Pears and PPP-Hashing

Behrang QasemiZadeh, Laura Kallmeyer, Aurelie Herbelot

zadeh@phil.hhu.de
Self-Awareness
Memory, and Reasoning System
Environment and stimuli
Input Sensory

Read
Facial Expression
Speech
Response

Human Agent
Some can be sensed with input sensory of other agents.

Some cannot be sensed by input sensory of other agents.

Some can be probed from traces.
Von Neumann’s Computer Robots

- Output Devices
- Arithmetic / Logic Unit (Reasoning)
- Control Unit
- Memory
- Input Devices
Human Language Technology
Human Language Technology is what it is ...

- human checking
- human computer interaction
- human conversation
- human error analysis
- human gene identification
- human interference
- human language acquisition
- human language generation
- human language processing
- human machine interface
- human query formulation
- human sentence processing
- human speech processing
- human-computer communication
- human-computer dialogue
- human-computer interaction
- human-computer interface
- human-human-system interaction
- human-like dialogue
- human-like learning
- human-machine communication
- human-machine dialog
FRANK PASQUALE
THE BLACK BOX SOCIETY
The Secret Algorithms That Control Money and Information

Google

Transparency
About PeARS

PeARS (Peer-to-peer Agent for Reciprocated Search) is a lightweight, distributed search engine. It relies on people going about their normal business and browsing the Web. While they do so, the pages they visit are indexed in the background and assigned a 'meaning' (is this page about cats, fashion, ancient history? what's the 'gist' of the document?) From time to time, they can choose to share some or all of these meanings with others, providing the building stones of a giant search engine network, distributed across users.

Think of PeARS as a layer of virtual agents underlying a community of real people. Your virtual agent is responsible for sharing your Web knowledge in the way you choose, and for contacting other people’s agents to help you answer your questions. All completely automatically.
Finding PeArs ... Which one?!
Let’s get more pragmatic

Whatever PeArs is, and whatever policy it uses for finding peers, it must be programmed!

We need to think in terms of data structure, data flow, memory, CPU, ...
Processing Natural Language Text (Communication Mediums)

• For Computers, these are simply a sequence of bits and bytes.
Processing Natural Communication Mediums (natural language text)

• For Computers, these are simply a sequence of bits and bytes.
Vector Space Model (VSM)

• Vector spaces are one of the models employed for text processing.
  • A Mathematical Structure $\langle V, \mathbb{R}, +, \cdot \rangle$ that satisfy certain axioms.

• Text elements are converted to real-valued vectors.

• Each dimension of the vector represents some information about text elements.
VSMs: an Example

• In a text-based information retrieval task:
  • Each dimension of the VSM represents an index term $t$.
  • Documents are represented by vectors.

Salton et. al. (1975)
VSMs: an Example

• In a text-based information retrieval task:
  • Each dimension of the VSM represents an index term $t$.
  • Documents are represented by vectors.

Salton et. al. (1975)
VSMs: an Example
VSMs: an Example

• In a text-based information retrieval task:
  • Each dimension of the VSM represents an index term $t$.
  • Documents are represented by vectors.
  • Queries are treated as pseudo-documents.

Salton et. al. (1975)
VSMs: an Example

Salton et. al. (1975)
VSMs: an Example

• In a text-based information retrieval task:
  • Each dimension of the VSM represents an index term $t$.
  • Documents are represented by vectors.
  • Queries are treated as pseudo-documents.
  • Using a norm structure, a notion of distance is defined and used to assess the similarity between vectors, i.e. documents and queries.
VSMs: an Example

The L2 or Euclidean norm, \( \| \mathbf{v} \|_2 = \sqrt{\sum_i v_i^2} \)

\[
\begin{align*}
\text{dist}_2(d_1, q) &= \|d_1 - q\|_2 = \sqrt{(4 - 3)^2 + (4 - 1)^2} = \sqrt{10} \\
\text{dist}_2(d_2, q) &= \|d_2 - q\|_2 = \sqrt{(1 - 3)^2 + (2 - 1)^2} = \sqrt{5}
\end{align*}
\]

Salton et. al. (1975)
VSMs: an Example

The L1 norm, $\|v\|_1 = \sum_i |v_i|$ 

$\text{dist}_1(d_1, q) = \|d_1 - q\|_1 = |(4 - 3)| + |(4 - 1)| = 4$
$\text{dist}_1(d_2, q) = \|d_2 - q\|_1 = |(1 - 3)| + |(2 - 1)| = 3$
VSMs: an Example

The L1 norm, \( \|v\|_1 = \sum_i |v_i| \)

Finding peers for a PeArs is similar, isn’t!?
The dimensionality barrier

Index Terms
Context words
Visited URLs, etc.
The dimensionality barrier
Overcoming the dimensionality barrier

- Dimensionality Reduction Techniques are employed to resolve the problem.

```
  Dimensionality Reduction
    \---------------------------\   
    |                          |   
    |  Selection Process (based on a heuristic) \   
    |                                          |   
    \----------------------------------------\   
                  Transformation-based methods

  \---------------------------\   
  |                          |   
  |  Locality Sensitive Hashing \   
  \---------------------------\   
```

...
Locality Sensitive Hashing

• **IDEA:** Instead of building high-dimensional vectors, build *signatures.*
  • Signatures = Vectors of Reduced Dimensionality
  • Signatures = Hash codes which are similar for similar items!

