Text Mining

SIAM Presentation
Ashok Srivastava, Ph.D.

Recurring Anomaly Detection System (ReADS)
Ashok Srivastava, Ph.D.
Dawn McIntosh
Pat Castle
Manos Pontikakis
Vesselin Diev
Brett Zane-Ulman
Eugene Turkov
Ram Akella, Ph.D.
Zuobing Xu
Sakthi Preethi Kumaresan
Advance Engineering Network Team (NASA Ames Research Center)

Ideas

- Needle in haystack problem
- Sampling data does not work (may not sample the entire needle)
- Outline
  - Problem
  - Approach
    - Supervised, unsupervised, semisupervised
    - New similarity measures
    - Kernel methods
    - PC
    - MDS with kernels

Problem Introduction

NASA programs have large numbers (and types) of problem reports.

- ISS PRACA: 3000+ records, 1-4 pages each;
- ISS SCR: 28,000+ records, 1-4 pages each;
- Shuttle CARS: 7000+ records, 1-4 pages each;
- ASRS: 27000+ records, 1 paragraph each

These free text reports are written by a number of different people, thus the emphasis and wording vary considerably

With so much data to sift through, analysts (subject experts) need help identifying any possible safety issues or concerns and to help them confirm that they haven't missed important problems.

- Unsupervised clustering is the initial step to accomplish this;
- We think we can go much farther, specifically, identify possible recurring anomalies.
  - Recurring anomalies may be indicators of larger systemic problems.

Text Mining Solution - ReADS

Recurring Anomaly Discovery System (ReADS):

- The Recurring Anomaly Detection System (ReADS) is an integrated secure online tool to analyze text reports, such as aviation reports and maintenance records.
  - Text clustering algorithms group large quantities of reports and documents.
    - Reduces human error & fatigue
  - Automates the discovery of unknown recurring anomalies;
  - Identifies interconnected reports;
  - Provides a visualization of the clusters and recurring anomalies

- We have illustrated our techniques on data from

Recurring Anomaly "Fingerprints"

- ✓ Recurrent failures
- ✓ Problems that cross traditional system boundaries so failure effects are not fully recognized
- • Evidence of unconfirmed or random failures
- ✓ Problems that have been accepted by repeated waivers
- ✓ Discrepant conditions repeatedly accepted by routine analysis
- • Problems that are the focus of alternative opinions within the engineering community

ReADS Text Mining Algorithms

Unsupervised Clustering:

Spherical k-means → modified von Mises Fisher.

Recurring Anomaly Identification:

1. Identify reports which mention other reports as a recurring anomaly;

2. Detect recurring anomalies,
   a. find the similarity between documents to detect recurring anomalies using cosine distance similarity measure,
   b. then according to the similarity measure, run the hierarchical clustering algorithm to cluster the recurring anomalies.

Similarity between Reports

Cosine Similarity Measure

Calculate the inner product of the normalized term frequency vectors

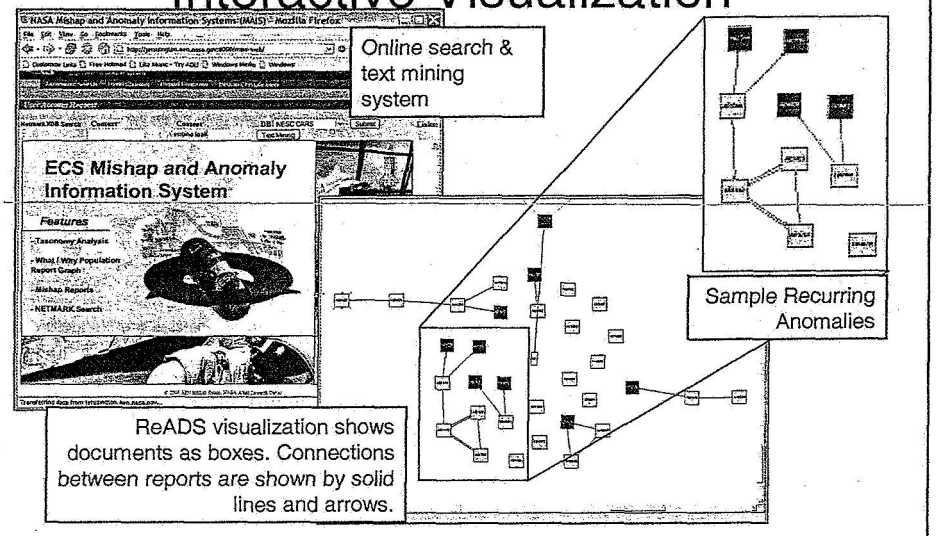
$$R\left(d_t \middle| d_i\right) = \cos d_t d_i$$

