Data Void Exploits: Tracking & Mitigation Strategies

Miro Mannino
miro.mannino@nyu.edu
NYU Abu Dhabi
Abu Dhabi, United Arab Emirates

Junior Garcia
juniorgarcia@nyu.edu
NYU Abu Dhabi
Abu Dhabi, United Arab Emirates

Reem Hazim
rh3015@nyu.edu
NYU Abu Dhabi
Abu Dhabi, United Arab Emirates

Azza Abouzied
azza@nyu.edu
NYU Abu Dhabi
Abu Dhabi, United Arab Emirates

Paolo Papotti
papotti@eurecom.fr
EURECOM
Biot, France

ABSTRACT

A data void is a gap in online information, providing an opportunity for the spread of disinformation or a *data void exploit*. We introduce lightweight measures to track the progress of data void exploits and mitigation efforts in two contexts: Web search and Knowledge Graph (KG) querying. We use case studies to demonstrate the viability of these measures as data void trackers in the Web search context. To tackle data voids, we introduce an adversarial game model involving two agents: a disinformer and a mitigator. Both agents insert content into the information ecosystem to have their narrative rank higher than their counterpart in search results. At every turn, each agent chooses which content to deploy within their resource constraints, mimicking real-world situations where different entities have varying levels of influence and access to resources. Using simulations of this game, we compare and evaluate different mitigation strategies to recommend ones that maximize mitigation impact while minimizing costs.

CCS CONCEPTS

• **Security and privacy** → **Human and societal aspects of security and privacy**.

KEYWORDS

misinformation, data void, web search, knowledge graph, exploit

ACM Reference Format:
Miro Mannino, Junior Garcia, Reem Hazim, Azza Abouzied, and Paolo Papotti. 2024. Data Void Exploits: Tracking & Mitigation Strategies. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management (CIKM '24), October 21–25, 2024, Boise, ID, USA*. ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3627673.3679781

1 INTRODUCTION

Search begins with keywords. When there is a dearth of information online that is relevant to the keywords, we are in a data void [15]. Data voids are not inherently problematic. A random string such as "xydea8gya8g7" or "battery equator jargon apple" may return no results or a few pages that are irrelevant with respect to the search keywords. We do not care about such information voids. Traveling back in time to the 2016 US Elections, the seemingly random set of keywords "satan pizza hillary children" would have brought users to the carefully constructed content around the pizzagate conspiracy theory, which implicated Hillary Clinton in a child sex ring run from a pizza restaurant [2]. In the short time frame from content creation to its coverage and debunking in main stream media, searchers were directed to the problematic disinformation. We care about these data voids.

Disinformers have capitalized on the presence of data voids and the operation of search engines to drive information seekers to their narratives. Tripodi outlines how political agents have exploited the information consumption habits of Evangelical groups in the US to push right-wing agendas on taxation, liberal corruption, deep-state conspiracies, etc. [41] As information seekers self-discover the content by searching for specific keywords on search engines, they deem it authentic as it was actively found rather than passively shared with them [41]. *Thus, an effective data void exploit can have deep and lasting impact on non-suspecting users.*

To understand how a data void exploit occurs and how a mitigation response works, consider the keyword search query in Figure 1, circa 2008. Disinformers manipulate search results for a fresh data void: "Obama born Kenya." They add web pages (red content) with high search relevance for the void's keywords. They may also deploy these pages in sites with high PageRanks [32] to boost the ranking of their narrative. The exploit is not limited to Web search results. Search engines rely on structured data, stored in Knowledge Graphs (KGs), to extend search beyond matching keyword queries to pages, to provide users with faster and richer results [17, 18]. Since KGs suffer from incompleteness [44], they depend on continual data curation and augmentation for accuracy and coverage of new facts [31]. This incompleteness allows attackers to inject fresh facts that "fill up" the data void. In Figure 1, disinformers further manipulate search results by adding KG facts (red edges) with high relevance for the void's keywords. Mitigators respond to such attacks by also filling up the void with counter-content (green) to rank their narrative higher in search results.

There is no easy "fix" for data voids and search platforms and mitigators need to work together to "identify vulnerabilities and respond to attacks" [15]. Their eye-opening report, however, leaves much to be determined as to how exactly can mitigators monitor and respond to data voids. Current search platforms either have limited bandwidth or awareness to mitigate all forms of disinformation, especially those beyond their regional legal liability.

CIKM '24, October 21–25, 2024, Boise, ID, USA

ACM ISBN 979-8-4007-0436-9/24/10
https://doi.org/10.1145/3627673.3679781

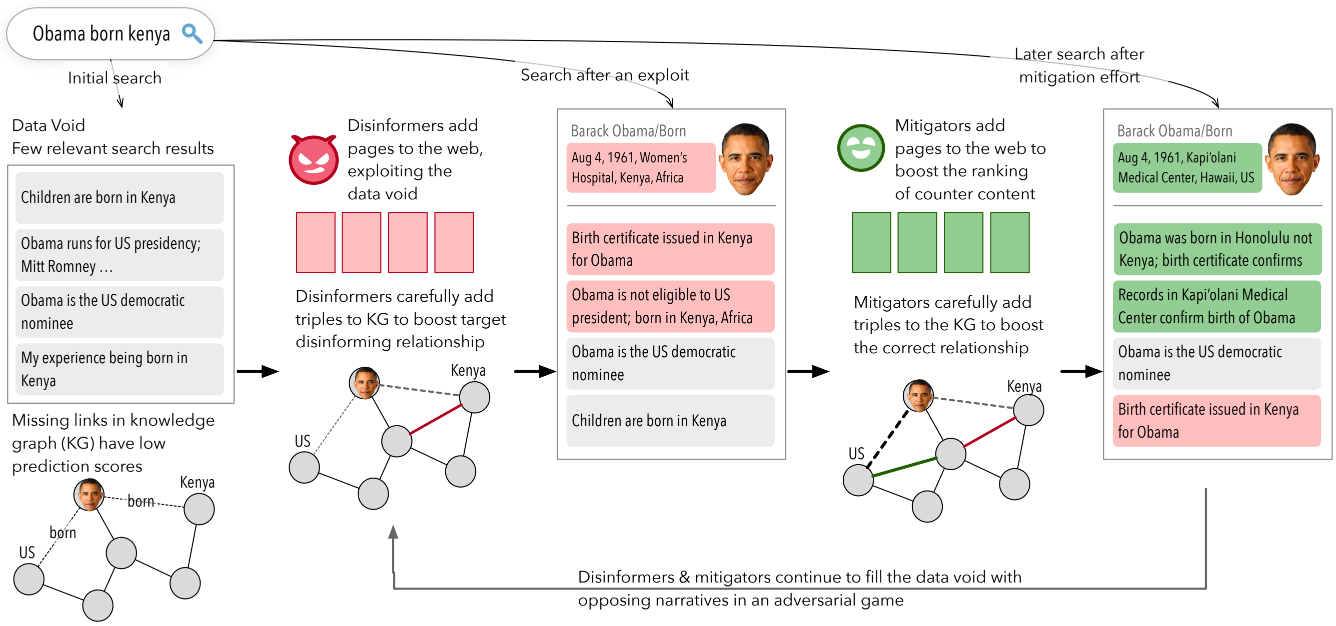

Figure 1: Data voids and how they evolve as disinformers (red) and mitigators (green) act to fill the void with content.

Starting, however, from the point of mitigators knowing exactly what the problematic data void keywords are[1], we argue that we can use light-weight measures to track a data void and the effectiveness of both disinformation and mitigation efforts on Web and KGs. Given this tracker, we show that we can maximize the effectiveness of mitigation efforts given constraints on resources[2] or actions that one can take on third-party search platforms or KG Q&A systems.

In particular, we first demonstrate that we can use *search result rank* to determine the effectiveness and progress of disinformation or mitigation efforts with respect to a set of data void keywords. We provide historical evidence of Google search rank changes of disinformation and its counter information over time using a canonical data void case study about American politics.

