Building a General Search Engine for Unstructured Data

Harold Trease
Senior Research Scientist
ICTAS HUME & SEEC Centers & Computer Science Department
Virginia Tech
Why Build a Search Engine?

- Lots of data that isn’t currently searchable because there is no metadata associated with it
  - image, (noisy) audio, video, communications, IoT

- Multi-INT Big Data -- The majority of unstructured data is “not” text
  - 88% Image/Video, 10% Audio/Communications, 2% Text
  - Video consumes ~75% of Internet bandwidth today (~85 - 90% 2018, Cisco)

- Help analysts/users find the information they want (actually, we provide pointers to information):
  - Google search for any type of digital data

- Multi-Sensor data fusion

- Making use of the structure of data (i.e., galaxy plots)

- Search for people, places, and things including their interactions (i.e., graphs)
Search Engine Technology

• Algorithms and Data Management:
  • Ingest: Unstructured data, compressed. Parallel I/O
  • Signature generation: Transform unstructured data into signatures
  • Signature comparison: Pairwise comparison of “unknown” signature with “known” signatures (database)
  • Connecting signatures into graphs
  • Results and Metrics: Not designed to be 100% accurate, but uses high probabilities
  • User Interfaces and user interaction

• Optimization:
  • Algorithms
  • Hardware and Code Performance
  • Usability

• Hybrid, Heterogeneous Parallel Computing
  • MPI, Hadoop/MapReduce
  • OpenMP, Cuda, OpenCL
Searching for Objects in Image/Video

Two typical frames from cell phone videos

⇐ Search for these objects, called “search criteria”. Wu, Hao, Kaixi, purple frisbee, building
Search for Wu: http://198.82.148.84/datafission/results/synergy_wu/html_search/

Search Criteria:
- Full frame results (ranked)
- Chipout results (ranked)
- Results with metadata
- File, time, frame #

Search Analyze Results:
Source:
/astro/www-data/mediaData/processedData/Synergy/Wu_01.jpg/Wu_01.jpg

Targets:
/astro/www-data/mediaData/processedData/Synergy/20140815_161203.mp4/20140815_161203.mp4
/astro/www-data/mediaData/processedData/Synergy/20140815_161444.mp4/20140815_161444.mp4

Table of top search frame results: 50
Search for a building/place: http://198.82.148.84/datafission/results/synergy_building/html_search/

Search Criteria

- Searchable Corpus

Search Results:

- Full frame results (ranked)
- Chipout results (ranked)
- Results with metadata
- File, time, frame #
Google Image Search vs. VT Search Engine

**Search Query**

**Google Image Search**

![Google Image Search](image)

**Bing Image Search**

![Bing Image Search](image)

**VT Search**

![VT Search](image)
Compound Search (i.e., more than one search criteria)

Google Image Search: (Google doesn't support multiple image search criteria)

(compound) VT Search

Search Query

Source:

Search Analyse Results:

Target:

Table of top search frame results: 50
Search for RB #22 "and" the football

Rank 1 & 2

Rank 9 & 11

Rank 3 & 4

Rank 15 & 16
Pattern-of-Life: Have three vehicles ever been in the same scene at the same time?

Search for 3 cars using Boolean “and” /”or” operators

(“and” operator)

(“or” operator)

(“or” operator)

(Compase_Clip_7_Red_Car.mpeg)

TOC
Search for the StateFarm logo in a set of 2012 and 2013 basketball highlight videos
Basic Abstraction: Generalized pattern search. Everything is represented as a signal/image.
**Signatures: Quantitative Information Representation**

**What do signatures look like?**

<table>
<thead>
<tr>
<th>Statistical Data Analysis</th>
<th>Shannon Entropy [P log(P)]</th>
<th>Spatial Frequencies (DoL)</th>
<th>Spectral Frequencies (FFT)</th>
</tr>
</thead>
<tbody>
<tr>
<td>80-D Signature Feature Vector [4 transforms * 4 spectral * 5 statistical moments]</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>20-D Native</th>
<th>20-D Entropy</th>
<th>20-D Spatial Frequency</th>
<th>20-D Spectral Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grey</td>
<td>Red</td>
<td>Green</td>
<td>Blue</td>
</tr>
</tbody>
</table>

**Signature Dope Vector:** 151 80 V:20#E:20#S:20#F:20# 66.45 57.47 0.65 2.43 4.02 91.99 91.18 0.69 1.98 2.59 55.02 51.40 1.02 3.72 9.03 53.25 50.39 1.20 4.27 11.36 4052.87 258199.4 2
Signatures: Why these components? Necessary/Sufficient?