$$= \frac{\sum_{k=1}^{n} w_k\left(d_i\right) w_k\left(d_t\right)}{\left\|d_t\right\| \left\|d_i\right\|}$$

Hierarchical Clustering of Recurring Anomalies

- After calculating the distance between each document, the algorithm applies single linkage, i.e., nearest neighbor, to create a hierarchical tree representing connections between documents.
  - Also generates an 'inconsistency coefficient' which is a measure of the relative consistency of each link in the tree.
- The hierarchical tree is partitioned into clusters by setting a threshold on the inconsistency coefficient.
  - A high inconsistency coefficient implies that the reports could be very different and still be sorted into the same cluster.
- Currently the inconsistency coefficient threshold is set very low, which returns many smaller clusters of very similar reports.
  - Clusters of single documents are excluded from the recurring

ReADS System & Interactive Visualization



Intro

- In an attempt to quantify any improvements Natural Language Processing (NLP) & text normalization have on text classification using Support Vector Machines (SVM) and Naïve Bayes, we did a direct comparison of classification rates of documents that has been processed by:

  (1) documents processed using a NLP tool & a text normalization tool, PLADS, and

  (2) the same documents with no preprocessing.

  Specifically, we:
  - Measured the difference in Precision, Recall, and F-Measure
  - Applied to 60 anomaly classification.
    - Not meant to be an optimum classifier technique. Precision and Recall results for the different preprocessing methods were compared. No work was done to improve either.
- Dataset used:
  - Aviation Safety Reporting System (ASRS)
  - ASRS is classified by anomalies. These reports are classified into over 100 anomalies. Each report may be classified in multiple anomaly classes.
    - 30% are in only one anomaly class
    - 50% are in 3 anomaly classes
  - Documents are short, approximately 6 sentences
  - 27,596 documents
  - Training Dataset: 20,000 docs dedicated to training, 4000 selected
  - Test Dataset: 7,000 docs dedicated to testing, 2000 selected
- Tools used:
  - MATLAB used for preprocessing
  - Weka implemented for SVM and Naïve Bayes classification

Sample PLADS Term Reduction

JUST PRIOR TO TOUCHDOWN, LAX TWR TOLD US TO GO AROUND BECAUSE OF THE ACFT IN FRONT OF US. BOTH THE COPLT AND I, HOWEVER, UNDERSTOOD TWR TO SAY, CLRED TO LAND, ACFT ON THE RWY. SINCE THE ACFT IN FRONT OF US WAS CLR OF THE RWY AND WE BOTH MISUNDERSTOOD TWR'S RADIO CALL AND CONSIDERED IT AN ADVISORY, WE LANDED. AS WE TAXIED TO THE GATE, TWR REQUESTED THAT I CALL THEM FROM A PHONE WHEN I HAD THE OPPORTUNITY (I CALLED FROM THE GATE). IT WAS ON THE PHONE THAT I DISCOVERED TWR HAD SENT US AROUND. IN HINDSIGHT, FROM THEIR PERSPECTIVE, GOING AROUND WAS THE PRUDENT THING TO DO. I HAVE BECOME TOO CONDITIONED IN THE PAST FEW YRS IN BEING VECTORED INTO A VISUAL APCH BEHIND AN ACFT THAT IS TOO CLOSE. REGRETTABLY, IN THIS SIT, CONFUSION AND MISUNDERSTANDING PUT US IN A DIFFICULT SIT.

