DSPIN: Detecting Automatically Spun Content on the Web

Qing Zhang, David Y. Wang, Geoffrey M. Voelker

University of California, San Diego
What is Spinning?

• A **Black Hat Search Engine Optimization (BHSEO)** technique that rewords original content to avoid duplicate detection

• Typically an **article (seed)** is spun multiple times creating **$N$ versions** of the article that will be posted on **$N$ different sites**

• Artificially **generate interest** to increase search result rankings of targeted site
Spinning Example

You have actually seen the feared demon-eye impact that occurs when the camera flash bounces off the eye of a person or animal. An otherwise fantastic image can be ruined by this. Technically, this is

You’ve seen the dreaded demon-eye impact that happens when the camera flash bounces off the eye of an individual or animal. An otherwise terrific picture can be ruined by this. Technically, this is
Spinning Approaches

Human Spinning
• Hire a real person from an online marketplace (i.e. Fiverr, Freelancer) to spin manually
  • Pros:
    – Reasonable text readability
  • Cons:
    – Expensive ($2-8 / hr)
    – Not scalable (humans)

Automated Spinning
• Run software to spin automatically
  • Pros:
    – Fast
    – Cheap ($5)
    – Scalable (500 articles / job)
    – Minimal human interaction
  • Cons:
    – Can read awkwardly
Spinning in BHSEO

Start with a seed article and SEO Software
Spinning in BHSEO

SEO Software submits the article to spinner (TBS)
Spinning in BHSEO

TBS spins the article and verifies plagiarism detection fails
Spinning in BHSEO

SEO Software receives spun article
Spinning in BHSEO

SEO Software posts articles on User Generated Content through proxies

http://<moneysite>
Spinning in BHSEO

Search Engine consumes user generated content.

Search Engine

User Generated Content
Goals

• Understand the current state of automated spinning software using one of the most popular spinners (The Best Spinner)
• Develop techniques to detect spinning using immutables + mutables
• Examine spinning on the Web using Dspin, our system to identify automatically spun content
The Best Spinner (TBS)

• TBS consists of **two parts**
  – Program (binary): provides the *user interface*
  – Synonym dictionary: a *homemade, curated list of synonyms* that are updated weekly

• **Replaces text** with synonyms from dictionary

• We **extract the synonym dictionary** through reverse engineering the binary
TBS Example

The LapBand \{procedure\|process\|method\} \{is a\|is really a\|is often a\|can be a\|is\} \{treatment\|surgical procedure\|surgery treatment\|medical operation\} that has included LapBand apart from other bariatric procedures? It has to do with risk and conv

{If you are\|If you’re\|In case you are\|Should you be\|For anyone who is\} overweight, previously stated, the LapBand \{surgery\|surgical treatment\|surgical procedure\} \{benefits\|advantages\}, but {it is\|it’s\|it can be\|it really is\|it truly is\} not \{right\|corre\} on for five \{signs\|indicators\} that the LapBand \{surgery\|surgical treatment\|surgi
Immutables + Mutables

• An article is composed of immutables (*NOT IN dictionary*) and mutables (*IN dictionary*)

   Wechseln zu: Navigation, Suche

Red Eye and Your Digital Camera

You have actually seen the feared demon-eye impact that occurs when the camera flash bounces off the eye of a person or animal. An otherwise fantastic image can be ruined by this. Technically, this is
Spinning Detection Algorithm

• **Immutables detection** computes the ratio of shared immutables between two pages
  • Works well in practice except in corner case where there are few immutables to compare

• **Mutables detection** computes the ratio of all shared words after two levels of recursively expanding synonyms
  • Also works well and handles corner case, but expensive
Other Approaches

- Duplicate content detection is a well known problem for Search Engines
- Explored other approaches:
  - Hashes of substrings [Shingling]
  - Parts of speech [Natural Language Processing]
- Spinning is designed to circumvent these approaches (i.e. replace every Nth word, synonym phrases)
Validation

• Setup controlled experiment using TBS
• 600 article test data set
  – Started with 30 seed articles
    • 5 articles from 5 different article directories
    • 5 articles randomly chosen from Google News
  – Each article spun 20 times w/ bulk spin option
• Immutables detects all spun content and matches with the source
DSpin

• Detection from **Search Engine POV**
  – **Input:** set of article pages crawled from the Web
  – **Output:** set of pages flagged as *auto spun*

• Build graph of **clusters of “similar” pages using immutables + mutables** approach
  – Each page represents a node
  – Create edges between pairs of nodes using immutables, verify edges using mutables
  – Each connected components is cluster
Results

• Ran DSpin on a real life data set
  – Set of 797 abused wikis
  – Crawl each wiki daily for newly posted articles
  – Collected 1.23M Articles from Dec 2012

• Address the following questions:
  – Is spinning a problem in the wild?
  – Can we characterize spinning behavior?
Filtering

• Filter out pages that are: non-English, exact duplicates, < 50 words, or primarily links

225K spun pages remaining. Spinning is for real.
Spinning campaigns target business + marketing terms
Cluster Size

• **12.7K clusters from 225K spun pages**

![Graph showing distribution of cluster sizes with annotations]

80% of cluster size ≤ 9
90% of cluster size ≤ 44

**Moderate clusters of spun articles in abused wikis**
Timing Duration

• Duration reveals how long a campaign lasts
• Compute by extracting dates, max – min

Most campaigns occur in bursts.
Conclusion

• Proposed + evaluated a spinning detection algorithm based on immutables + mutables that Search Engines can implement
• Demonstrated the algorithm's applicability on a real life data set (abused wikis)
• Characterized the behavior of at least one slice of the Web where spun articles thrive
Thank You!

• Q&A
TBS Coverage

• Only one synonym dictionary was used to implement DSpin, is this system still applicable widely (i.e. for other spinners)?
  – We had no prior knowledge about how articles from abused wikis were spun
  – Yet we still detected spun articles
Synonym Dictionary Churn

• How much does the synonym dictionary change over time?
  – We re-fetched synonym dictionary four months after the initial study and found that 94% of terms remain the same
  – Furthermore, DSpin detected spun articles posted months prior
Synonyms in the Cloud

• What if the spinner stores the synonym dictionary in the cloud?
  – There is an operational cost for the spinner (network bandwidth == $$$)
  – Can still reconstruct synonym dictionary through controlled experiments (i.e. submitting our own articles for spinning)
Scalability

• How can Search Engines implement the immutables algorithm?
  – Assume Search Engines already perform duplicate content detection
  – Can think of immutables approach as performing duplicate content detection on the immutables portion of the pages (a subset of what is already currently done)
Pages/Cluster vs. Domains/Cluster

![Graph showing the relationship between Pages/Cluster and Domains/Cluster. The x-axis represents Domains/Cluster on a log scale, and the y-axis represents Pages/Cluster on a log scale. The graph includes a trend line and clusters of data points.](image-url)
Backlinks

Wiki

GoArticles

Backlinks $\sim =$ # pages

Spammers are crafting unique links across spinning campaigns
Seed Page
Breakdown

**Wiki Dataset**
- Remaining: 43%
- Spun: 53%
- Near duplicates: 4%

**GoArticles**
- Remaining: 90%
- Spun: 7%
- Near duplicates: 3%
Domains/Cluster

- 70% for <= 4 domains/cluster
- 80% for <= 9 domains/cluster
- 90% for <= 42 domains/cluster
Percent Spun per Domain

![Percent Spun per Domain](image)
Seed Pages continued