CS 533 Information Retrieval Systems

Partial Duplicate Detection Using Parallel Processing

Short Project Description

Hamzeh Ahangari

ID: 21303606

Computer Engineering Department

Bilkent University

Onur Aydın

ID: 20902097

Computer Engineering Department

Bilkent University

1. Description of the problem

- Duplication detection is the process of finding similarity between huge collection of books.
- Each consists of thousands of words.
- DUPNIQ is an efficient solution [1]:
  - Each book is represented by sequence of unique words, keeping their orders
  - Longest Common Subsequence (LCS) algorithm is used to detects the similarity between sequence of unique words.

2. Motivation and Importance

- Every year new editions of same books are published
- Libraries or companies (Amazon etc.) may be interested to run duplication detection application regularly.
- Cheating detection: Detecting copied paragraph in papers

3. Methodology

- Stage1: Changing Representation of Characters
- Stage 2: Efficient Parallel Processing

3.1. Changing Representation

- Comparing Characters is too costly
- Eigenvalue based compression

$$Av = \lambda v$$

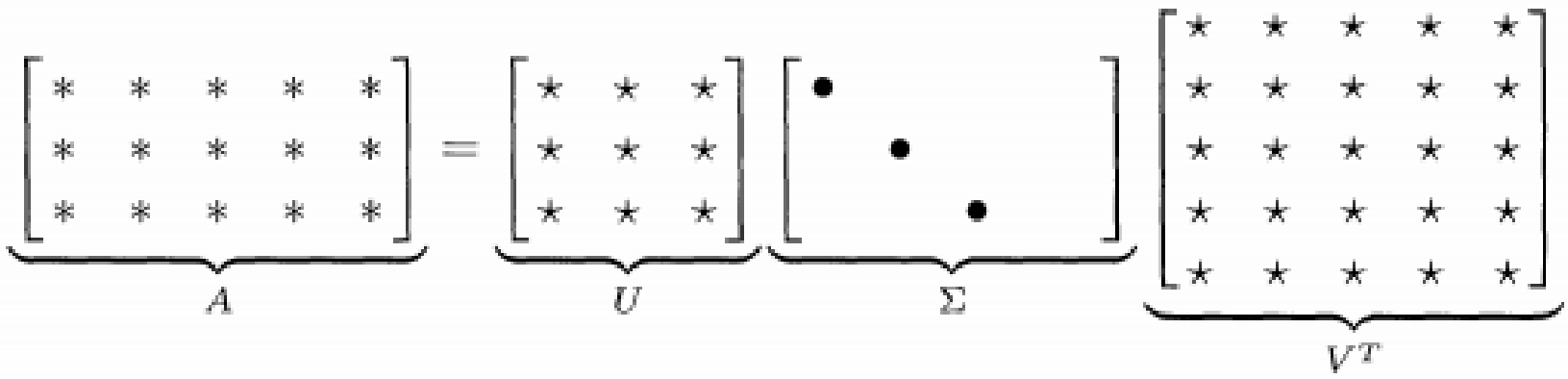
Hole	⇔	Hole
Ground		Ground
Nasty		Wet
Dirty		Dirty
Hobbit		Nasty
Hole		Comfort
Sandy		Bare

$\lambda_1 \Leftrightarrow \lambda_2$

3.1. Changing Representation

- Singular Value Decomposition (SVD)
  - Not all matrices are square

$$A = U\Sigma V^T$$



- Principal Component Analysis (PCA)

3.2. Parallel Computing

- LCS algorithms are investigated for potential parallelism opportunities.
  - Dependencies
  - Pipelining

- Use of multiple customized hardware core (co-processors) instead of multiple conventional parallel processors.

- This hardware platform will be more efficient in run-time, power, area and most importantly cost.

4. Expected Results

- Compressed representation: Because, number of bit comparisons decreases, pace of searching is expected to increase.

- Platform: Since our platform benefits from hardware specially designed for, we expect our platform to be by far more efficient and affordable than an expensive supercomputer to quickly find duplication among books.

5. References

- Yalniz, I. Z., Can, E. F., & Manmatha, R. (2011, October). Partial duplicate detection for large book collections. In *Proceedings of the 20th ACM international conference on Information and knowledge management* (pp. 469-474). ACM.
- Kumar, C. A., & Srinivas, S. (2006). Latent semantic indexing using eigenvalue analysis for efficient information retrieval. *Int. J. Appl. Math. Comput. Sci*,*16*(4), 551-558.
- Bryan, K., & Leise, T. (2006). The $25,000,000,000 eigenvector: The linear algebra behind Google. *Siam Review*, *48*(3), 569-581.
- Langville, A. N., & Meyer, C. D. (2005). A survey of eigenvector methods for web information retrieval. *SIAM review*, *47*(1), 135-161.
- Internet Archive. http://www.archive.org, 2010.
- Project Gutenberg. http://www.gutenberg.org, 2010.
- Hunt, J. W., & Szymanski, T. G. (1977). A fast algorithm for computing longest common subsequences. *Communications of the ACM*, *20*(5), 350-353.