On demonstrating that a lightweight measure based on search rank can track data voids, we consequently show how it can also be used to direct how mitigators should respond or *what strategy to employ when promoting counter-content.*

In this paper, we have three main contributions. First, we describe in detail a data void exploit case and show how search ranking can track the data void progression.

Second, we model disinformers and mitigators as adversarial agents with limited control over the strategies in simulated environments and hence inform mitigators how to best tackle data voids (§3).

Third, we empirically evaluate the effectiveness of different mitigation strategies across web search and KG querying (§4). We validate our simulation with real data from one of the case studies. Results show that the choice of mitigation strategy is crucial in the initial phases of a data void: an aggressive mitigation strategy outperforms the baseline 95% of the time. Finally, we discuss related works and differentiate our problem (§5). Our code and data are available at https://github.com/huda-lab/datavoids.

2 CASE STUDY ANALYSIS

Since Golebiewski's and Boyd's report on data voids, many researchers have analyzed and presented data void case studies. For example, *"crisis actor"* is a search query co-opted by conspiracy theorists to refer to victims of mass shootings to prove that an event was staged [25, 45]. *"Iowa Caucus, rigged"* is another search query co-opted by conservative propagandists to undermine the integrity of the Iowa democratic caucus following a glitch in the app that presented election results [6, 33]. Finally, *"pizzagate"* is a search query curated by conspiracy theorists to connect Clinton and Epstein to a sex ring operation run from a pizza store [2, 14, 24, 26]. Most analyses focus on the popularity of search terms over time, as measured by Google Search Trends [10, 30], with some research examining social media platforms like Twitter or YouTube [22, 46].

While search trend analysis has helped researchers understand the progression and impact of data voids, it provides little insight into the actual content created by either disinformers or mitigators as the data void evolves. Ideally, we would like to keep track of content as it is created, indexed, searched and accessed [10]. However, this data is not available.

For historical analysis, we consider an approximation of such data through the use of custom data range searches on a major search engine. This is a proxy dataset for what searchers see on searching data void keywords at different points in time. We search for the data void keywords using a custom date range: an unspecified start date and an increasing daily, weekly, or monthly end date. This creates an approximate snapshot of what users would

[1]Mitigators are often aware of data void keywords as the disinformation narrative is circulating within closed networks [29, 41].
[2]Resource scarcity plays a factor in deciding whether to create counter-content, since not all disinforming narratives gain sufficient traction to merit mitigation.

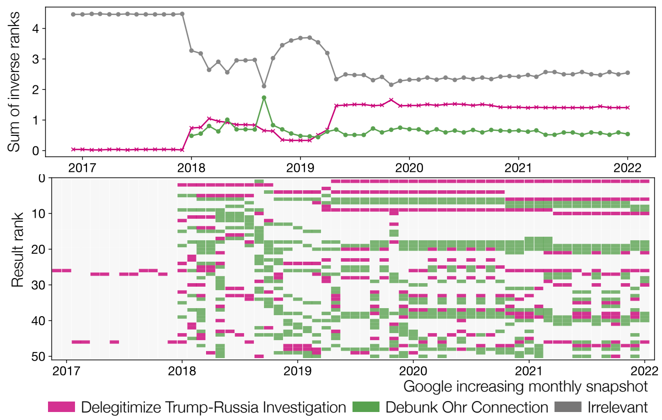

Figure 2: Tracking the effectiveness of disinformers and mitigators in filling the data void (§2.1) and influencing the search ranking of their content.

see —- results and their search ranks — if they had queried the data void terms on different past dates. A typical analysis of this data set would show an absence of any relevant information prior to the emergence of the data void, followed by a gradual increase in the search rankings of misinforming content, and possibly an increase in the rankings of counter content. We note that custom range searches suffer from issues that might affect the accuracy of this analysis: e.g. (i) some results are missing date meta-tags, (ii) page snippets and historical search indices maintained by the search engine might become invalidated by content changes, and (iii) it is difficult to retrieve the actual contents of a page that were published at a given date — archival sites have low coverage.

Data Extraction Process. We implemented a pipeline that extracts search rank data. Search results are obtained through Selenium web automation [9], which emulates clicks on the Google webpage, and the Google Custom Search API [16]. After collecting each link from the search results within a user-specified time range, the pipeline extracts and stores a timestamped copy of the page content using the Trafilatura [3] library to extract text from the raw HTML.

The above process may result in the same URL appearing multiple times across snapshots, requiring the handling of various versions for the same webpage. Each newly extracted copy is compared with previously downloaded pages and only pages exhibiting significant differences from earlier versions are marked as new. Using an LLM (ChatGPT-3.5), the pipeline automatically assesses whether the content of new pages is irrelevant to the data void, or if it falls into one of two or more categories. The categories for the case study are *disinformation*, *mitigation* and *irrelevant*. Using a template, the analyst provides prompts that describe how to differentiate the categories. The initial zero-shot labeling by the LLM has been manually vetted for accuracy by the authors.

The pipeline generates a visualization with a color-coding of the top-50 search rank results over time and a line chart showing the aggregate inverse rank of pages in each category ($\sum \frac{1}{\text{rank}}$). The higher the sum of inverse ranks of content from one side, the more prominent it is in the search results indicating that it is *"winning"*.

2.1 The Nellie Ohr Data Void

Overview: The Trump-Russia collusion investigation began in 2016 to determine if they colluded to manipulate the 2016 US Elections. Steele, a British Intelligence officer, produced a 'dossier' with unverified claims on Trump's connections to Russia. This dossier was a point of contentious political discourse. In mid-2017, a name, Nellie Ohr, emerged in a conspiracy theory connecting the dossier and the collusion allegations. Nellie Ohr's emergence was due to a concerted effort of *keyword curation*: Nellie Ohr was a data void — before then she was a relatively unknown figure with scant information about her online — that was ripe for exploitation. Exploiting this data void started by seeding the internet with stories about Ohr's connection to the Department of Justice through her husband, who was involved in the collusion investigations. QAnon, a far-right group, was an early propagator, introducing unverified information about her on platforms like Reddit and Twitter. These mentions filled the void with a particular narrative before authoritative sources could. Searches for "Nellie Ohr" would land information seekers on content delegitimizing the investigation. In 2018, influential conservative platforms consistently echoed her name and the conspiracy theory, effectively gaming search algorithms.

Results: The rank analysis in Figure 2 shows how the data void was initially filled by sites supporting the narrative that delegitimizes the investigation. When mitigators start to push content, both narratives climb up in the ranking. Over time, both agents put resources into the game with alternating success until the situation stabilizes after two years. Note that the highest mitigation peak coincides with a peak in search trends, which is attributed to the involvement of main stream media (e.g. NYTimes) in debunking the conspiracy [10]. This provides some evidence of the robustness of our methods in tracking certain historical data voids despite the limitations of custom range searches.

3 DATA VOIDS AS AN ADVERSARIAL GAME

We model data void exploits as a game played between two adversarial agents: a disinformer d and a mitigator m who each take turns choosing which content to deploy (e.g., which page in the case of web search, or which triple in the case of the KG). Their goal is to have their own content ranked higher by some user-facing algorithm accessing the information ecosystem, either Web or KG. This game applies to differing narratives, opinions, etc. beyond the narrow lens of factually incorrect as "disinformation" and fact-checks as "mitigation". Our assumption of a fixed set of resources to choose from mimics the resources and access constraints in real-world settings. For example, a disinformation campaign run by a state actor may have access to state-run, media news channels, whereas a political fact-checking team may not.

We study two settings. In the Web setting, agents compete in having their content ranked higher than the counter-part. In the KG setting, a data void is when the two competing, disinformation and mitigation, claims have low *link prediction* scores [44] — the graph may not even have either claim. Since link prediction scores determine the ranking of a claim as an answer to a KG query, we track and measure the effectiveness of disinformation or mitigation actions (e.g. adding new triples to the KG) in terms of this ranking of the competing claims. In the example in Figure 1, the agents add new triples[3] to increase the likelihood of the triple that supports their narrative — (Obama, born, Kenya) vs. (Obama, born, US).