↓ Expand Acronyms, Simplify Punctuation ↓

JUST PRIOR TO TOUCHDOWN, LAX tower TOLD US TO GO AROUND BECAUSE OF THE aircraft IN FRONT OF US. BOTH THE copilot AND I, HOWEVER, UNDERSTOOD tower TO SAY, clear TO LAND, aircraft ON THE runway. SINCE THE aircraft IN FRONT OF US WAS clear OF THE runway AND WE BOTH misunderstand tower RADIO CALL AND CONSIDERED IT AN ADVISORY, WE LANDED. AS WE TAXIED TO THE GATE, tower REQUESTED THAT I CALL THEM FROM A PHONE WHEN I HAD THE OPPORTUNITY I CALLED FROM THE GATE. IT WAS ON THE PHONE THAT I DISCOVERED tower HAD SENT US AROUND. IN HINDSIGHT, FROM THEIR PERSPECTIVE, GOING AROUND WAS THE PRUDENT THING TO DO. I HAVE BECOME TOO CONDITIONED IN THE PAST FEW year IN BEING VECTORED INTO A VISUAL approach BEHIND AN aircraft THAT IS TOO CLOSE. REGRETTABLY, IN THIS situation, CONFUSION AND MISUNDERSTANDING PUT US IN A DIFFICULT situation.

↓ Stemming, Remove Non-Informative Terms, Phrasing ↓

PRIOR _ TOUCHDOWN _ tower TOLD _ _ goaround _ _ _ aircraft _ FRONT _ _ _ _ copilot _ _ _ understand tower _ SAY clear _ LAND aircraft _ _ runway _ _ aircraft _ FRONT _ _ _ clear _ _ runway _ _ _ misunderstand tower RADIO CALL _ consider _ _ advise _ lan _ _ taxiedto _ GATE tower request _ _ CALL _ _ _ PHONE _ _ _ _ OPPORTUNITY _ call _ _ GATE _ _ _ _ PHONE _ _ discover tower _ SENT _ _ _ HINDSIGHT _ _ PERSPECTIVE go _ _ _ prudentthing _ _ _ _ _ _ condition _ _ PAST _ year _ _ vector _ _ VISUAL approach _ _ aircraft _ _ _ CLOSE REGRETTABLY _ _ situate confuse _ misunderstand PUT _ _ _ difficultsituation

Raw Text & PLADS Comparison

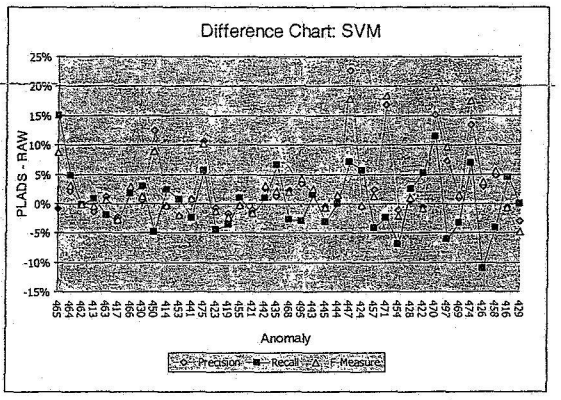
In order to classify the documents, they are first formatted into a document-term frequency matrix. The cells of the matrix are the frequency count of the terms that appear in the document.

	Term 1	Term 2	Term 3	Term 4
Document 1	0	1	0	4
Document 2	0	3	0	0
Document 3	2	8	1	0

- PLADS reduced the total number of terms in 27000 documents from 44940 to 31701
- PLADS reduced classification computation time by 0%-10%

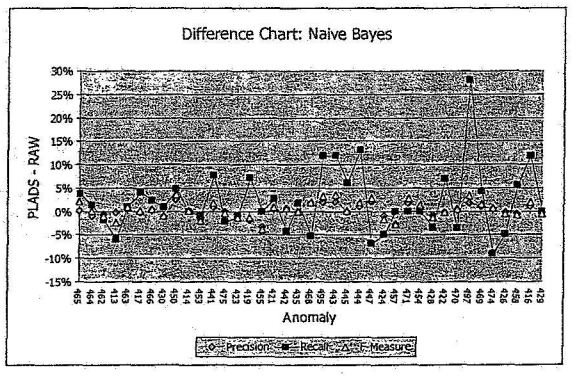
Comparison of Raw Text vs. PLADS using SVM

