Component Benchmark: The essentials of modern big data and AI workloads

Wanling Gao

ICT, Chinese Academy of Sciences & BenchCouncil

HPCA 2019, Washington D.C., USA
A scalable big data and AI benchmark suite

- Treat big data, AI and Internet service workloads as a pipeline of units of computation handling (input or intermediate) data

- Target: find the main abstractions of time-consuming units of computation (data motifs)
  - The combination of data motifs = complex workloads
    - Similar to Relational Algebra
  - Data motifs-based scalable benchmarking methodology

Scalable Benchmark Methodology

- **Traditional**: create each benchmark or proxy for every possible workload
- **Our**: **Data motif-based (Scalable)**
  - Micro Benchmark---*Single* data motif
  - Component Benchmark---*Data motif combination with different weights*
  - Application Benchmark---*End-to-end* application
BigDataBench Publications

- Data Motifs: A Lens Towards Fully Understanding Big Data and AI Workloads. PACT’18.
- Understanding Big Data Analytics Workloads on Modern Processors. TPDS’16
- Auto-tuning Spark Big Data Workloads on POWER8: Prediction-Based Dynamic SMT. PACT’16
- BigDataBench: a Big Data Benchmark Suite from Internet Services. HPCA’14
- CVR: Efficient Vectorization of SpMV on X86 Processors. CGO’18.
- Data Motif-based Proxy Benchmarks for Big Data and AI Workloads. IISWC 2018.
Component Benchmarks

- BigDataBench 5.0 has 15 component benchmarks
  - 6 big data component benchmarks
  - 9 AI component benchmarks

Diverse Software stacks

- Flink
- GraphX
- Spark
- MVAPICH
- Hadoop
- DataMPI
- NoSql
- TensorFlow
- ORACLE
- GraphLab
- PyTorch
- Caffe
- Impala
- Hive
- Shark
- Hadoop RDMA
- MPI
Application Diversity

- Image classification
- Image generation
- Text-to-Text Translation
- Image-to-Text
- Image-to-Image
- Speech-to-Text
- Face embedding
- Search Engine Indexing
- Object detection
- Recommendation
- PageRank
- Graphical Model
- Clustering
- Text Classification
- Feature Exaction
Content

- For each component benchmark
  - Workload type
  - Application domain
  - Dataset
  - Algorithms and involved data motifs
#1---Image classification

- Image classification
  - Extracting different thematic classes in the image
  - A fundamental problem in computer vision

- Component benchmark—Image classification
  - A supervised learning problem
    - define a set of target classes (objects to identify in images), and train a model to recognize them using labeled example photos
Dataset - ImageNet

- ImageNet
  - an image database organized according to the WordNet hierarchy
    - each node of the hierarchy is depicted by hundreds and thousands of images
  - 15 million labeled high-resolution images belonging to roughly 22,000 categories.
  - The images were collected from the web and labeled by human labelers.
  - Presented for the first time as a poster at CVPR 2009.

Deng, Jia; Dong, Wei; Socher, Richard; Li, Li-Jia; Li, Kai; Fei-Fei, Li (2009), "ImageNet: A Large-Scale Hierarchical Image Database", 2009 conference on Computer Vision and Pattern Recognition
ImageNet - ILSVRC

- ImageNet Large-Scale Visual Recognition Challenge (ILSVRC)
  - Held since 2010, as part of the Pascal Visual Object Challenge
  - uses a subset of ImageNet with roughly 1000 images in each of 1000 categories. In all, there are roughly 1.2 million training images, 50,000 validation images, and 150,000 testing images.
AlexNet

- On September 30, 2012, AlexNet achieved a top-5 error of 15.3%, more than 10.8 percentage points lower than that of the runner up.