An agent here can represent the actions of multiple decentralized agents with an overlapping agenda. The fact that online content is often produced by multiple different entities who may have little influence on each other is tangential to our analysis. First, prior work does shows that single entities (e.g. a state actors) do independently launch large-scale disinformation campaigns and decentralized campaigns are typically coordinated [29]. Second, the goal of our work is to inform a global strategy, which can direct the efforts of mitigation teams even if they operate independently. Nevertheless, our adversial agents in a game framing is amenable to future extensions where one can study the effects of coordination on disinformation or mitigation strategies.

3.1 The Game-playing Scenario

Five main elements define the game-playing scenario.

Turn. Each agent, d or m, selects a piece of information (x_d or x_m) from their **resource pools** (D, M) to modify the information ecosystem U_{t-1} at each turn t, where $\{t \in \mathbb{Z} | t \geq 1\}$. Let U_0 represent the data void and D_0, M_0 the initial resource pools, then $U_t := U_{t-1} \cup \{x_d, x_m\}$, $D_t := D_{t-1} - x_d$, and $M_t := M_{t-1} - x_m$.

The specific choice of what information to use is guided by the agent's strategy. At each turn both agents act simultaneously. An agent may skip their turn.

Effect. $\mathcal{E} : U \rightarrow \mathbb{R}_d \times \mathbb{R}_m$ Each turn alters the state of the information ecosystem, either amplifying the disinformation or its mitigation. The effect of each move is quantitively tracked by reevaluating the rank (a proxy for visibility and therefore influence) of the disinformation and its mitigation after each turn. For notational convenience, $\mathcal{E}_d$, and $\mathcal{E}_m$ respectively refer to the disinformation and mitigation components of effect.

Cost. $C : X \rightarrow \mathbb{R}$ The cost in this game is assumed to be proportional to the *influence* of an information item within the information ecosystem. Often a proxy for influence is used. Metrics such as node centrality, pagerank, and degree can be used to determine how well connected a page is within the Web or a triple is within the KG and thus their influence or capacity to promote a certain narrative. More influential items are more costly, e.g., adding triples associated with a celebrity to a KG may undergo more scrutiny and vetting, hence more costs may be involved to circumvent or pass these checks than adding triples associated with a less-known figure.

Winning. $\mathcal{W} : (\mathbb{R}_D, \mathbb{R}_M) \rightarrow \{d, m\}$ Agents measure their success based on the rank of their content in the information ecosystem at each turn. The disinformer is winning when the disinformation has higher ranking, and conversely for the mitigator. Thus, $\mathcal{W}(\mathcal{E}(U_t))$ will declare an agent winner if its effect (e.g. ranking) is higher than the other agent at turn t.

While the initial resources an agent has could predetermine the game's final outcome to a certain extent, we are more concerned about the immediate impact and the maximization of the mitigator limited resources' effectiveness while minimizing costs. Therefore, it is not just about who wins in the end, but how effectively the actors influence the information ecosystem as the game progresses.

Strategy. $\mathcal{S} : X \times U \rightarrow x$ In the context of this game, a strategy is an agent's set of rules that dictates which content to deploy when it is their turn. It guides the actions of an agent based on the available resources (D_{t-1} for d and M_{t-1} for m) and the current state of the information ecosystem (U_{t-1}). A strategy can take various forms, such as prioritizing deployment of the most crucial information first or deploying content in a random fashion. The choice and effectiveness of a strategy influences the course of the game.

The strategy is the most important element and the one that is controlled by the players to win the game. We study the following three noting that our model (and instantiated simulations) can be easily extended to support other strategies:

(1) **Random**: In this baseline strategy, an agent chooses a random piece of content to add to their information pool each time. This strategy does not account for the impact or cost of the selected content; therefore, its result can vary greatly across runs.

(2) **Greedy**: The ranking of an information item in the ecosystem is often determined by a variety of factors including its relevance to the search query (e.g., keyword match similarity), the item's influence (e.g., pagerank), etc. In this strategy, the resource pool is sorted once, in decreasing order, apriori by a weighted combination of these factors. At each turn, the agents pulls the topmost item from this pool. Note that these factors alone do not determine the exact final ranking of an item: that depends on all the items currently in the information ecosystem U_t. This aggressive strategy often chooses more costly items to add first.

(3) **Multiobjective Greedy**: A modification of the Greedy strategy, incorporating cost considerations. It sorts the items in the resource pool, once, in decreasing order, apriori using a weighted combination of search-rank factors and negatively weighted cost. This strategy aims to strike a balance between high impact and low cost at each turn, but may need fine-tuning the weights for different ecosystems.

Using this abstraction of data voids, we build two simulators that model the actions of disinformers and mitigators in a *realistic* setting. In the simulated environment, the set of pages or claims available to an agent needs to plausibly represent the influence that such a page has in a web setting or a claim has in a KG. We use search over Wikipedia pages as a stand in for searching over the web and link prediction over FB15k-237 as a stand in of KG answering[4]. These datasets are large enough to support rich queries and data void scenarios, but also sufficiently small to execute complex analysis and simulations in a reasonable time frame.

3.2 Simulating the Web Search Game

We describe our simulation of the game starting with Web search. A corpus of interlinked pages, such as Wikipedia, is given as input. In this setting, an agent aims at making their own narrative more supported, i.e., the pages supporting their narrative must appear higher in search result ranking than the opponent's pages.

[3]Most KGs are bootstrapped and updated according to open resources, such as DBpedia and Wikidata. While both agents can add triples, updates must be done carefully and parsimoniously to avoid spam and vandalism detection techniques [1, 19].

[4]One can disagree with our choice of Wikipedia as a stand-in for the web and FB15k as a stand-in for KGs. Nevertheless, as G. Box : "all models are wrong; some are useful."

Table 1: Properties of the simulated data void scenarios in the Wikipedia data set, and the real Nellie Ohr data void scenario.

Narratives		Size		Power	Data void keywords	Avg keyword frequency in		
d	m	$\|D\|$	$\|M\|$	$\mathcal{E}(U)$		D	M	U_0
Declarative Language	Procedural Language	32	39	$(3.04, 1.36)$	lisp, semantics, javascript, ...	0.35	0.36	0.0006
Optimism	Pessimism	119	133	$(3.42, 2.69)$	nihilism, affective, depressive, ...	0.16	0.20	0.0005
Rationalism	Empiricism	58	117	$(3.04, 2.70)$	descartes, leibniz, gottfried, ...	0.30	0.38	0.0004
Classical Economics	Keynesian Economics	240	90	$(1.41, 4.97)$	maynard, keynes, laissez, faire, ...	0.32	0.33	0.0005
Delegitimize Investigations	Debunk Ohr Connection	13	19	$(1.71, 2.61)$	Nellie, Ohr	0.41	0.34	0.04

The Resource Pools, D, and M. To compose the set of pages available to a disinformer, D, and to a mitigator M, we pick two seed pages representing divergent viewpoints about topics in a domain, e.g., "Declarative Language vs. Procedural Language" or "Rationalism vs. Empiricism". A seed page is labeled arbitrarily as a disinformer, s_d, or mitigator page, s_m. D and M are the disjoint in-neighbors (N^-) of their corresponding seed pages: $D = N^-(s_d) - (N^-(s_d) \cap N^-(s_m))$ and $M = N^-(s_m) - (N^-(s_d) \cap N^-(s_m))$. Let U be the universe of Wikipedia pages, we then identify a set of data void keywords such that each keyword appears in approximately as many pages in D as in M and infrequently in $U - D - M$. Together the keywords cover all pages in $D \cup M$.

We construct the data void by removing all the pages in D and M from U (i.e., $U_0 := U - D - M; D_0 := D; M_0 = M$). At this starting point, a search for the data void keywords will yield results that are by construction irrelevant. As each agent adds a page from their respective set, they change the search results and their performance is evaluated by the ranks of their added pages in the results.