80-D Signature Feature Vector [4 transforms * 4 spectral * 5 statistical moments]

20-D Native

<table>
<thead>
<tr>
<th>Grey</th>
<th>Red</th>
<th>Green</th>
<th>Blue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Five Statistical Moments</td>
<td>Five Statistical Moments</td>
<td>Five Statistical Moments</td>
<td>Five Statistical Moments</td>
</tr>
</tbody>
</table>

20-D Entropy

<table>
<thead>
<tr>
<th>Grey</th>
<th>Red</th>
<th>Green</th>
<th>Blue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Five Statistical Moments</td>
<td>Five Statistical Moments</td>
<td>Five Statistical Moments</td>
<td>Five Statistical Moments</td>
</tr>
</tbody>
</table>

20-D Spatial Frequency

<table>
<thead>
<tr>
<th>Grey</th>
<th>Red</th>
<th>Green</th>
<th>Blue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Five Statistical Moments</td>
<td>Five Statistical Moments</td>
<td>Five Statistical Moments</td>
<td>Five Statistical Moments</td>
</tr>
</tbody>
</table>

20-D Spectral Frequency

<table>
<thead>
<tr>
<th>Grey</th>
<th>Red</th>
<th>Green</th>
<th>Blue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Five Statistical Moments</td>
<td>Five Statistical Moments</td>
<td>Five Statistical Moments</td>
<td>Five Statistical Moments</td>
</tr>
</tbody>
</table>

Statistics: If two signatures are going to match, then the statistical moments "necessarily" have to be similar.

Entropy: Entropy of man-made vs. natural objects is a discriminator. Natural objects tend to have higher entropy. Man made objects tend to have lower entropy because of (unnatural) uniformity.

Spatial frequencies: Edges, curvature, corners are important discriminators for the human vision system.

Spectral frequency: Used as a discriminator, even though the audible frequency distribution may not mean anything to a human or DSP algorithm.
Signatures: Computational Kernels

<table>
<thead>
<tr>
<th>Statistical Data Analysis</th>
<th>Shannon Entropy [P()log(P())]</th>
<th>Spatial Frequencies (DoL)</th>
<th>Spectral Frequencies (FFT)</th>
</tr>
</thead>
<tbody>
<tr>
<td>80-D Signature Feature Vector [4 transforms * 4 spectral * 5 statistical moments]</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>20-D Native</th>
<th>20-D Entropy</th>
<th>20-D Spatial Frequency</th>
<th>20-D Spectral Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grey</td>
<td>Grey</td>
<td>Grey</td>
<td>Grey</td>
</tr>
<tr>
<td>Red</td>
<td>Red</td>
<td>Red</td>
<td>Red</td>
</tr>
<tr>
<td>Green</td>
<td>Green</td>
<td>Green</td>
<td>Green</td>
</tr>
<tr>
<td>Blue</td>
<td>Blue</td>
<td>Blue</td>
<td>Blue</td>
</tr>
</tbody>
</table>

| Kernel 1: Mean |
| Kernel 2: Generating function for variance, skew, kurtosis, hyper-skew |
| Kernel 3: Histogram and Histogram to P() normalization |
| Kernel 4: P()log(P()) |
| Kernel 5: Difference-of-Laplacians (DoL) edge detection filter. 5-Point / 9-Point Stencil |
| Kernel 6: 1-D FFT |
Exploiting the Structured Data

Haiti Videos

Food Science Video Data

YouTube Face Database

HG78 Genome

Westminster Video Data

RF Wireless Spectrum

Figure 5.2: PCA of BM waveforms, view 2
Visualization of Data: classification, clustering, graph analysis, summarization/search results
1) Sources: Getting data
- Sensor networks
- Prototypical field data
- Simulated data (more control over SNR, format, volume)

2) Database: Storing data
- Streaming vs. archive
- Ingest, codecs, multi-INT
  - Ingest and indexing/summarization: image, video, audio, sensor data, binary data, text (Transformation into N-dimensional signatures)
  - Ingest and DSPs: GNU Radio

3) Knowledge Base: Storing things
- Hierarchical associative memories: Remembering things
- Sparse Representations
- PDFs (e.g., histograms)
- NoSQL, SQL, Triple/quad stores
  - Hierarchical representation: subdivision and aggregation of entities
  - Types of metadata: Time, location, mode, modality, frequency of occurrence, order of occurrence
  - Modal metadata: Image, Video, Audio, WiFi, GSM, Radar, etc.

