% of named entities with the highest IDF values extracted by NLTK-NER. The IDF values of these named entities are calculated using abstracts from Web of Science (WoS). In the example above, "BrainNet" was added back to the list because it has a high IDF.

3.5 Candidate Paper Retrieval (CPR)

In the CPR module, we query the arXiv search API for relevant candidate papers using the extracted DKEs. To demonstrate the efficacy of DKEs, we compare DKEs with two baselines.
(1) **TextRank** is a graph-based algorithm to extract keyphrases. The idea was inspired by the Google's PageRank algorithm [9](2) **Stanford NER** is an annotator of the Stanford CoreNLP toolkit [8]. The output contains named entities of 3 types: Person, Organization, and Location.
The top 10 results returned are retrieved for each query. For each article, there are 35-40 DKEs extracted (after adjustment by NLTK-NER results), so roughly 350-400 candidate scholarly papers that are ranked in the PRR module.

3.6 Paper Reranking (PRR)

In the PRR module, a ranking function is designed to compare the overall semantic similarity of a given news article with candidate papers retrieved in the CPR module. We use abstracts for each paper because the full text is not always accessible. Each pre-processed news article can be represented as vectors $\mathbf{V}_n$ using TF-IDF or pre-trained word embedding (WE) models. Similarly, each candidate paper can be represented using a vector $\mathbf{V}_p$. We then calculate the cosine similarity between $\mathbf{V}_n$ and $\mathbf{V}_p$ (Figure 1).
(1) **TF-IDF**. A document is represented by a sparse vector containing $|V|$ elements, $|V|$ being the vocabulary size of a *retrieval corpus*. A *retrieval corpus* contains a news article and the abstracts of all candidate papers. The TF-IDF value for each term is calculated based on the *retrieval corpus* it belongs to.
(2) **Word2vec** was trained on 100 billion Google News words [10]. At first, sentences are tokenized, and then each token is represented as a 300 dimensional vector using the pre-trained Skip Gram model. The representation of the document is obtained by calculating the arithmetic average of vectors of all tokens.
(3) **GloVe** learns word representations from the co-occurrence matrix. It is trained on 6 billion tokens [13]. We apply GloVe in a similar way as Word2vec. The dimension of each vector is 50.
(4) **BERT** was trained on Wikipedia and Book Corpus dataset consisting of 10,000 different genres and its vector representation is context-dependent [6]. We apply BERT in a similar way as Word2vec. The dimension of each vector is 768.
(5) **USE** (Universal sentence encoder) was trained on the SNLI corpus consists of 570k human-written English sentence pairs and the semantic similarity between each pair [3]. A document is represented by the arithmetic average of vectors for all sentences. The dimension of each vector is 512.

3.7 Evaluation

Three methods are used for evaluation. The first is the mean reciprocal rank (MRR), defined as: $\mathrm{MRR} = \left(\sum 1/rank(i)\right)/Q$, in which Q is the total number of queries and $rank(i)$ is position of the first relevant item. MRR assumes there is only one relevant document in search results of each query. However, there are queries that return multiple relevant documents, so we use the average normalized discounted cumulative gain (NDCG) with the binary graded relevance (0 or 1). We also calculate the percentages of relevant documents

[4]ScienceAlert: https://www.sciencealert.com/; ScienceNews: https://www.sciencenews.org/; EurekAlert: https://www.eurekalert.org/; forbes:www.forbes.com

within the top 1, 5, 20, and 50 results returned. We also measure the running time for each method if possible.

Table 1: A comparison of models investigated.

Repres. method	Query Type	% of relevant papers at Ranks 1/5/20/50	MRR	Average NDCG	Time (sec)
Baseline[1]	–	14%/–/–/38%	–	–	–
TF-IDF	KP[2]	18%/22%/34%/34%	0.23	0.27	0.03
TF-IDF	NE[3]	26%/32%/36%/36%	0.29	0.34	0.03
TF-IDF	**DKE**	**38%/54%/66%/ 72%**	**0.47**	**0.57**	**0.03**
Word2vec	DKE	22%/44%/58%/64%	0.34	0.42	0.33
GloVe	DKE	12%/32%/36%/48%	0.22	0.29	6.46
BERT	DKE	6%/12%/28%/46%	0.13	0.24	108.7
USE	DKE	30%/48%/64%/68%	0.39	0.43	1611

[1] Quoted from [7]. The specific corpus used is not available, making it impossible to make a fair comparison.
[2] Keyphrases extracted using TextRank [9].
[3] Named entities extracted using Stanford CoreNLP. [8].

4 Results and Discussions

In the experiment results (Table 1), it is evident that TF-IDF representations of papers returned by querying DKEs achieve the best performance, followed by the sentence encoder (USE). However, the latter takes a dramatically long time, so it is impractical to build a search system based on it. DKE based Word2vec finishes the ranking in subsecond time, but the top rank fractions, MRR, and average NDCG are lower than the DKE based TF-IDF method. The BERT representation takes the second-longest time but achieves the worst performance in 3 evaluation metrics.