• **The Goal of LSH:**
  • If the (L2) distance between V1 and V2 in the original space is X, then the
distance between their signatures must be bounded to X + epsilon.
  • Preserve topological order of points: If V1 is closer to V2 than V3, then this
  relationship must also hold between their signatures.

• **Result:** Much less computations for finding similar items (e.g., as used
  in approximate nearest neighbor search, deduplication, etc.)
LSH as put by Victor Lavrenko
(The University of Edinburgh)

Locality Sensitive Hashing

1. Want similar hashcodes for nearby points
2. Generate random hyperplanes: $h_1, h_2, h_3$
3. Hash-code for $a: H_{100}[a h_1 a h_2 a h_3] = 100$
4. Compare $a$ to points with same hash-code
   - $b$... indeed similar to $a$
   - $d$... false positive, will be eliminated
   - $c$... different hash-code, will miss it
5. Repeat w. different hyperplanes: $h_4, h_5, h_6$

Computational cost:
- $N$ points, $D$-dimensional, $K$ hyperplanes
- $E K$... find bucket where point lands
- $N/2^K$... points in that bucket (on avg)
- $D N/2^K$... cost of comparisons
- repeat everything $L$ times (#tables)

LSH: $LEK + LDN/2^K \rightarrow O (\log N \# K - \log N)$
Index: $D (ND) / \rightarrow O (N)$
Brute-force: $DN \rightarrow O (N)$

Source: https://www.youtube.com/watch?v=Arni-zkqMBA
Many roads that lead to the same destination...

- Johnson and Lindenstrauss
- Robert Hecht-Nielsen
- Pentti Kanerva
- Piotr Indyk
- Sanjoy Dasgupta
- Ping Li
- ...

Let’s look at an example...

- **The MEN relatedness test** consists of 3000 pairs of words.
  - Using a scale of 0 to 50, each pair is rated as “semantically related” by human.
  - Human ratings are used to obtain a ranked list of word pairs.
    - sun-n sunlight-n 50.000000
    - automobile-n car-n 50.000000
    - river-n water-n 49.000000
    - ...
    - cafe-n frog-n 4.000000
    - bakery-n zebra-n 0.000000
Let’s look at a particular example…

• The MEN relatedness test consists of 3000 pairs of words.
  • Using a scale of 0 to 50, each pair is rated as “semantically related” by human.
  • Human ratings are used to obtain a ranked list of word pairs.
    • sun-n sunlight-n 50.000000
    • automobile-n car-n 50.000000
    • river-n water-n 49.000000
    • cafe-n frog-n 4.000000
    • bakery-n zebra-n 0.000000

• EXCEPTION (Figure of Merit): A *good* algorithms must generate a similar ranking as human!

(source: Pecier Dicierdo; http://filipinofreethinkers.org/2012/06/23/turings-tremendous-talent-and-trenchant-test/.)
How to Generalize Relatedness to Pears?

- sun-n sunlight-n 50.000000
- automobile-n car-n 50.000000
- river-n water-n 49.000000
- cafe-n frog-n 4.000000
- bakery-n zebra-n 0.000000
Crude L2 Distances are *Insufficient*!

![Graph showing Spearman’s Correlation ρ vs Context Window Size]

- **State-of-the-Art**
- **Count-based+Cos**
So are LSH techniques for L2-normed spaces!

RI is an implementation of LSH using Gaussian Kernels.
PPP-Hashing

Let’s forget about normed spaces, Bayesian mathematics, etc.

• After all, there are evidence that they may not work/model our world as expected!
PPP-Hashing

Let’s forget about normed spaces, Bayesian mathematics, etc.

• After all, there are evidence that they may not work/model our world as expected!
  • Proof: Are Brexit and President Donald Trump enough for you?
    • Why the actual result was different from what was anticipated/predicted/expected?
PPP-Hashing: Philosophy

1) Distributional Hypothesis:
   • [Linguistic] items with similar distributions have similar [meanings]
     • Diana Maynard: You shall know a person by the company s/he keeps!
1) Distributional Hypothesis:
   - [Linguistic] items with similar distributions have similar [meanings]
   - Diana Maynard: You shall know a person by the company s/he keeps!
PPP-Hashing: Philosophy

1) Distributional Hypothesis:
   • [Linguistic] items with similar distributions have similar [meanings]
   • Diana Maynard: You shall know a person by the company s/he keeps!
1) The Randomized Distributional Hypothesis:
   • [Linguistic] items with similar distributions over randomly partitioned event spaces have similar [meanings]
PPP-Hashing: Philosophy

1) The Randomized Distributional Hypothesis:
   • [Linguistic] items with similar distributions over randomly partitioned event spaces have similar [meanings]
Claim:
The smaller vector
is as good as, or even better than, the larger vector!
Implementation

• Set a size for signature, i.e., an array of bytes
  • E.g., 500 bytes.