We posit that constructing a data void by subtracting the disjoint in-neighbors of the seed pages realistically approximates an agent's capacity to influence web search. With this construction, we do not assume (i) the distribution of pageranks of the pages available to an agent, (ii) the graph properties of D, M or U, or the (iii) the sizes of D and M relative to U. These are naturally determined by the graph of pages in Wikipedia. This construction might also give one agent more power than the other (e.g, $\mathcal{W}(U) = d$). This is also true of real-world agents who have access to different resources.

Effect. The effect of a mitigator or disinformer's actions at turn t is reflected in the aggregate ranking of disinformer or mitigator pages in U_t. Thus,

$$\mathcal{E}(U_t) = \left(\sum_{x \in D} \frac{\mathbf{1}_{U_t}(x)}{\mathsf{rank}_{U_t}(x)}, \sum_{x \in M} \frac{\mathbf{1}_{U_t}(x)}{\mathsf{rank}_{U_t}(x)} \right)$$

We use the inverse-rank weighted sum of pages as a measure of effect: the higher the search rankings of a disinformer's pages (i.e., the page appears in the search results with lower-valued ranks), the higher the effect of its component and the lower the effect of the mitigator and vice-versa. This is because each rank can only have one page.

In our simulator, given the graph of web pages that contains all pages in U_t, $\mathsf{rank} : x \rightarrow \mathbb{N}$ is computed as follows for a collection of data void keywords:

(1) Let $\mathsf{relevance} : x \rightarrow [0, 1]$ be a measure of how relevant documents are to the data void keywords: higher values mean more text matches in the page. We use Postgres's ts_rank, which takes into account lexical, proximity, and structural information [35].

(2) Let $\mathsf{pagerank} : x \rightarrow [0, 1]$ be the numerical score assigned to a page by the PageRank algorithm [32]. It represents the likelihood that a random walk on the graph ends at page x. It measures a page's relative influence; central pages with higher in-degrees typically have higher pageranks. For example, a news media outlet like CNN has higher pagerank than a blog with few followers. Every turn, $U_0, U_1, \ldots$ requires the recomputation of pageranks as adding a page (node) to the graph also adds its links. However, as computing page rank at every turn is computationally expensive, we compute it once for all pages in U, with the assumption that certain pages stay more important than others.

(3) Now $\frac{1}{2}(\mathsf{relevance}(x) + \mathsf{pagerank}(x))$ is the search $\mathsf{score}(x)$ of a page. We sort pages in descending order of score breaking ties arbitrarily. Thus $\mathsf{rank}(x_1) < \mathsf{rank}(x_2)$ if $\mathsf{score}(x_1) > \mathsf{score}(x_2)$.

Cost. The cost of a page is determined by its pagerank capturing the intuition that pages with higher pageranks require more effort, access, influence or monetary resources to add:

$$C(x) = e^{\mathsf{pagerank}(x)}$$

Winning. We determine the winner at every turn as follows:

$$\mathcal{W}(U_t) = \begin{cases} d & \text{if} \quad \mathcal{E}_d(U_t) > \mathcal{E}_m(U_t) \\ m & \text{otherwise} \end{cases}$$

Strategy. We implement the following three strategies in our simulator; at each turn t:

- **Random**. An agent randomly selects a page from its resource pool without replacement and adds its to U_{t-1}.
- **Greedy**. An agent pulls the top page from its pool (ordered in descending order of $\mathsf{pagerank}$) and adds its to U_{t-1}.
- **Multiobjective Greedy**. For each page x, we compute a linear weighted sum of an estimate of its cost and effect:

$$\frac{1}{2}\left(\mathsf{score}(x) - \frac{C(x) - 1}{e - 1}\right)$$

Pages in the resource pool are ordered in descending order of this weighted sum. At each turn the agent pulls the top page from its pool and adds it to U_{t-1}.

Table 1 summarizes the properties of the resource pools of disinformers and mitigators in four data void scenarios. The simulator can easily be extended to support more scenarios, strategies, etc.

Table 2: Properties of the different simulated data void scenarios in the FB15k-237 knowledge graph.

Query head	rel	**Target Claim** d	m	**Initial KG State** $\mathcal{E}(\mathsf{KG})$	**Size** $\lvert D\rvert, \lvert M\rvert$	**Data void State** $\mathcal{E}(U_0)$	**Power** $\mathcal{E}(U)$
Ben Affleck	director	Argo	The Town	(0.5, 1)	21	(0.1, 0.01)	(0.25, 1)
George Clooney	actor	Good Night, and Good Luck.	Ocean's Twelve	(1, 0.24)	21	(0.14, 0.016)	(1, 0.25)
Ben Affleck	producer	Argo	Pearl Harbor	(0.5, 0.2)	23	(0.17, 0.013)	(0.25, 0.2)
Steven Spielberg	director	Saving Private Ryan	Amistad	(0.5, 1)	19	(0.1, 0.077)	(0.25, 0.5)

3.3 Simulating the KG Querying Game

In this setting, on a given KG, the agents compete on making a triple (e.g., a fact such as the birthplace of a president) more supported and thus higher in the ranking than others by adding new triples.

The Resource Pools, D and M. We pick two target claims or triples s_d : $(\mathsf{head},\ \mathsf{rel},\ \mathsf{tail}_d)$ and s_m : $(\mathsf{head},\ \mathsf{rel},\ \mathsf{tail}_m)$ with the same head and relationship, but different tails. A query (head, rel, ?) returns both claims at different ranks according to their likelihood from a link prediction algorithm. For example, the query (Ben Affleck,directed, ?) on the FB15k-237 KG returns target claims (Ben Affleck, directed, Argo) at rank 2 and (Ben Affleck, directed, The Town) at rank 1. The claims are labeled arbitrarily as disinformer or mitigator claims. We choose four such one-to-many or many-to-many claims from FB15k-237. For simulation run-time scalability, we subsample U, the universe of claims, such that U is the breadth-first neighborhood of both claims (5% of the entire KG).

We then search for the top-25 triples, $\hat{D}$, that explain the disinformer claim and similarly the top-25 triples that explain the mitigator claim, $\hat{M}$, using Kelpie's *necessary tail explanations* [38]. A triple is a necessary tail explanation if (i) it has the form (head, ?, ?) or (?, ?, head), and (ii) removing it from the KG reduces the tail prediction score of the target claim it explains. For example, given the target claim (Obama, nationality, United States), (United States, hadPresident, Obama) is a necessary tail explanation.

For each explanation triple x, Kelpie also produces a relevance score. It is a straightforward score for ranking candidate triples for injection, more intricate methods for ordering candidate triples can be explored in the future [4, 47]. $\mathsf{relevance} : x \longrightarrow \mathbb{R}^+$ describes how well a triple explains the target claim. It is the expected rank worsening of the target claim associated with removing the triple. A higher value means that removing the triple causes a higher increase in the rank-value of the target claim. $\hat{D}$ and $\hat{M}$ are the top-25 necessary tail explanations with the highest relevance scores.

We eliminate any triples that exist in both sets such that the disinformer triples are $D = \hat{D} - (\hat{D} \cap \hat{M})$ and mitigator triples are $M = \hat{M} - (\hat{D} \cap \hat{M})$ and we remove $\{s_d, s_m\} \cup (\hat{D} \cap \hat{M})$ from U. We construct the data void by removing the target claims and all the triples in D and M from U ($U_0 = U - D - M; D_0 = D; M_0 = M$). The query (head, rel, ?) may return the target claims with very low rankings and prediction scores as they no longer exist in the KG and all the supporting (or explanation) triples have been removed. As each agent adds a triple from their set to the KG, the prediction score and the ranking of their target claim increases.

Again, our construction of a void by subtracting sets of triples pre-labeled as disinformer/mitigator based on how well they explain a target claim is a solid approximation of the agents' capacity. With this construction, we do not generate triples that an agent can plausibly add to boost a missing claim, we use what already exists. As with Web search, this construction might give one side more power than the other based on the set of triples. This also is true of real-world agents who can access different resources.