- Supervised/Unsupervised search and graph analysis
- Content vs. metadata

5) Interaction Graphs: Interrelating things
- Links, associations, connections, relationships, networks
- Relating people, places, and things to events, activities
- Patterns-of-Life
- Context and Situational Awareness:
  - Metadata (time, (geo)location, attribution)

6) Conflict Resolution: Resolving conflicting information
- Lack of conflict resolution leads to false positives

7) Trending and Prediction:
- Big Data vs. Models vs. Deep data analytics

8) Belief Propagation: Inferring things
- Graph Analysis:
  - What/who is around (centrality)
  - What/who points to me (pagerank)
  - PGMs, Bayesian networks

9) Testing and evaluation: V&V&UQ
- Real data (Validation)
- Synthetic data (Validation)
- Ideal data (Verification)
- Uncertainty quantification
  - How good is the information
  - Accuracy of the solutions
D-Wave Adiabatic Quantum Computer

- Computes all results at once, reads out one solution in about 100msec.
- Crudely similar to simulated annealing.
- Express problem as a QUBO (Quadratic Unconstrained Binary Optimization).
- Transformed into a QMI (Quantum Machine Instruction).
- Results read out as a binary bit vector.
- Application areas:
  - Shor’s Algorithm (exponential speedups)
    - Prime number factorization
  - Grover’s Algorithm (SQRT(N) speedup)
    - Unstructured search
- Application areas:
  - Traveling salesman problem
  - Knapsack problem
  - Machine learning
Simple program (sample.toq)

```toq
# --- sample.toq --- Regression test
#
bool: @a, @b
bool: @c, @d

#~~~
assert: And( Or(@a,@b), Or(@c,@d) )
assert: @c !== @d
assert: (@a+@d >= (@b+@c))
#~~~
end:
```

sample2.toq

```toq
# --- sample2.toq --- Regression test
#
bool: @a, @b
bool: @c, @d
real: x, y, z

#~~~
x = pi
y = sqrt(x+2)
z = x-y
assert: And(Or(@a,@b),Or(@c,@d))
if: ((x*y)/3 > 3.11)
    assert: @c !== @d
endif:
assert: (@a+@d >= (@b+@c))
#~~~
printvars:
end:
```

sample2.toq (results)

```
<table>
<thead>
<tr>
<th>Variable</th>
<th>#</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1)</td>
<td>@a</td>
<td>1</td>
</tr>
<tr>
<td>2)</td>
<td>@b</td>
<td>0</td>
</tr>
<tr>
<td>3)</td>
<td>@c</td>
<td>0</td>
</tr>
<tr>
<td>4)</td>
<td>@d</td>
<td>1</td>
</tr>
</tbody>
</table>
```

1. Enter "toq =QM" for an explanation of why (Sum(occurrences) == cases)
Unstructured Search on the D-Wave Adiabatic Quantum Computer

- Relax a surface onto the surface of “unknown” signatures minus “database” signatures
Virginia Tech Search Engine (VTSE) Platform Architecture

Client/Server Interoperability
- M2M Client-Server Interfaces:
  - Build using LAMP
  - HTTP GUI
  - RESTful C++/Curl, Python, Java, and PHP
  - API Client-Server Interface for PHP and C++
  - Inherits the security domain from client
  - Deployable to sensitive environments

Web Browser
- HTTP GUI Client Interface

Mobile Browser/Apps
- HTTP GUI Client Interface

Rich Client
- REST Web Client Interface

Data-Services:
- Data from clients
- Results from database
- Bulk data loads

Server
- Video Server:
  - Algorithm
  - TOC, Index, Search
  - Trending/Deep Analytics/Prediction
  - Parallel Computing
  - Data Management

- Python API Server:
  - Index & Summarize
  - Compare
  - Find/Search
  - Tracking
  - Cleanup

Database Management:
- MySQL (metadata)
- MySQL (results)
- HyperTable/Hadoop/HDFS