- All terms used, no additional term reduction applied
- PLADS improves precision 2% on average
- PLADS improves recall 2% on average



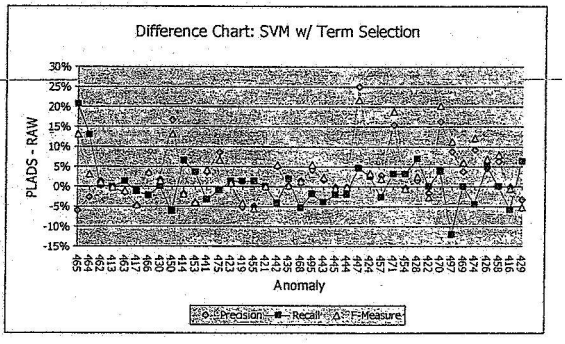
Comparison of Raw Text vs. PLADS using Naïve Bayes

- All terms used, no additional term reduction applied
- PLADS improves Naïve Bayes precision 1% on average
- PLADS improves Naïve Bayes recall 2% on average



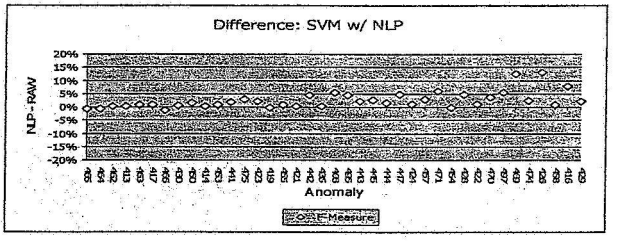
Comparison of Raw Text vs. PLADS with Terms Selection

- 1000 terms selected using Information Gain
- PLADS improves precision 2% on average
- PLADS improves recall 3% on average



Comparison of Raw Text vs. NLP with Terms Selection

- 500 terms selected using Information Gain
- NLP improves F-measure 3% on average



Text Mining

Sakthi Preethi Kumaresan
Graduate Student, UCSC (shakthi@soe.ucsc.edu)
Prof. Ramakrishna Akella(UCSC) in collaboration with
Dr. Ashok Srivastava (ARC, NASA)

Text Categorization - Applications

- Automated sorting of scientific articles according to predefined thesauri of technical words.
- Filing patents into patent directories
- Selective dissemination of information to consumers
- Automated population of hierarchical catalogues of web resources
- Spam filtering
- Identification of document genre
- Authorship attribution
- Automated Essay Grading

Applications – contd.

- Detection of Recurring Anomalies:
  - Complex Systems have significant amount of maintenance and problem databases.
  - Clustering helps detect recurring anomalies and relations in problem reports that indicate larger systemic problems.

Tool Kit

- Involves the synergy of the Information Retrieval (IR) Technology and Machine Learning (ML) Technology
  - Support Vector Machines
  - Neural Networks
  - Boosting Algorithms
- Latent Semantic Analysis:

  Natural Language Processing (NLP) can be used to integrate morphological, syntactic and semantic analysis with the process of clustering documents,

TF-IDF Classifier

- Uses the Bag of Words representation.
- Term Document Matrix:
  - Each Document is represented as a row.
  - The columns represent the union of all words in all documents
- Cells of the matrix are TFIDF(Term Frequency Inverse Document Frequency) for the corresponding word and document.

$$IDF(w) = log\left(\frac{|D|}{DF(w)}\right)$$

$$d^{(i)} = TF(w_i, d) \cdot IDF(w_i)$$

- *Term Frequency (TF)*: Frequency of occurrence of word w_i in document d_j
- $|D|$: Total number of documents
- $DF(w)$: Total number of documents containing word w_j

