Mr. Tucker, Meet Mr. Shakespeare

Charles Nicholas
nicholas@umbc.dot.edu
CSEE Department, UMBC

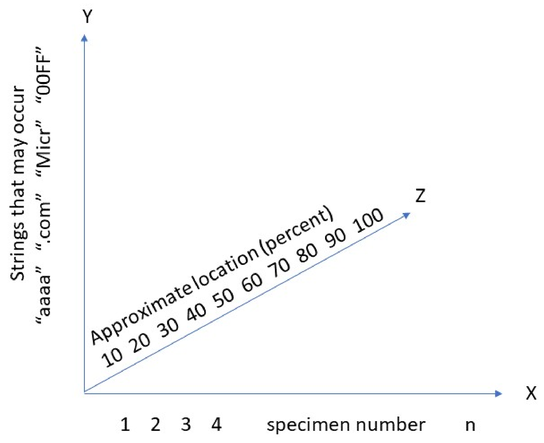
Objectives

To investigate the use of tensor decomposition in static malware analysis - on a large scale

- Malware analysis is often done, and taught, "in the small", that is, on one specimen at time [1]
- We need ways to do malware analysis "in the large"
- Malware specimens, in the form of executable binaries for the Windows platform, are abundant. Can we use tensor decomposition to gain insight into large collections of malware? We selected a specific malware family, the well-known Zeus Trojans [2], as test subjects.

Introduction

There are many ways to build a tensor for such objects, but we chose something simple: tabulate the occurrence of specific 4-grams (of which there are *many*) in a 3-d array, with X being the specimen ID, y being a 4-gram that may or may not occur in a given specimen, and z being that 4-gram's relative position within the specimen. So in the 3-d array shown below, entry $x_{i,j,k}$ is how many times 4-gram j occurs, in decile k, of specimen i.



Tucker Decomposition [3]

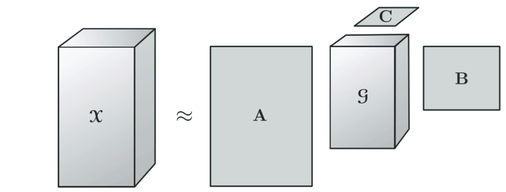
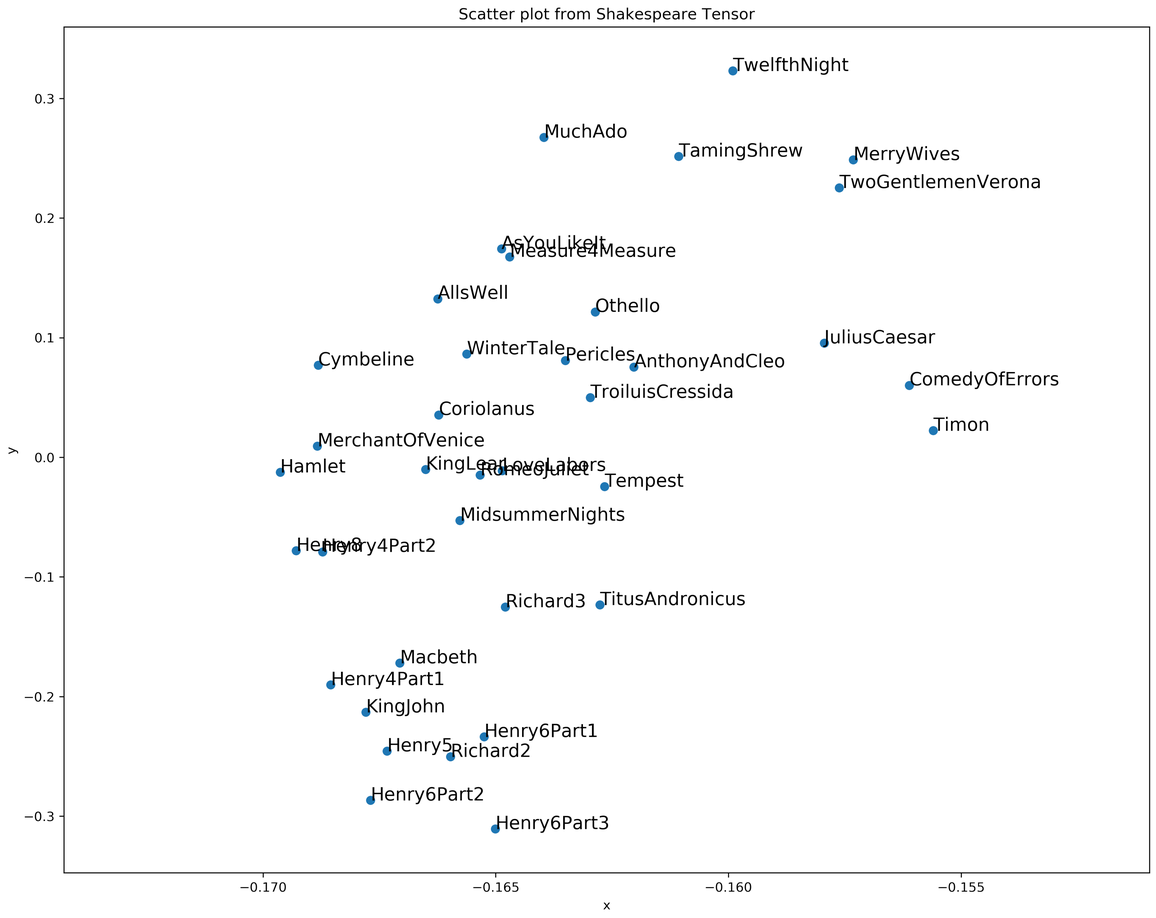


Fig. 4.1 *Tucker decomposition of a three-way array.*

The Matrix A from Tucker Decomposition



Sanity Check - Shakespeare

Before jumping into the Zeus data, we wanted to try it with a smaller corpus - the Shakespearean plays. [4] Using Python packages `tensorflow` (to do the tensor calculations) and `sklearn` (to parse the text data), in a Jupyter Notebook, we built the tensor X and ran both HOSVD and HOOI versions of Tucker.

Observations

- In the Shakespearean tensor X, entry $x_{i,j,k}$ is the number of times word j occurs in Act k of play i. The value of i ranges from 1 to 37, j ranges from 1 to about 30,000, and k ranges from 1 to 5. The tensor is quite sparse.
- HOSVC and HOOI gave similar results
- Pleased with the unsupervised clustering, especially of the history plays.

Research Continues

- Malware binaries will have *many* more terms, so we need to be selective.
- Only some of the Zeus binaries are unpacked, but focus on those first.

References

Michael Sikorski and Andrew Honig.
Practical Malware Analysis.
no starch press, 2012.

Abedelaziz Mohaisen, Omar Alrawi, and Omar Mohaisen, Abedelaziz Alrawi.
Unveiling Zeus: automated classification of malware samples.
In *Proceedings of the 22nd international conference on World Wide Web companion*, pages 829–832, 2013.

Tamara G. Kolda and Brett W. Bader.
Tensor Decompositions and Applications.
SIAM Review, 51(3):455–500, 2009.

Phani Teja Kesha.
Detection of Malware using Tensor Decomposition.
Technical report, UMBC M.S. Writing Project, 2019.

Acknowledgements

This work is supported by the Laboratory for Physical Sciences.
Presented at the 2019 High Performance Computing and Data Analytics Workshop