  - “Suddenly people started to pay attention, not just within the AI community but across the technology industry as a whole” --- The Economist

- one of the most influential papers published in computer vision

- having spurred many more papers published employing CNNs and GPUs to accelerate deep learning
AlexNet Structure

- AlexNet contained eight layers
  - the first five: **convolutional** layers
  - some of them followed by **max-pooling** layers
  - and the last three were fully connected layers.
  - Non-saturating **ReLU** activation function
ResNet

- The most popular and wildly used model in modern times
- very deep convolutional neural network
- Residual neural networks utilize *skip connections* or *short-cuts* to jump over some layers
  - the vanishing/exploding gradients, which hamper the convergence

### ResNet Structure

<table>
<thead>
<tr>
<th>layer name</th>
<th>output size</th>
<th>18-layer</th>
<th>34-layer</th>
<th>50-layer</th>
<th>101-layer</th>
<th>152-layer</th>
</tr>
</thead>
<tbody>
<tr>
<td>conv1</td>
<td>112×112</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>7×7, 64, stride 2</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>3×3 max pool, stride 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>conv2.x</td>
<td>56×56</td>
<td>[3×3, 64]×2</td>
<td>[3×3, 64]×3</td>
<td>[1×1, 64]</td>
<td>1×1, 64</td>
<td>1×1, 64</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[3×3, 64]</td>
<td>[3×3, 64]</td>
<td>[3×3, 64]</td>
<td>[3×3, 64]</td>
<td>[3×3, 64]</td>
</tr>
<tr>
<td>conv3.x</td>
<td>28×28</td>
<td>[3×3, 128]×2</td>
<td>[3×3, 128]×4</td>
<td>[1×1, 128]</td>
<td>1×1, 128</td>
<td>1×1, 128</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[3×3, 128]</td>
<td>[3×3, 128]</td>
<td>[3×3, 128]</td>
<td>[3×3, 128]</td>
<td>[3×3, 128]</td>
</tr>
<tr>
<td>conv4.x</td>
<td>14×14</td>
<td>[3×3, 256]×2</td>
<td>[3×3, 256]×6</td>
<td>[1×1, 256]</td>
<td>1×1, 256</td>
<td>3×3, 256</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[3×3, 256]</td>
<td>[3×3, 256]</td>
<td>[3×3, 256]</td>
<td>[3×3, 256]</td>
<td>[3×3, 256]</td>
</tr>
<tr>
<td>conv5.x</td>
<td>7×7</td>
<td>[3×3, 512]×2</td>
<td>[3×3, 512]×3</td>
<td>[1×1, 512]</td>
<td>1×1, 512</td>
<td>1×1, 2048</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[3×3, 512]</td>
<td>[3×3, 512]</td>
<td>[3×3, 512]</td>
<td>[3×3, 512]</td>
<td>[3×3, 512]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[3×3, 512]</td>
<td>[3×3, 512]</td>
<td>[1×1, 2048]</td>
<td>[1×1, 2048]</td>
<td>[1×1, 2048]</td>
</tr>
<tr>
<td>1×1</td>
<td>average pool, 1000-d fc, softmax</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1×1, 1024</td>
<td></td>
<td>1×1, 1024</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1×1, 1024</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1×1, 1024</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>FLOPs</th>
</tr>
</thead>
<tbody>
<tr>
<td>conv1</td>
<td>1.8×10⁹</td>
</tr>
<tr>
<td>conv2.x</td>
<td>3.6×10⁹</td>
</tr>
<tr>
<td>conv3.x</td>
<td>3.8×10⁹</td>
</tr>
<tr>
<td>conv4.x</td>
<td>7.6×10⁹</td>
</tr>
<tr>
<td>conv5.x</td>
<td>11.3×10⁹</td>
</tr>
</tbody>
</table>

#2---Image generation

- Unsupervised learning component benchmark
  - Learn to mimic the distribution of data
Why Generative Model

- Training and sampling from generative models is an excellent test of our ability to represent and manipulate high-dimensional probability distributions.
- Can be incorporated into reinforcement learning in several ways.
- Can be trained with missing data and can provide predictions on inputs that are missing data:
  - semi-supervised learning, in which the labels for many or even most training examples are missing.
  - Generative models are able to perform semi-supervised learning reasonably well.
Generative adversarial networks (GANs)

- Introduced by Ian Goodfellow *et al.* in 2014

- “Most Interesting Idea in Last 10 Years” --- Yann LeCun
GANs

- A class of artificial intelligence algorithms used in unsupervised machine learning, implemented by a system of two neural networks contesting with each other in a zero-sum game framework.