TFIDF Classifyer

- Prototype vectors are generated for each class:

$$\vec{c} = \sum_{d \in C} \vec{d}$$

- Decision is taken by measuring the cosine of the angle between the prototype vector and the data vector.

$$H_{TFIDF}(d') = \underset{C \in C}{\operatorname{argmax}} cos(\vec{d'}, \vec{c})$$

Naive Bayes Classifier

- Underlying Assumptions:

  1. We have |C| probability distributions.

  2. Each document is generated from the p.d.f. associated with that particular class. The 'i'th word of the document is generated from the 'i' th independent trial.

$$H_{BAYES}(d') = \arg\max_{C \in \mathcal{C}} \Pr(C|d')$$

This is calculated by Bayes rule:

$$\Pr(C|d') = \frac{\Pr(d'|C) \cdot \Pr(C)}{\sum_{C \in \mathcal{C}} \Pr(d'|C) \cdot \Pr(C)}$$

Because of the assumption, Pr(d'/c) can be written as:

$$\Pr(d'|C) = \prod_{i=1}^{|d'|} \Pr(w_i|C)$$

Pr TFIDF

- Assumptions:
  - Each document has a representation 'x'.
  - These representations are not unique. A function θ maps a document to its representation.

$$H_{PrTFIDF}(d') = \arg\max_{C \in \mathcal{C}} \Pr(C|d', \Theta)$$

$$\Pr(C|d', \Theta) = \sum_{x} \Pr(C|x) \cdot \Pr(x|d', \Theta)$$

- Function θ is design choice. Lets say $x = w$ and $Pr(x/d',\theta)=Pr(w/d',\theta)$

  So documents are represented by single words. They do not have one fixed representation.

$$\Pr(x|d', \Theta) = \Pr(w|d', \Theta) = \frac{TF(w, d')}{\sum_{w' \in F} TF(w', d')}$$ and $$\Pr(C|w) = \frac{\Pr(w|C) \cdot \Pr(C)}{\sum_{C' \in \mathcal{C}} \Pr(w|C') \cdot \Pr(C')}$$

where, Pr(C) is the prior probability and $$\Pr(w|C) = \frac{TF(w, C)}{\sum_{w' \in F} TF(w', C)}$$

- Resulting Rule:

$$H_{PrTFIDF}(d') = \arg\max_{C \in \mathcal{C}} \sum_{w \in F} \frac{\Pr(w|C) \cdot \Pr(C)}{\sum_{C' \in \mathcal{C}} \Pr(w|C') \cdot \Pr(C')} \cdot \Pr(w|d', \Theta)$$

Relating the PrTFIDF to TFIDF

- Assumptions to show the equivalence of PrTFIDF and TFIDF:
  - Equal Prior probabilities.
  - There is a λ such that: $\lambda \cdot \sum_{w' \in F} TF(w', C) = \sqrt{\sum_{w' \in F} (TF(w', C) \cdot IDF(w'))^2}$
  - Define IDF as: $IDF'(w) = sqrt\left(\frac{|D|}{DF'(w)}\right)$ $DF'(w) = \sum_{C \in C} \frac{TF(w, C)}{\sum_{w' \in F} TF(w', C)}$
- Under these assumptions it can be shown that:

$$H_{PrTFIDF}(d') = H_{TFIDF}(d')$$

- Thus this explains the statistical framework behind the TFIDF vector approach for data classification.

Clustering Of Directional Data Two Broad Kinds Of Algorithms

1. Generative (parametric)

Examples : Mixture of Gaussians;
Mixture of VMF distributions
Model using the exponential family

2. Discriminative (non - parametric)

K-means - measures the Euclidean distance

spK-means - measures cosine similarity

fsK-means - frequency sensitive

More On Generative Models

- Effect is analogous to the use of Euclidean Distances from the discriminative perspective.
- Often involve an appropriate use of the Expectation Maximization Algorithm.