For certain news articles[5], DKE based TF-IDF ranking fail to rank relevant scholarly paper at the top position (false negatives), but embedding based methods (Word2Vec, USE) were able to. These articles usually paraphrase the terms used in the paper cited. Therefore, embedding based methods can infer its semantic meaning from surrounding contexts. There are news articles that no methods were able to rank at top positions. This is likely because the scientific news is written without using any DKEs or the DKEs extracted exist in many scientific papers of the same topics. In certain cases, although the DKE based TF-IDF ranking failed to retrieve the exact scientific papers linked to the news article, it retrieves articles that are very similar to the articles in the ground truth. These cases, which potentially increases the NDCG, are not counted in our evaluation.

5 Conclusion and Future Work

In this work, we attempted to build a system to find the missing links between scientific news articles and scholarly papers. The results indicate that the retrieval model using DKEs as queries significantly outperforms query models using general keyphrase terms or named entities. This is likely because WE tends to find and boost the ranks of papers that are semantically similar but are not exactly linked to the news article content. The DKE-based TF-IDF model has the potential to find out the source scholarly papers more accurate and faster.

[5]https://www.sciencealert.com/this-ai-tries-to-guess-what-you-look-like-based-on-your-voice

Our methods can be improved in the following aspects. (1) We limited the search scope to arXiv papers, which focuses on a limited scope of domains. We will query general search engine APIs (e.g., Bing), which can increase the domain diversity of the candidate papers. (2) The DKE extractor extracts unwanted domain entities that ultimately hampers our model performance. An improved model will combine a gazetteer with the attentive LSTM model, proven to be powerful in capturing answers to questions from free text [4]. Built on top of this work, we will develop models to compare the consistency of assertions in news articles and retrieved papers.

References

[1] Taruna Agrawal, Rahul Gupta, and Shrikanth Narayanan. 2017. Multimodal detection of fake social media use through a fusion of classification and pairwise ranking systems. In *EUSIPCO, 2017*. IEEE, 1045–1049.
[2] Isabelle Augenstein, Mrinal Das, Sebastian Riedel, Lakshmi Vikraman, and Andrew McCallum. 2017. SemEval 2017 Task 10: ScienceIE - Extracting Keyphrases and Relations from Scientific Publications. In *Proceedings of the 11th International Workshop on Semantic Evaluation*. 546–555.
[3] Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Brian Strope, and Ray Kurzweil. 2018. Universal Sentence Encoder for English. In *Proceedings of EMNLP 2018: System Demonstrations*. 169–174.
[4] Danqi Chen. 2018. *Neural Reading Comprehension and Beyond*. Ph.D. Dissertation. Stanford University.
[5] Niall J Conroy, Victoria L Rubin, and Yimin Chen. 2015. Automatic deception detection: Methods for finding fake news. In *Proceedings of the 78th ASIS&T Annual Meeting: Information Science with Impact: Research in and for the Community*. American Society for Information Science, 82.
[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of NAACL-HLT 2019*. 4171–4186.
[7] Eliza Harrison, Paige Martin, Didi Surian, and Adam G Dunn. 2019. Recommending research articles to consumers of online vaccination information. *arXiv preprint arXiv:1904.11886* (2019).
[8] Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit. In *ACL System Demonstrations*. 55–60.
[9] Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing Order into Text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing , EMNLP 2004*. 404–411.
[10] Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013. Linguistic Regularities in Continuous Space Word Representations. In *Processings of 2013 Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics*. 746–751.
[11] Bhaskar Mitra, Nick Craswell, et al. 2018. An introduction to neural information retrieval. *Foundations and Trends® in Information Retrieval* 13, 1 (2018), 1–126.
[12] Rehana Moin, Khalid Mahmood Zahoor-ur Rehman, Mohammad Eid Alzahrani, and Muhammad Qaiser Saleem. 2018. Framework for Rumors Detection in Social Media. *Framework* 9, 5 (2018).
[13] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global Vectors for Word Representation. In *Proceedings of EMNLP 2014*, Alessandro Moschitti, Bo Pang, and Walter Daelemans (Eds.). 1532–1543.
[14] Dwaipayan Roy, Debasis Ganguly, Sumit Bhatia, Srikanta Bedathur, and Mandar Mitra. 2018. Using Word Embeddings for Information Retrieval: How Collection and Term Normalization Choices Affect Performance. In *Proceedings of the 27th CIKM*. 1835–1838.
[15] Dietram A Scheufele and Nicole M Krause. 2019. Science audiences, misinformation, and fake news. *Proceedings of the National Academy of Sciences* (2019), 201805871.
[16] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake news detection on social media: A data mining perspective. *ACM SIGKDD Explorations Newsletter* 19, 1 (2017), 22–36.
[17] Weiming Wen, Songwen Su, and Zhou Yu. 2018. Cross-Lingual Cross-Platform Rumor Verification Pivoting on Multimedia Content. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 3487–3496.
[18] Jian Wu, Sagnik Ray Choudhury, Agnese Chiatti, Chen Liang, and C Lee Giles. 2017. HESDK: A hybrid approach to extracting scientific domain knowledge entities. In *Proceedings of the 17th JCDL*. 241–244.
[19] Chenyan Xiong, Russell Power, and Jamie Callan. 2017. Explicit Semantic Ranking for Academic Search via Knowledge Graph Embedding. In *Proceedings of the 26th International Conference on World Wide Web*. 1271–1279.