- This technique can generate photographs that look at least superficially authentic to human observers, having many realistic characteristics.
GANs Network

- One network generates candidates (generator) and the other evaluates them (discriminator).
  - The generator takes in random numbers and returns an image.
  - This generated image is fed into the discriminator alongside a stream of images taken from the actual dataset.
  - The discriminator takes in both real and fake images and returns probabilities, a number between 0 and 1, with 1 representing a prediction of authenticity and 0 representing fake.
Algorithm---WGAN

■ WGAN
  ■ WGAN can improve the stability of learning, get rid of problems like mode collapse, and provide meaningful learning curves useful for debugging and hyperparameter searches
  ■ Using wasserstein distance to measure the distance between two probability distributions
  ■ Simplify the training of GANs

WGAN Structure

Generator

```python
class Generator(nn.Module):
    def __init__(self, ngpu):
        super(Generator, self).__init__()
        self.ngpu = ngpu
        self.main = nn.Sequential(
            # input is Z, going into a convolution
            nn.ConvTranspose2d( nz, ngf * 8, 4, 1, 0, bias=False),
            nn.BatchNorm2d(ngf * 8),
            nn.ReLU(True),
            # state size. (ngf*8) x 4 x 4
            nn.ConvTranspose2d(ngf * 8, ngf * 4, 4, 2, 1, bias=False),
            nn.BatchNorm2d(ngf * 4),
            nn.ReLU(True),
            # state size. (ngf*4) x 8 x 8
            nn.ConvTranspose2d(ngf * 4, ngf * 2, 4, 2, 1, bias=False),
            nn.BatchNorm2d(ngf * 2),
            nn.ReLU(True),
            # state size. (ngf*2) x 16 x 16
            nn.ConvTranspose2d(ngf * 2, ngf, 4, 2, 1, bias=False),
            nn.BatchNorm2d(ngf),
            nn.ReLU(True),
            # state size. (ngf) x 32 x 32
            nn.ConvTranspose2d(ngf, nc, 4, 2, 1, bias=False),
            nn.Tanh()
            # state size. (nc) x 64 x 64
        )
```

Generator

- strided two dimensional convolutional transpose layers
- a 2d batch norm layer and a relu activation
- The output of the generator is fed through a tanh function to return it to the input data range of $[-1,1]$. 
Discriminator

```python
class Discriminator(nn.Module):
    def __init__(self, ngpu):
        super(Discriminator, self).__init__()
        self.ngpu = ngpu
        self.main = nn.Sequential(
            # input is (nc) x 64 x 64
            nn.Conv2d(nc, ndf, 4, 2, 1, bias=False),
            nn.LeakyReLU(0.2, inplace=True),
            # state size. (ndf) x 32 x 32
            nn.Conv2d(ndf, ndf * 2, 4, 2, 1, bias=False),
            nn.BatchNorm2d(ndf * 2),
            nn.LeakyReLU(0.2, inplace=True),
            # state size. (ndf*2) x 16 x 16
            nn.Conv2d(ndf * 2, ndf * 4, 4, 2, 1, bias=False),
            nn.BatchNorm2d(ndf * 4),
            nn.LeakyReLU(0.2, inplace=True),
            # state size. (ndf*4) x 8 x 8
            nn.Conv2d(ndf * 4, ndf * 8, 4, 2, 1, bias=False),
            nn.BatchNorm2d(ndf * 8),
            nn.LeakyReLU(0.2, inplace=True),
            # state size. (ndf*8) x 4 x 4
            nn.Conv2d(ndf * 8, 1, 4, 1, 0, bias=False),
            nn.Sigmoid()
        )
```

Discriminator

- a series of Conv2d, BatchNorm2d, and LeakyReLU layers, and outputs the final probability through a Sigmoid activation function
Dataset - LSUN

- Large-scale Scene Understanding
  - about one million labelled images, classified into 10 scene categories and 20 object categories
# LSUN Category