Brief look at EM and K- means

- We have M data points S that we want to fit using a mixture of K univariate Gaussian distributions with identical and known variance.
- Problem: We don't know which data point was generated using which of the distributions. Represent data points as $\langle Y_m, w_{m1}, w_{m2}, \ldots w_{mK} \rangle$

where $w_{mk} = 1$, if Y_m was generated using distribution k, otherwise 0.

The ML solution is given by: $\mu_k = \frac{1}{M_k} \sum_{m=1}^{M} w_{mk} Y_m$

where $\sum_{m=1}^{M} w_{mk}$ and $k = 1,..K$.

But w_{mk} are not known. So we know neither w_{mk} nor the μ_k. The idea of EM is to estimate both simultaneously.

K-means in the EM Frame work

- Expectation (E) :
  Calculate the expected value of the W_{mk} based on assumed values or current estimates of μ_k

$$E[w_{mk}] = p(k \mid Y_m) = \frac{p(Y_m \mid k)P(k)}{p(Y_m)} = \frac{p(Y_m \mid \mu_k)P(k)}{\sum_{j=1}^{K} p(Y_m \mid \mu_j)P(j)} = \frac{e^{-\frac{(Y_m-\mu_k)^2}{2\sigma^2}}}{\sum_{j=1}^{K} e^{-\frac{(Y_m-\mu_j)^2}{2\sigma^2}}}$$

- This corresponds to clustering data points by minimizing the Euclidean distances in the k-means algorithm.

- Maximization (M):
  Using the Expected values of W_{mk} the ML estimates of μ_k are calculated. This corresponds to updating the k-means at every iteration of the k-means algorithm.

- The EM algorithm is also used on a mixture of VMF distributions.

- VMF :

  - Introduced by von Mises to study the deviations of measure atomic weights from integral values.

  - Its importance in statistical inference on a circle is almost the same as that of the normal distribution on a line.

Von Mises Fisher Distribution

- A circular random variable 'θ' is said to follow a von Mises Distribution if its p.d.f. is given by:

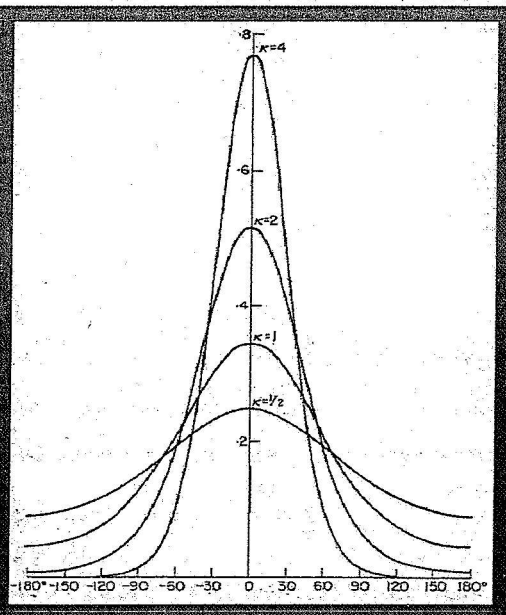
$$g(\theta;\mu_o,\kappa)=\frac{I}{2\pi I_0(\kappa)}e^{\kappa\cos(\theta-\mu_o)},0\le\theta\le 2\pi,\kappa>0,0\le\mu_o\le 2\pi,$$