Effect. The effect of a mitigator or disinformer's actions at turn t is the inverse-rank of their target claim in U_t. Thus,

$$\mathcal{E}(U_t) = \left(\mathsf{rank}_{U_t}(s_d)^{-1}, \mathsf{rank}_{U_t}(s_m)^{-1}\right)$$

In KG querying, especially when the KG suffers from incompleteness, *link prediction* is used for query answering. Here, a KG embedding (KGE) facilitates the prediction of a missing tail in a triple [13]. The link prediction algorithms in our simulator predicts answer tails for the given data void query (head, rel, ?), with a score for each predicted triple [42]. Query answers are sorted in decreasing order of prediction scores to derive the rank of the mitigator or disinformer target claims. We retrain the KGE model after each triple addition to get ranking changes in the prediction score.

Cost. Let $\mathsf{degree} : x \longrightarrow \mathbb{Z}^+$ be the degree of the head or tail entity in x that is not the head entity of the data void query. We define cost as $C(x) = \mathsf{degree}(x)$. This function captures the intuition that higher degree entities are more popular and are often subject to additional scrutiny when their facts are added to the KG. Hence, adding these triples requires more access, influence or resources [8, 21].

Winning. We determine the winner at every turn as follows:

$$\mathcal{W}(U_t) = \begin{cases} d & \text{if} \quad \mathcal{E}_d(U_t) > \mathcal{E}_m(U_t) \\ m & \text{otherwise} \end{cases}$$

Strategy. We implement the following three strategies:

- **Random.** At turn t, an agent randomly selects a triple from their resource pool without replacement and adds it to U_{t-1}.
- **Greedy.** An agent's resource pool is ordered in decreasing order of the triple's relevance. At each turn t, the agent pulls the top triple from the pool and adds it to U_{t-1}.
- **Multiobjective Greedy.** For each triple x, we compute a linear weighted sum as an estimate of its cost and effect:

$$\frac{1}{2}\left(\mathsf{relevance}(x) - \frac{C(x) - 1}{\max_{y \in U} C(y) - 1}\right)$$

  Triples in the resource pools, D or M, are ordered in decreasing order of this weighted sum. At each turn t the agent pulls the top page from its pool and adds it to U_{t-1}.

Table 2 shows for every disinformer-mitigator claim, their initial effect (inverse ranks), $\mathcal{E}(\mathsf{KG})$, their effect, $\mathcal{E}(U)$, after they are removed, and after their supporting sets D, M are removed, $\mathcal{E}(U_0)$.

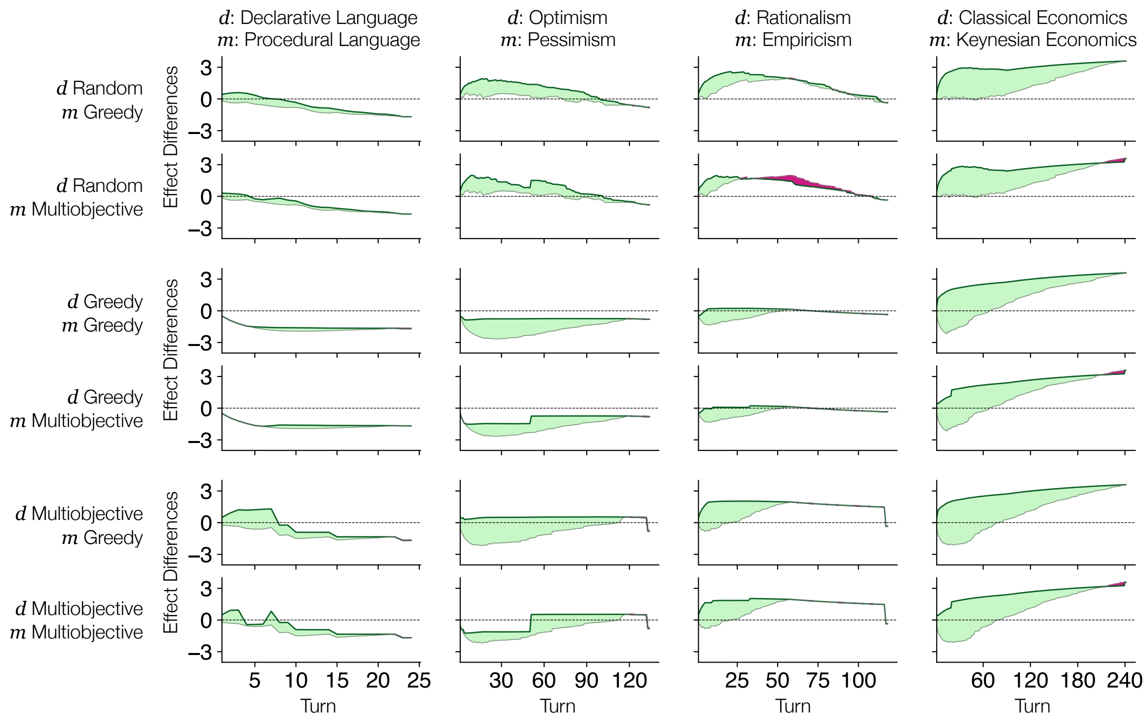

Figure 3: Differences of effects $\mathcal{E}_m(U_t) - \mathcal{E}_d(U_t)$ at every turn t of the Web search simulation across four data void scenarios.

4 EXPERIMENTAL RESULTS

Which strategies maximize the effect of the agents' actions?

We are interested in which strategies are more impactful and cost-effective over the course of the simulation. So, even if an agent never strictly *wins* (§3), they did the best with what they have. Figures 3 and 5 illustrate the effects of simulating different strategies across the data void scenarios described in Tables 1 (Web setting) and 2 (KG setting). The Random strategy (§3.1) is the baseline to beat. We fix the strategy of the disinformer to one among Random, Greedy, and Multiobjective; this choice does not influence the choice of the mitigator's strategy. When simulating the random strategy, we compute averages and variance over 15 runs in web search and over 10 runs in KG querying. We then evaluate the performance of the mitigator when employing one of two strategies, Greedy or Multiobjective, against its performance when employing the Random strategy. Each plot in Figures 3 and 5 shows (i) the baseline performance of $\mathcal{E}_m(U_t) - \mathcal{E}_d(U_t)$ at every turn t when the mitigator is employing the Random strategy — a gray line — and (ii) the performance of the evaluated strategy — a thick green line. The shaded area between the two curves illustrates how well or poorly a strategy performs when compared to the baseline. We shade this region green to indicate that the evaluated strategy is outperforming the baseline and red otherwise. If at turn t, we are above or at the 0 line, the mitigator strategy is also *winning*, i.e., its effect being greater than or equal to the disinformer strategy at turn t.

Figures 4 and 6 report the cost of each strategy as more of the mitigator's pages or triples are added at every turn. As the mitigator picks a strategy without information about the disinformer strategy at hand, the cost graphs are independent of the latter.

We find the following insights from the analysis of the results for the Web search game (Figs. 3, 4):

► Greedy is the most aggressive strategy and allows mitigators to get ahead of an emerging data void scenario even if their resources overall are limited. This is especially true if the disinformer is non-strategic, i.e., using a Random strategy, or cost-cutting with a Multiobjective strategy. In the first three scenarios of the web game (Figure 3), the disinformer ultimately wins. But following a Greedy strategy allows the mitigator to maximize their effect for the longest duration of turns initially. Across all scenarios, Greedy outperforms the other strategies 95% of the time, in the first half of the simulation. This might be important in a situation where the mitigator wishes to get ahead of a trending situation and reach early searchers of the data void, limiting exposure and a possible escalation of the disinforming narrative.

► A Random strategy has little chance at outperforming a strategic disinformer with better resource pools — across all scenarios, Random only outperforms other strategies 3% of the time in the first half of the simulation. For example, notice the gray line in the second data void scenario where the disinformer promotes 'Optimism' and uses either a Greedy or a Multiobjective strategy and the mitigator promotes 'Pessimism' and uses a Random strategy.