<table>
<thead>
<tr>
<th>Category</th>
<th>Training</th>
<th>Validation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bedroom</td>
<td>3,033,042 images (43 GB)</td>
<td>300 images</td>
</tr>
<tr>
<td>Bridge</td>
<td>818,687 images (16 GB)</td>
<td>300 images</td>
</tr>
<tr>
<td>Church Outdoor</td>
<td>126,227 images (2.3 GB)</td>
<td>300 images</td>
</tr>
<tr>
<td>Classroom</td>
<td>168,103 images (3.1 GB)</td>
<td>300 images</td>
</tr>
<tr>
<td>Conference Room</td>
<td>229,069 images (3.8 GB)</td>
<td>300 images</td>
</tr>
<tr>
<td>Dining Room</td>
<td>657,571 images (11 GB)</td>
<td>300 images</td>
</tr>
<tr>
<td>Kitchen</td>
<td>2,212,277 images (34 GB)</td>
<td>300 images</td>
</tr>
<tr>
<td>Living Room</td>
<td>1,315,802 images (22 GB)</td>
<td>300 images</td>
</tr>
<tr>
<td>Restaurant</td>
<td>626,331 images (13 GB)</td>
<td>300 images</td>
</tr>
<tr>
<td>Tower</td>
<td>708,264 images (12 GB)</td>
<td>300 images</td>
</tr>
<tr>
<td><strong>Testing Set</strong></td>
<td><strong>10,000 images (173 MB)</strong></td>
<td></td>
</tr>
</tbody>
</table>
#3---Text-to-Text Translation

- The most important field of computational linguistics
- Translate text from one language to another

To read an input sentence “ABC” and produce “WXYZ” as the output sentence
Dominate Models

- Recurrent or convolutional neural networks that include an encoder and a decoder
  - RNN is trained to map an input sequence to an output sequence which is not necessarily of the same length.
    - The Encoder RNN reads the input sequence and generates the fixed-size context vector which represents a semantic summary of the input sequence.
    - The fixed-size context vector is given as input to the decoder RNN.
    - The fixed-size context can be provided as the initial state of the Decoder RNN, or it can be connected to the hidden units at each time step. These two ways can also be combined.
Transformer

- relying entirely on an attention mechanism to draw global dependencies between input and output
  - Attention mechanisms have become an integral part of compelling sequence modeling and transduction models in various tasks

Encoder

- The encoder is composed of a stack of $N=6$ identical layers. Each layer has two sub-layers. The first is a multi-head self-attention mechanism, and the second is a simple, position-wise fully connected feed-forward network. It employs a residual connection around each of the two sub-layers, followed by layer normalization.
Transformer structure (Cont’)

- Decoder
  - The decoder is also composed of a stack of $N = 6$ identical layers. In addition to the two sub-layers in each encoder layer, the decoder inserts a third sub-layer, which performs multi-head attention over the output of the encoder stack. Similar to the encoder, we employ residual connections around each of the sub-layers, followed by layer normalization.
Dataset

- WMT English-German from Bojar, O.; Buck, C.; Federmann, C.; Haddow, B.; Koehn, P.; Monz, C.; Post, M.; Specia, L., ed. (2014), Proceedings of the Ninth Workshop on Statistical Machine Translation, Association for Computational Linguistics, Baltimore, Maryland, USA.

Train (4.5M sentence pairs): [train.en] [train.de]

Test: [newstest2012.en] [newstest2012.de] [newstest2013.en] [newstest2013.de] [newstest2014.en] [newstest2014.de] [newstest2015.en] [newstest2015.de]

Vocabularies (top 50K frequent words): [vocab.50K.en] [vocab.50K.de]

Dictionary (extracted from alignment data): [dict.en-de]
Automatically describing the content of an image is a fundamental problem in artificial intelligence that connects computer vision and natural language processing. A description must capture not only the objects contained in an image, but it also must express how these objects relate to each other as well as their attributes and the activities they are involved in. Need a language model to express in a natural language.
Algorithm

- **Neural Image Caption**
  - an end-to-end neural network system that can automatically view an image and generate a reasonable description in plain English
  - extracted the representation by a deep convolution neural network (CNN) from a image
  - generated captions by a recurrent neural network (RNN) taking image representation as extra input

![Diagram showing the process of neural image captioning](image)
NIC Structure

- Left: inception net, a series of conv-batchNorm-relu-pooling layers
- Right: LSTM layer
Dataset

- MS COCO dataset
  - To advance the state-of-the-art in object recognition by placing the question of object recognition in the context of the broader question of scene understanding
  - 82783 training samples, 40504 validation samples, 40775 test samples

#5---Image-to-Image

- Image-to-image translation is a class of vision and graphics problems where the goal is to learn the mapping between an input image and an output image using a training set of aligned image pairs.
- Image to image translation, converting an image from one representation of a given scene.
Algorithm