• Map your symbol under investigation to an empty array
  • E.g., a Person in Pears network.

• Update signatures by event occurrences:
  • E.g., a peer in Pears network visit a URL.

• At any time, compute signatures’ correlation to get a measure of similarity/relatedness between them!
Updating Signatures

signature[modulos(a_hash_function(event), len(signature))]++;

That is it!
int event_hashcode = Jenkins_hash_function(byte[] event);
int target_byte = mod(event_hashcode, length(signature));
signature[target_byte]++;

Updating Signatures
Updating Signatures

```c
int event_hashcode = Jenkins_hash_function(byte[] event);
int target_byte = mod(event_hashcode, length(signature));
signature[target_byte]++;

uint32_t jenkins_one_at_a_time_hash(char *key, size_t len){
    uint32_t hash, i;
    for(hash = i = 0; i < len; ++i) {
        hash += key[i];
        hash += (hash << 10);
        hash ^= (hash >> 6);
    }
    hash += (hash << 3);
    hash ^= (hash >> 11);
    hash += (hash << 15);
    return hash;
}
```
Computing correlations, 
*aka*, Signature Comparator:

- Cast byte arrays to unsigned numbers (it is just the matter of perception).
- Use any standard correlation coefficient measure to compute similarities:
  - Kendall’s Tau;
  - Pearson’s Rho;
  - etc.

\[
\rho_{X,Y} = \text{corr}(X, Y) = \frac{\text{cov}(X, Y)}{\sigma_X \sigma_Y}
\]

\[
r = \frac{1}{n-1} \sum \frac{(X - \mu_X)(Y - \mu_Y)}{\sigma_X \sigma_Y}
\]

- Lessons learnt from past experiments:
  - be AVID! Any correlation measure reveals an aspect of similarity!
    - Each measure unravel some regularities between signatures!
A big gap between the state-of-the-art neural mappings and and the Crude PPP-Hashing!
Do not worry!
There are solutions to shorten the GAP!
First, what is the cause?

Counting \textit{catdog} in two different buckets is inherently wrong: although they are two different persona, bumping to cat/dog always entail bumping to dog/cat
First, what is the cause?

Noise is inevitable!
1) To get rid of double counts;
2) To cancel the effect of noise.

Normalize and truncate event counts by their strength of association:
- odds ratio,
- relative risk,
- positive-pointwise mutual information,
- ...

\[
\text{pmi}(x; y) = \log \frac{p(x, y)}{p(x)p(y)} = \log \frac{p(x|y)}{p(x)} = \log \frac{p(y|x)}{p(y)}.
\]
Why PPP-Hashing?

• Can we use weighting on LSH techniques to improve their performance?

• Can we build neural-optimized mappings that are developed separately on different data points but are still comparable to each other?

• Can we translate neural-signatures to the original symbolic gist of events?

• What about training time? Can we skip it?

• Can we access signatures while they are learnt?

• Can we update neural signatures at any time? If yes, are they still going to give an optimised mapping?
Why PPP-Hashing?

• Small computational and memory complexity:
  • Highly scalable UP and OUT!
  • Suitable for mobile devices.

• Incremental: Signatures can be updated at any time while they are used.
  • Remember PeARSs are highly active nodes (they learn, they change, etc.)

• Provides a simple mechanism to guarantee interoperability:
  • Use the same hash function when generating signatures.
  • Note: this can be employed as a method for encryption, too.
Interoperability

• How to make sure signatures from different PeARSs are comparable?
  • That is to say, how to make sure that signatures are built with respect to the same reference? (after all, DM is built on top of relative frequencies).

Cosine as similarity measure: Pairwise commonalities.
Interoperability (between PeArs)

- How to make sure signatures from different PeARSs are comparable?
- That is to say, how to make sure that signatures are built with respect to the same reference? (after all, DM is built on top of relative frequencies).
Interoperability at the Web scale!

• What if I want to share my signatures/vectors on the web?
• What if I want to switch from PeArs to TeArs?!
• How to ensure interoperability of signatures?
Is not the right time to call for the Web of Open Vectors?! 

- The “Web of Open Vectors” to complement the “Web of Open Data” aka Linked-Data.
- The “Web of Open Vector Standards” to foster transparency and interoperability?!
Some More Questions for PeArs:

• How PeArs can be different to other P2P search engines?
  • FAROO, Seeks, YaCy, Opencola, etc.

• How PeArs can be different from Intranet/Enterprise search engines?
  • Previous applications include expert finding, collaborative call centres, etc.

• How to make PeArs learn and adapt based on information seeking activities and human agent preferences?
  • And, how is that going be different from “Personalized Search Engines”?