► Multiobjective strategies have less pronounced effects compared to Greedy ones. While they outperform Random strategies, they may not yield as many "wins" against the disinformer. Nonetheless, initially, they are less expensive than greedy ones (Fig. 4).

We obtain similar insights from the analysis of the results for the KG query answering game (Figs. 5 and 6):

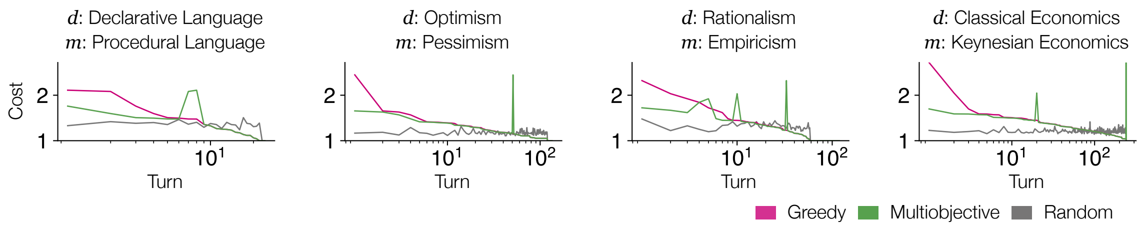

Figure 4: Mitigator costs for different strategies at each turn of the web search simulation across four data void scenarios.

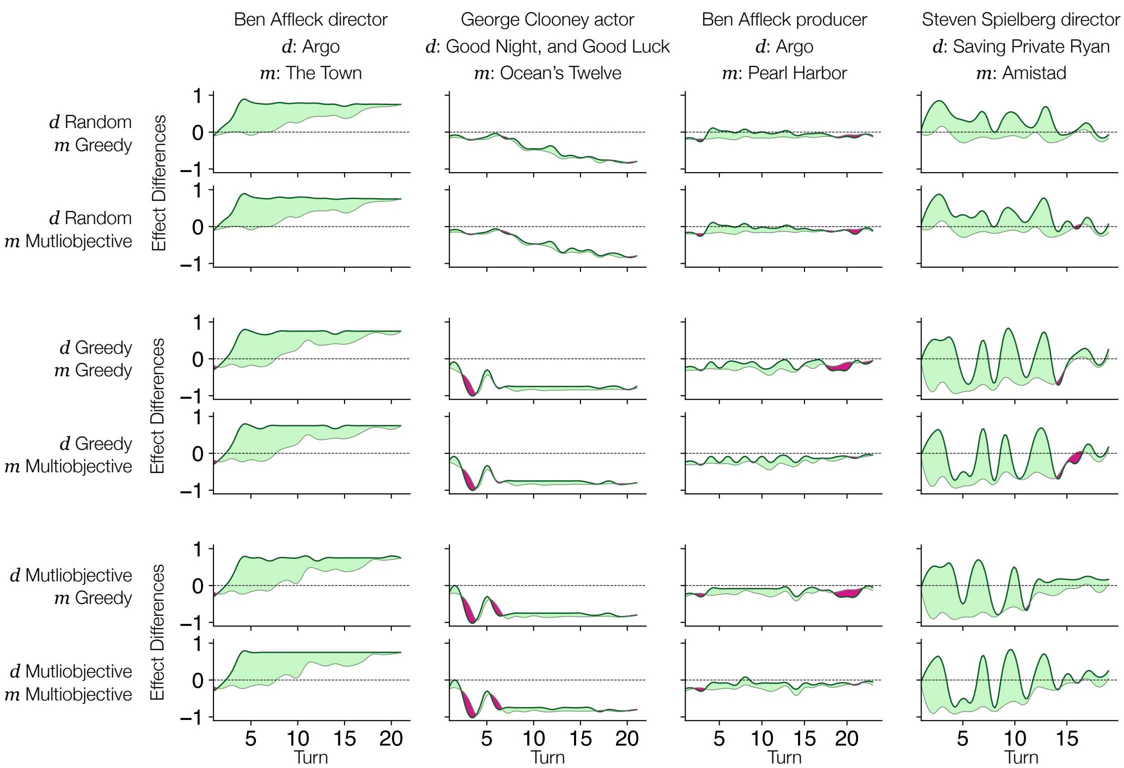

Figure 5: Differences of effects $\mathcal{E}_m(U_t) - \mathcal{E}_d(U_t)$ at every turn t of the KG Q&A simulation across four data void scenarios.

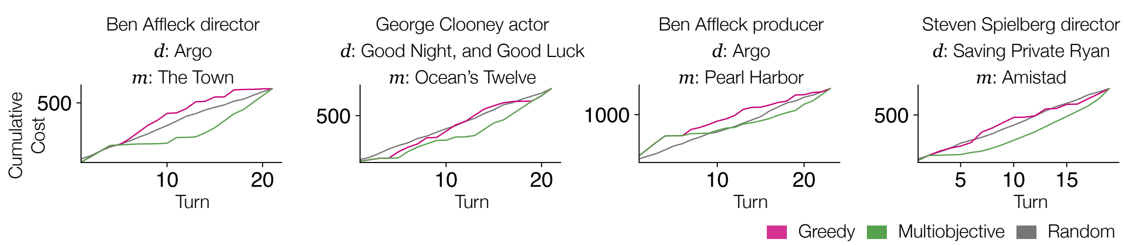

Figure 6: Mitigator's *cumulative* costs for different strategies at each turn of the KG Q & A simulation across 4 data voids.

► Both Greedy and Multiobjective strategies are more effective than the Random strategy. Across all scenarios, the informed strategies give overall better results.

► However, the role of the data is even more important than that played by the strategies. All strategies fall short in the cases where the mitigator's resources are less effective than the disinformer's ones. In the first and last data void scenarios of Figure 5, when the mitigator has overall a more effective set of triples to choose from, a Greedy or Multiobjective strategy performs much better than a Random strategy. However, for the second and third scenarios, the mitigator can barely overtake the disinformer and there is little benefit to Greedy or Multiobjective strategies compared to Random.

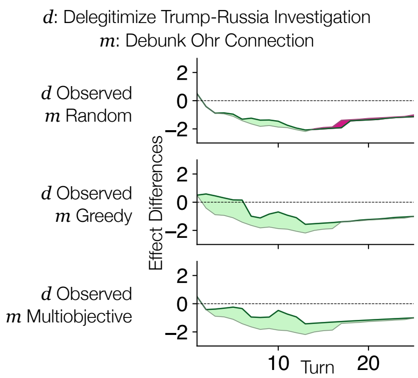

Figure 7: Differences of effects $\mathcal{E}_m(U_t) - \mathcal{E}_d(U_t)$ at every turn t of simulating the web search game with data scraped from the Nellie Ohr case study (§2.1).

► As in the Web setting, the Greedy strategy quickly consumes more resources, as depicted in Fig. 6. The Multiobjective starts lower while providing comparable performance according to Fig. 5.

Validation with the Nellie Ohr Case. Using the monthly historical search results for the Nellie Ohr data void described in §2.1, we examine in Fig. 7 the impact of choosing one of the three strategies on the mitigation efforts in the web search setting.

Unlike the previous evaluation, we set the baseline strategy for disinformer and mitigator to the order in which pages appeared on Google search. We then plot the differences in effect when the mitigator uses one of three strategies. We estimate the Pagerank of every page using Moz URL Metrics [39]. We find the following:

► Both Greedy and Multiobjective outperform Random and the observed baseline strategy, but with Greedy the mitigator beats the disinformer for more turns initially.

► The observed (baseline) mitigator strategy and random strategy are similar. This is not surprising as we believe different mitigators may add web pages without coordination.

Practical Takeaways.

► In both settings, in the case of early detection of a new data void, mitigators have alternative options for *matching* the disinformer depending on the quality of the available resources and budget.

► For better ranking results and faster impact, an informed strategy (Greedy, Multiobjective) should be favored to a Random one, with Multiobjective chosen in cases of a limited budget.