- cycleGAN
  - learning to translate an image from a source domain X to a target domain Y in the absence of paired examples
  - translate between domains without paired input-output examples
    - powerful translation systems in the supervised setting, where example image pairs are available
    - obtaining paired training data can be difficult and expensive
  - Jun-Yan Zhu et, al. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks
cycleGAN Structure

- **Generator**
  - contains two stride-2 convolutions, several residual blocks, and two fractionally-strided convolutions

- **Discriminator**
  - a series of Conv2d, BatchNorm2d, and LeakyReLU layers, and outputs the final probability through a Sigmoid activation function
Dataset

- Cityscapes dataset

  - focuses on semantic understanding of urban street scenes
  - a large-scale dataset that contains a diverse set of stereo video sequences recorded in street scenes from 50 different cities, with high quality pixel-level annotations of 5,000 frames in addition to a larger set of 20,000 weakly annotated frames
Cityscapes dataset

- **30 classes**

<table>
<thead>
<tr>
<th>Group</th>
<th>Classes</th>
</tr>
</thead>
<tbody>
<tr>
<td>flat</td>
<td>road · sidewalk · parking+ · rail track+</td>
</tr>
<tr>
<td>human</td>
<td>person* · rider*</td>
</tr>
<tr>
<td>vehicle</td>
<td>car* · truck* · bus* · on rails* · motorcycle* · bicycle* · caravan+ · trailer+</td>
</tr>
<tr>
<td>construction</td>
<td>building · wall · fence · guard rail+ · bridge+ · tunnel+</td>
</tr>
<tr>
<td>object</td>
<td>pole · pole group+ · traffic sign · traffic light</td>
</tr>
<tr>
<td>nature</td>
<td>vegetation · terrain</td>
</tr>
<tr>
<td>sky</td>
<td>sky</td>
</tr>
<tr>
<td>void</td>
<td>ground+ · dynamic+ · static+</td>
</tr>
</tbody>
</table>
#6---Speech-to-Text (Speech recognition)

- enables the recognition and translation of spoken language into text by computers
- has a long history with several waves of major innovations. Most recently, the field has benefited from advances in deep learning and big data.
Algorithm

- Deep speech
  - end-to-end deep RNN
    - with one or more convolutional input layers, followed by multiple recurrent (uni or bidirectional) layers and one fully connected layer before a softmax layer
      - using the CTC loss function

Dataset

- Librispeech
  - a corpus of read English speech, suitable for training and evaluating speech recognition systems
  - derived from audiobooks that are part of the LibriVox project, and contains 1000 hours of speech sampled at 16 kHz

Librispeech

- Large-scale (1000 hours) corpus of read English speech

- dev-clean.tar.gz [337M] (development set, "clean" speech) Mirrors: [China]
- dev-other.tar.gz [314M] (development set, "other", more challenging, speech) Mirrors: [China]
- test-clean.tar.gz [346M] (test set, "clean" speech) Mirrors: [China]
- test-other.tar.gz [328M] (test set, "other" speech) Mirrors: [China]
- train-clean-100.tar.gz [6.3G] (training set of 100 hours "clean" speech) Mirrors: [China]
- train-clean-360.tar.gz [23G] (training set of 360 hours "clean" speech) Mirrors: [China]
- train-other-500.tar.gz [30G] (training set of 500 hours "other" speech) Mirrors: [China]
- intro-disclaimers.tar.gz [695M] (extracted LibriVox announcements for some of the speakers) Mirrors: [China]
- original-mp3.tar.gz [87G] (LibriVox mp3 files, from which corpus' audio was extracted) Mirrors: [China]
- original-books.tar.gz [297M] (Project Gutenberg texts, against which the audio in the corpus was aligned) Mirrors: [China]
- raw-metadata.tar.gz [33M] (Some extra meta-data produced during the creation of the corpus) Mirrors: [China]
- md5sum.txt [600 bytes] (MD5 checksums for the archive files) Mirrors: [China]
Face embedding is the transformation of a face image into a vector in embedding space. With this embedding vector, various tasks can be implemented, such as:

- face verification simply involves thresholding the distance between the two embeddings;
- recognition becomes a k-NN classification problem;
- clustering can be achieved using off-the-shelf techniques such as k-means or agglomerative clustering.
Algorithm

