Identifying Information Provenance in Support of Intelligence Analysis, Sharing, and Protection[1]

Terrance Goan, Emi Fujioka, Ryan Kaneshiro, and Lynn Gasch

Stottler Henke Associates, Inc.
1107 NE 45th St., Suite 310, Seattle, WA 98105
{goan, emifuji, ryank, lynng}@stottlerhenke.com

1 Introduction

In recent years, it has become clear that our ability to create vast information assets far outstrips our ability to exploit and protect them. The Intelligence Community faces particularly significant information management challenges as they seek to: improve information awareness amongst analysts; improve the reliability of intelligence; safely share information with warfighters and allies; and root out malicious insiders. One means to mitigating these challenges is to provide reliable knowledge of the provenance (i.e., lineage) of documents. This knowledge would allow, for instance, analysts to identify source information underpinning an intelligence report.

There are two primary approaches to establishing information provenance. First, we might seek to develop information systems or processes that track (through metadata) the source of data imported into new intelligence products. Unfortunately, information system heterogeneity makes such an approach largely impractical.

The more attractive alternative is to recover provenance knowledge as required by users. This approach is exemplified by plagiarism detection tools. The most common approach employed by these systems is to calculate and compare compact document fingerprints by hashing select substrings. Unfortunately these approaches are only effective when documents are near-duplicates [3]. The commercial Turnitin plagiarism detection service utilizes a different approach – relying on detecting long strings of words shared by co-derived documents. Regrettably this tactic is susceptible to errors when faced with heavily edited text or shared (but inconsequential) boilerplate text.

2 Efficiently and Accurately Identifying Text Provenance

In order to support new Intelligence Community applications and to overcome the shortcomings of past approaches, we have developed a new approach that enables the scalable comparison of the full text of documents. Fundamental to our InfoTracker system is the concept of a suffix tree [2]. Our contribution is in the development of a

[1] This work was supported by the ARDA/DTO Advanced Information Assurance program under contract NBCHC030077.

means to detect derivative text (and thereby information provenance) even when overlaps are much less pervasive than assumed by previous approaches.

We accomplish this by contrasting two distinct corpora, one composed of documents of interest (e.g., intelligence reports) and the other composed of background knowledge made up of, for instance, hundreds of thousands of randomly selected Web documents. By analyzing string overlaps in the light of general language usage, InfoTracker is better able to judge the likelihood that text strings could be produced independently. In other words, while previous approaches exploit the rareness of long strings (e.g., eight words or more), InfoTracker can exploit its greater knowledge of common text patterns to recognize much shorter strings of text that are likely coderived. This allows InfoTracker to succeed in a number of situations where past approaches would fail including identifying co-derivative relationships between paraphrases or between documents processed with different Optical Character Recognition (OCR) systems (each of which generates its own errors).

Enabling this new capability is a data structure based on String B-Trees [1], which is disk resident and allows incremental updating. Our approach also scales well – with the size of the index being linear in the size of the corpus, and document updates to the index and query parsing both being linear in the length of the new input.

3 Applications

In our poster and accompanying software demonstration we will provide details of our approach and describe how InfoTracker can be profitably employed in a wide variety of applications of import to the Intelligence Community. Some of these include:

- Identifying documents underpinning integrated intelligence products. This would allow analysts to independently verify and more easily reuse component data.
- Detecting malicious insiders. Infotracker can detect attempts to place classified material within documents with unclassified labels for exfiltration.
- Facilitating legitimate declassification. We have found, unexpectedly, that InfoTracker can effectively track the contents of short text redactions, and thereby improve the accuracy, consistency, and efficiency of future declassification tasks.
- Data spill recovery. Methods for identifying derivative text could be extremely helpful in determining where accidentally released intelligence has reached.

References

1. Ferragina, P. and Grossi, R. “The String B-Tree: A New Data Structure for String Search in External Memory and its Applications.” Journal of the ACM 46(2):236-280 (1999).
2. Gusfield, D. Algorithms on strings, trees, and sequences: computer science and computational biology, Cambridge University Press (1997).
3. Hoad, T. & Zobel, J. “Methods for identifying versioned and plagiarized documents,” JASIST 54(3), 203-215 (2003).