► Determining the relative impact of a triple ,might be difficult if the mitigators do not have access to the full KG or its link prediction models. Figure 6 shows a correlation between cost, determined by degree, and the order of edges added by Greedy strategy. Hence, triple degree can be an indication of its relevance.

► Greedy strategies need access to "influential" pages or entities. In a greedy suppression, mitigators need to create consortia, hyperlink their resources, attract the sponsorship of high-value entities or nodes, to maximize the ranking of deployed mitigation content by search engines and KG answering systems[5].

5 RELATED WORK

Our work lies in the intersection of two fields: Web Search and KG querying attacks, and data void studies.

Our Web Search game aligns with studies exploring the malicious or intentional manipulation of ranking systems, including PageRank exploits [7], web spam detection [40, 43], and search engine poisoning [5, 27, 28]. We adopt simple techniques to illustrate our framework, leaving more intricate strategies to future work.

Our KG querying game resembles *data poisoning attacks* - an adversarial assault on ML models where the attacker manipulates a subset of the training data to tailor the model [47]. More precisely, both agents act as attackers by inserting triplets in the KG. These triplets act as training data for the link prediction model that determines the ranking of the agent's target triple. KG poisoning attacks [4, 11, 34, 47] often study a variety of direct and indirect attacks including deletions and relationship modifications. We simulate agent attacks as selecting which triples to add from a set of facts from a KG that is reduced for the simulation. We leave to future research how to create "new" triples for manipulation.

Several studies explore the presence and impact of data void exploits and their early detection [15, 30, 41]. Beyond web search and KG answering, the impact of data voids on search-adjacent systems such as auto-play, auto-fill has also been studied [20, 23, 36, 37]. A recent system [12] helps mitigators construct counter-content by using NLP techniques to analyze the disinformation text. However, we are the only work that proposes a concrete method to track the progress of data voids and to evaluate mitigation strategies in terms of content deployment.

6 CONCLUSION

We develop rank-based measures to track the progress of disinformers or mitigators in filling up data voids in the Web and KGs. We illustrate the power of such a tracker with real case-studies. We formulate data void exploits and response as an adversarial game between disinformers and mitigators and use a simulator modeled on the game to help mitigators determine effective response strategies given their resource constraints. Future work directions include extending the framework with more sophisticated (and costly) strategies and better estimation of the effectiveness of the content available to agents [4, 11]. Finally, we plan to conduct user studies on emerging real-world data voids with information integrity teams, who list monitoring tools as a missing asset in assessing disinformation threats [29]. With this work, we now have a practical way to forward the conversation on data voids from "no easy fix" [15] to developing cost-effective ways to tackle them.

ACKNOWLEDGMENTS

This work was supported by the ASPIRE Award for Research Excellence (AARE-2020) grant AARE20-307, NYUAD CITIES through Tamkeen - Research Institute Award CG001, and in part by the ANR project ATTENTION (ANR-21-CE23-0037).

[5]In the "Nellie Ohr" case, disinformers promoted their narrative on The Daily Caller — a site with high influence, on par with the Washington Post in SEO metrics [39]. The Department of Justice, which responded, ranked low. This disparity can be found globally, e.g., Italy's state-run media Rai ranks as high as the Washington Post, but Pagella Politica, a political fact-checker, is on par with the DOJ.

REFERENCES

[1] B Thomas Adler, Luca De Alfaro, Santiago M Mola-Velasco, Paolo Rosso, and Andrew G West. 2011. Wikipedia vandalism detection: Combining natural language, metadata, and reputation features. In *Computational Linguistics and Intelligent Text Processing: 12th International Conference, CICLing*. Springer, 277–288.

[2] Gregor Aisch, Jon Huang, and Cecilia Kang. 2016. Dissecting the #PizzaGate Conspiracy Theories. *The New York Times* (10 Dec 2016). https://www.nytimes.com/interactive/2016/12/10/business/media/pizzagate.html

[3] Adrien Barbaresi. 2021. Trafilatura: A Web Scraping Library and Command-Line Tool for Text Discovery and Extraction. In *Proceedings of the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*. Association for Computational Linguistics, 122–131. https://aclanthology.org/2021.acl-demo.15

[4] Peru Bhardwaj, John Kelleher, Luca Costabello, and Declan O'Sullivan. 2021. Poisoning Knowledge Graph Embeddings via Relation Inference Patterns. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Association for Computational Linguistics, Online, 1875–1888. https://doi.org/10.18653/v1/2021.acl-long.147

[5] Nicholas Boucher, Luca Pajola, Ilia Shumailov, Ross Anderson, and Mauro Conti. 2023. Boosting Big Brother: Attacking Search Engines with Encodings. In *Proceedings of the 26th International Symposium on Research in Attacks, Intrusions and Defenses* (, Hong Kong, China,) *(RAID '23)*. Association for Computing Machinery, New York, NY, USA, 700–713. https://doi.org/10.1145/3607199.3607220

[6] Jordan Chariton. 2020. Investigator: DNC Was "Directly Involved" in Iowa Caucus App Development, Countering DNC Denial. *The Intercept* (2020). https://theintercept.com/2020/12/23/dnc-iowa-caucus-app-shadow/ Accessed: 2024-01-15.

[7] Alice Cheng and Eric J. Friedman. 2006. Manipulability of PageRank under Sybil Strategies. https://api.semanticscholar.org/CorpusID:14707993

[8] Omkar Deshpande, Digvijay S. Lamba, Michel Tourn, Sanjib Das, Sri Subramaniam, Anand Rajaraman, Venky Harinarayan, and AnHai Doan. 2013. Building, maintaining, and using knowledge bases: a report from the trenches. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*. Association for Computing Machinery, New York, NY, USA, 1209–1220. https://doi.org/10.1145/2463676.2465297

[9] Selenium Developers. 2023. selenium. https://pypi.org/project/selenium/ Python package version 4.14.0.

[10] EC, Rafiq Copeland, Jenny Fan, and Tanay Jaeel. 2020. Filling the Data Void. https://datavoids.2020.bkmla.org/

[11] Minghong Fang, Guolei Yang, Neil Zhenqiang Gong, and Jia Liu. 2018. Poisoning attacks to graph-based recommender systems. In *Proceedings of the 34th annual computer security applications conference*. 381–392.

[12] Claudia Flores-Saviaga, Shangbin Feng, and Saiph Savage. 2022. Datavoidant: An AI System for Addressing Political Data Voids on Social Media. *Proc. ACM Hum.-Comput. Interact.* 6, CSCW2, Article 503 (nov 2022), 29 pages. https://doi.org/10.1145/3555616

[13] Xiou Ge, Yun-Cheng Wang, Bin Wang, and C. C. Jay Kuo. 2023. Knowledge Graph Embedding: An Overview. arXiv:2309.12501 [cs.AI]

[14] Joshua Gillin. 2016. How Pizzagate went from fake news to a real problem for a D.C. business. *PolitiFact* (2016). https://www.politifact.com/article/2016/dec/05/how-pizzagate-went-fake-news-real-problem-dc-busin/ Accessed: 2024-01-15.

[15] Michael Golebiewski and Danah Boyd. 2019. *Data voids: Where missing data can easily be exploited.* Technical Report. Data & Society Research Institute.

[16] Google. 2024. Google Custom Search JSON API Overview. https://developers.google.com/custom-search/v1/overview

[17] Google Blog. 2012. *Introducing the Knowledge Graph: things, not strings.* https://blog.google/products/search/introducing-knowledge-graph-things-not/ Accessed on 2024-01-15.

[18] Google Support. 2024. *The Knowledge Graph.* https://support.google.com/knowledgepanel/answer/9787176?hl=en-GB Accessed on 2024-01-15.

[19] Thomas Green and Francesca Spezzano. 2017. Spam users identification in wikipedia via editing behavior. In *Proceedings of the International AAAI Conference on Web and Social Media*, Vol. 11. 532–535.