- **FaceNet**
  - consists of a batch input layer and a deep CNN followed by L2 normalization
  - Triplet Loss minimizes the distance between an anchor and a positive, both of which have the same identity, and maximizes the distance between the anchor and a negative of a different identity
FaceNet Structure

- learns a mapping from face images to a compact Euclidean space where distances directly correspond to a measure of face similarity

- uses a deep convolutional network trained to directly optimize the embedding itself, rather than an intermediate bottleneck layer as in previous deep learning approaches
Dataset

- Labeled Faces in the Wild (LFW)
  - a database of face photographs designed for studying the problem of unconstrained face recognition
  - the de-facto academic test set for face verification
  - contains more than 13,000 images of faces collected from the web
    - Each face has been labeled with the name of the person pictured. 1680 of the people pictured have two or more distinct photos in the data set.

FLW Dataset

- Four different sets
  - the original and three different types of "aligned" images
    - funneled images
    - LFW-a
    - deep funneled
Object detection is the problem of finding and classifying a variable number of objects on an image.

- widely used in computer vision task such as face detection, face recognition, video object co-segmentation.
- also used in tracking objects, e.g. tracking a ball during a football match, tracking movement of a cricket bat, tracking a person in a video.
Algorithm

- Faster R-CNN
  - shares full-image convolutional features with the detection network, thus enabling nearly cost-free region proposals
  - a fully convolutional network that simultaneously predicts object bounds and objectness scores at each position
  - State-of-the-art object detection networks depend on region proposal algorithms to hypothesize object locations.
  - Advances like SPPnet and Fast R-CNN have reduced the running time of these detection networks, exposing region proposal computation as a bottleneck

- Shaoqing Ren et al. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks
Faster R-CNN Structure

- Conv layers could use vgg net

<table>
<thead>
<tr>
<th>ConvNet Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
</tr>
<tr>
<td>11 weight layers</td>
</tr>
<tr>
<td>conv3-64</td>
</tr>
<tr>
<td>maxpool</td>
</tr>
<tr>
<td>maxpool</td>
</tr>
<tr>
<td>maxpool</td>
</tr>
<tr>
<td>maxpool</td>
</tr>
</tbody>
</table>
Dataset

- MS COCO dataset
  - 82783 training samples, 40504 validation samples, 40775 test samples
Recommender systems are utilized in a variety of areas including movies, music, news, books, research articles, search queries, social tags, and products in genera.

Collaborative filtering build a model from a user's past behaviour (items previously purchased or selected and/or numerical ratings given to those items) as well as similar decisions made by other users.
Algorithm

- Matrix factorization
  - A typical method of collaborative filtering

\[
\begin{bmatrix}
r_{11} & r_{12} & \cdots & r_{1n} \\
r_{21} & r_{22} & \cdots & r_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
r_{m1} & r_{m2} & \cdots & r_{mn}
\end{bmatrix}
\times
\begin{bmatrix}
u_{11} & u_{12} & \cdots & u_{1k} \\
u_{21} & u_{22} & \cdots & u_{2k} \\
\vdots & \vdots & \ddots & \vdots \\
u_{m1} & u_{m2} & \cdots & u_{mk}
\end{bmatrix}
\times
\begin{bmatrix}
v_{11} & v_{12} & \cdots & v_{1n} \\
v_{21} & v_{22} & \cdots & v_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
v_{k1} & v_{k2} & \cdots & v_{kn}
\end{bmatrix}^T
\]
Dataset

- **MovieLens**
  - describe people’s expressed preferences for movies.
  - 20 million ratings and 465,000 tag applications applied to 27,000 movies by 138,000 users.
PageRank (PR) is an algorithm used by search engine to rank web pages in their search engine results. PageRank is a way of measuring the importance of website pages.

Dataset

Google Web Graph

- This data set is unstructured, containing 875713 nodes representing web pages and 5105039 edges representing the links between web pages.
Graphical Model

- Use a graph expresses the conditional dependence structure between random variables.
- They are commonly used in probability theory, statistics—particularly Bayesian statistics—and machine learning.
Algorithm

- Latent dirichlet allocation (LDA)
  - a generative statistical model that allows sets of observations to be explained by unobserved groups that explain why some parts of the data are similar

1. Choose $N \sim \text{Poisson}(\xi)$.
2. Choose $\theta \sim \text{Dir}(\alpha)$.
3. For each of the $N$ words $w_n$:
   - (a) Choose a topic $z_n \sim \text{Multinomial}(\theta)$.
   - (b) Choose a word $w_n$ from $p(w_n | z_n, \beta)$, a multinomial probability conditioned on the topic $z_n$. 
Dataset

- Wikipedia Entries
  - The Wikipedia data set is un-structured, consisting of 4,300,000 English articles.
  - https://dumps.wikimedia.org/
#12---Clustering

- The task of grouping a set of objects in such a way that objects in the same group (called a cluster) are more similar (in some sense) to each other than to those in other groups (clusters).
  - It is a main task of exploratory data mining, and a common technique for statistical data analysis
  - Used in many fields, including machine learning, pattern recognition, image analysis, information retrieval, bioinformatics, data compression, and computer graphics.
Algorithm

- **K-means**
  - to partition \( n \) observations into \( k \) clusters in which each observation belongs to the cluster with the nearest mean
  - involved motif: Matrix, Sort

1. randomly generating \( k \) initial means are
2. Creating \( k \) clusters by associating every observation with the nearest mean
3. Computing the new mean of each clusters
4. Repeating steps 2 and 3
Dataset

- Facebook Social Graph
  - Undirected graph. This data set contains 4039 nodes, which represent users, and 88234 edges, which represent friendship between users.
  - http://snap.stanford.edu/data/egonets-Facebook.html
Text classification is the process of assigning tags or categories to text according to its content.

It’s one of the fundamental tasks in Natural Language Processing (NLP) with broad applications such as sentiment analysis, topic labeling, spam detection, and intent detection.
Algorithm

- Naive Bayes

- probabilistic classifiers based on applying Bayes' theorem with strong independence assumptions between the features


\[
\hat{y} = \arg\max_{k \in \{1, \ldots, K\}} p(C_k) \prod_{i=1}^{n} p(x_i | C_k).
\]
Dataset

- **20NEWS GROUP**
  - training set contains 11269 text documents, and the test set contains 7505 text documents. Each document is classified into one topic out of 20.
#14---Feature Exaction

- feature extraction starts from an initial set of measured data and builds derived values (features) intended to be informative and non-redundant, facilitating the subsequent learning and generalization steps.

- Feature extraction is a dimensionality reduction process, where an initial set of raw variables is reduced to more manageable groups (features) for processing, while still accurately and completely describing the original data set.
Algorithm

- Scale-invariant feature transform (SIFT)
  - a feature detection algorithm in computer vision to detect and describe local features in images
  - involved motif: Matrix, Transform, Sampling, Sort, Statistics
Dataset - ImageNet

- ImageNet
  - 15 million labeled high-resolution images belonging to roughly 22,000 categories.
  - The images were collected from the web and labeled by human labelers.
  - Presented for the first time as a poster at CVPR 2009.

Deng, Jia; Dong, Wei; Socher, Richard; Li, Li-Jia; Li, Kai; Fei-Fei, Li (2009), "ImageNet: A Large-Scale Hierarchical Image Database", 2009 conference on Computer Vision and Pattern Recognition

#15---Search Engine Indexing

- Search engine indexing collects, parses, and stores data
  - facilitate fast and accurate information retrieval
  - optimize speed and performance in finding relevant documents for a search query
    - Without an index, need to scan every document, which is time-consuming and highly-cost
**Algorithm**

**Inverted Index**

- An index data structure storing a mapping from content, such as words or numbers, to its locations in a database file, or in a document or a set of documents.

<table>
<thead>
<tr>
<th>Words</th>
<th>Document</th>
</tr>
</thead>
<tbody>
<tr>
<td>ant</td>
<td>doc1</td>
</tr>
<tr>
<td>demo</td>
<td>doc2</td>
</tr>
<tr>
<td>world</td>
<td>doc1, doc2</td>
</tr>
</tbody>
</table>
Dataset

- Wikipedia Entries
  - The Wikipedia data set is un-structured, consisting of 4,300,000 English articles.
  - https://dumps.wikimedia.org/
QUESTIONS
And
Answers