[20] Eslam Hussein, Prerna Juneja, and Tanushree Mitra. 2020. Measuring Misinformation in Video Search Platforms: An Audit Study on YouTube. *Proc. ACM Hum.-Comput. Interact.* 4, CSCW1, Article 48 (may 2020), 27 pages. https://doi.org/10.1145/3392854

[21] Viet-Phi Huynh and Paolo Papotti. 2019. A Benchmark for Fact Checking Algorithms Built on Knowledge Bases. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM*. ACM, 689–698. https://doi.org/10.1145/3357384.3358036

[22] Prerna Juneja, Md Momen Bhuiyan, and Tanushree Mitra. 2023. Assessing Enactment of Content Regulation Policies: A Post Hoc Crowd-Sourced Audit of Election Misinformation on YouTube. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 545, 22 pages. https://doi.org/10.1145/3544548.3580846

[23] Prerna Juneja, Md Momen Bhuiyan, and Tanushree Mitra. 2023. Assessing enactment of content regulation policies: A post hoc crowd-sourced audit of election misinformation on YouTube. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (, Hamburg, Germany,) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 545, 22 pages. https://doi.org/10.1145/3544548.3580846

[24] Catherine Kim. 2024. Pizzagate, QAnon, and the 'Epstein List': Why the Far Right Is Obsessed with Sex Trafficking. *Politico* (2024). https://www.politico.com/news/magazine/2024/01/09/why-maga-is-obsessed-with-jeffrey-epstein-00134394 Accessed: 2024-01-15.

[25] Jason Koebler. 2018. Where the 'Crisis Actor' Conspiracy Theory Comes From. *Vice* (2018). https://www.vice.com/en/article/pammy8/what-is-a-crisis-actor-conspiracy-theory-explanation-parkland-shooting-sandy-hook Accessed: 2024-01-15.

[26] Kim LaCapria. 2016. Is Comet Ping Pong Pizzeria Home to a Child Abuse Ring Led by Hillary Clinton? *Snopes* (2016). https://www.snopes.com/fact-check/pizzagate-conspiracy/ Accessed: 2024-01-15.

[27] Nektarios Leontiadis, Tyler Moore, and Nicolas Christin. 2014. A Nearly Four-Year Longitudinal Study of Search-Engine Poisoning. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security* (Scottsdale, Arizona, USA) *(CCS '14)*. Association for Computing Machinery, New York, NY, USA, 930–941. https://doi.org/10.1145/2660267.2660332

[28] Dirk Lewandowski, Sebastian Sünkler, and Nurce Yagci. 2021. The influence of search engine optimization on Google's results: A multi-dimensional approach for detecting SEO. In *Proceedings of the 13th ACM Web Science Conference 2021* (Virtual Event, United Kingdom) *(WebSci '21)*. Association for Computing Machinery, New York, NY, USA, 12–20. https://doi.org/10.1145/3447535.3462479

[29] Shujaat Mirza, Labeeba Begum, Liang Niu, Sarah Pardo, Azza Abouzied, Paolo Papotti, and Christina Popper. 2023. Tactics, threats and targets: Modeling disinformation and its mitigation. In *NDSS 2023, Network and Distributed System Security Symposium, 27 February-3 March 2023, San Diego, California, USA*, Usenix (Ed.). San Diego. http://dx.doi.org/10.14722/ndss.2023.23657

[30] Ov Cristian Norocel and Dirk Lewandowski. 2023. Google, data voids, and the dynamics of the politics of exclusion. *Big Data & Society* 10, 1 (2023), 20539517221149099. https://doi.org/10.1177/20539517221149099

[31] Stefano Ortona, Venkata Vamsikrishna Meduri, and Paolo Papotti. 2018. Robust Discovery of Positive and Negative Rules in Knowledge Bases. In *34th IEEE International Conference on Data Engineering, ICDE*. IEEE Computer Society, 1168–1179. https://doi.org/10.1109/ICDE.2018.00108

[32] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1998. *The PageRank Citation Ranking: Bringing Order to the Web.* Technical Report. Stanford Digital Library Technologies Project. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.31.1768

[33] Tyler Pager. 2020. Iowa autopsy report: DNC meddling led to caucus debacle. *Politico* (2020). https://www.politico.com/news/2020/12/12/iowa-caucus-dnc-report-444649 Accessed: 2024-01-15.

[34] Pouya Pezeshkpour, Yifan Tian, and Sameer Singh. 2019. Investigating Robustness and Interpretability of Link Prediction via Adversarial Modifications. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, Minneapolis, Minnesota, 3336–3347. https://doi.org/10.18653/v1/N19-1337

[35] PostgreSQL. 2023. Text Search Ranking - PostgreSQL Documentation. https://www.postgresql.org/docs/current/textsearch-controls.html#TEXTSEARCH-RANKING

[36] Manoel Horta Ribeiro, Raphael Ottoni, Robert West, Virgílio A. F. Almeida, and Wagner Meira. 2020. Auditing radicalization pathways on YouTube. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency* (Barcelona, Spain) *(FAT* '20)*. Association for Computing Machinery, New York, NY, USA, 131–141. https://doi.org/10.1145/3351095.3372879

[37] Ronald E. Robertson, Shan Jiang, Kenneth Joseph, Lisa Friedland, David Lazer, and Christo Wilson. 2018. Auditing Partisan Audience Bias within Google Search. *Proc. ACM Hum.-Comput. Interact.* 2, CSCW, Article 148 (nov 2018), 22 pages. https://doi.org/10.1145/3274417

[38] Andrea Rossi, Donatella Firmani, Paolo Merialdo, and Tommaso Teofili. 2022. Explaining Link Prediction Systems Based on Knowledge Graph Embeddings. In *Proceedings of the 2022 International Conference on Management of Data* (Philadelphia, PA, USA) *(SIGMOD '22)*. Association for Computing Machinery, New York, NY, USA, 2062–2075. https://doi.org/10.1145/3514221.3517887

[39] SEOMoz Inc. 2024. *Moz URL Metrics.* https://moz.com/help/links-api/making-calls/url-metrics

[40] Nikita Spirin and Jiawei Han. 2012. Survey on web spam detection: principles and algorithms. *SIGKDD Explor. Newsl.* 13, 2 (may 2012), 50–64. https://doi.org/10.1145/2207243.2207252

[41] Francesca Bolla Tripodi. 2022. *The Propagandists' Playbook: How Conservative Elites Manipulate Search and Threaten Democracy.* Yale University Press, New

Haven and London.

[42] Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. 2016. Complex Embeddings for Simple Link Prediction. arXiv:1606.06357 [cs.AI]

[43] Matta Sree Vani and O. Yaswanth Babu. 2013. A Survey on Link Based Algorithms for Web Spam Detection. https://api.semanticscholar.org/CorpusID:55491284

[44] Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. 2017. Knowledge Graph Embedding: A Survey of Approaches and Applications. *IEEE Transactions on Knowledge and Data Engineering* 29, 12 (2017), 2724–2743. https://doi.org/10.1109/TKDE.2017.2754499

[45] Elizabeth Williamson. 2022. Sandy Hook Hoax: The Power of Conspiracy Theories. *NPR* (2022). https://www.npr.org/2022/03/09/1084912392/sandy-hook-hoax-elizabeth-williamson Accessed on 2024-01-15.

[46] Muheng Yan, Yu-Ru Lin, and Wen-Ting Chung. 2022. Are Mutated Misinformation More Contagious? A Case Study of COVID-19 Misinformation on Twitter. In *Proceedings of the 14th ACM Web Science Conference 2022* (Barcelona, Spain) *(WebSci '22)*. Association for Computing Machinery, New York, NY, USA, 336–347. https://doi.org/10.1145/3501247.3531562

[47] Hengtong Zhang, Tianhang Zheng, Jing Gao, Chenglin Miao, Lu Su, Yaliang Li, and Kui Ren. 2019. Data Poisoning Attack against Knowledge Graph Embedding. (2019), 4853–4859. https://www.ijcai.org/proceedings/2019/674