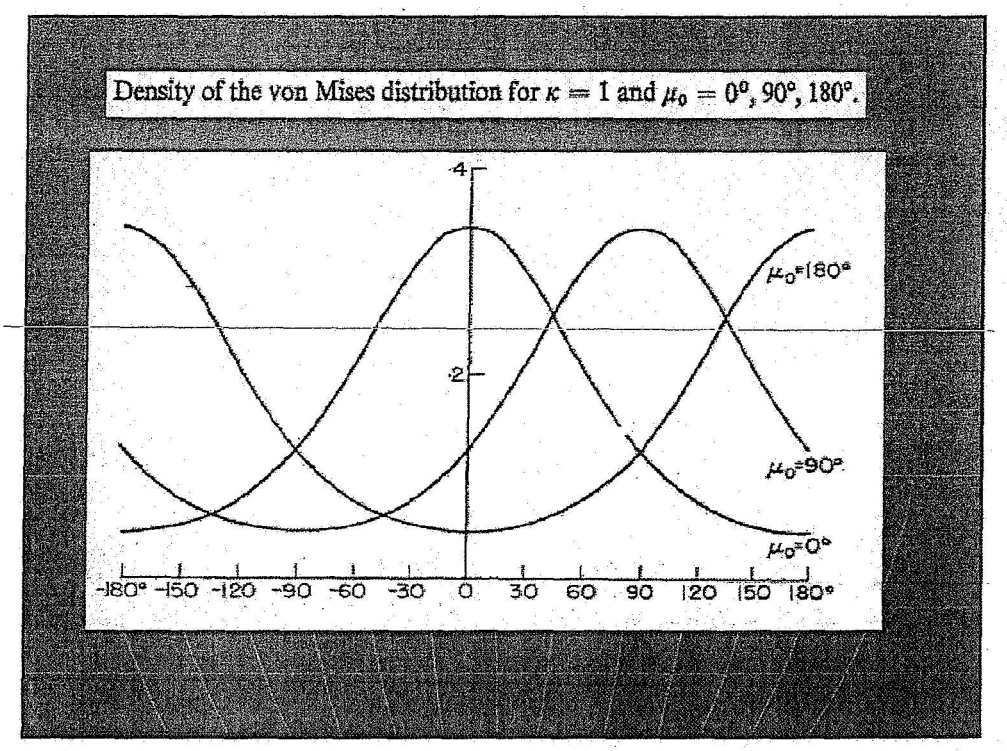
where, $I_0(\kappa)$ is the modified bessel function of the first kind and order 0.

- The parameter μ_0 is the mean direction while the parameter K is described as the concentration parameter.
- A '*d*' dimensional unit random vector, x with ||x||=1 is said to have d-variate VMF distribution if its p.d.f. is given by, $f(\mathbf{x}|\mu,\kappa)=c_d(\kappa)e^{\kappa\mu^T\mathbf{x}}$,

where $||\boldsymbol{\mu}|| = 1$ and K >= 0 and $c_d(\kappa)=\frac{\kappa^{d/2-1}}{(2\pi)^{d/2}I_{d/2-1}(\kappa)}$

Density of the von Mises distribution for $\mu_0 = 0°$ and $\kappa = \frac{1}{2}, 1, 2, 4$.





Why is the text data Directional ?

- Preprocessing step before applying the algorithms to text data : The (tf-idf) document vectors are L_2 normalized to make them unit norm.
- Assumption : Direction of documents is sufficient to get good clusters.
- For Eg: Two documents - one small, one lengthy - on the same topic will have the same direction and hence put in the same cluster.
- This unit normalized data lives on a sphere in a $R^{(d-1)}$ dimensional space.

Why is VMF an appropriate Model for Directional data ?

Analogies to the Normal Distribution

1. For large K the random variable 'θ' is distributed as $N(\mu_o, 1/K^{1/2})$
   - (proof in handout1).

2. Relation to Bivariate Normal Distribution:
   Let x and y be independent normal variables with means (cos μ_o , *sin* μ_o) and equal variances 1/ K. The p.d.f. of the polar variables (r, θ) is :

$$const \quad .r \exp\left[-\frac{\kappa}{2}\{ r^2 - 2r\cos(\theta - \mu_o)\}\right]$$

The conditional distribution of θ for r = 1, is the VMF(μ_o, K).

These clearly indicate that μ_o behaves like the mean while 1/ K influences the distribution in the same way as σ^2 influences normal Distribution.

Analogies to Normal contd.

3. The Maximum Likelihood Characterization:

For a p.d.f. on the line $f(x - \mu)$, The maximum likelihood estimate for the mean is given by the sample mean if only if the p.d.f. is Gaussian.

Likewise, for a p.d.f. on the circle $g(\theta - \mu_o)$ *the ML estimate of the mean is the sample mean* $\overline{x_o}$ *if only if g is a VMF distribution.*

Proof:

According to Log Likelihood Function if, the ML estimate of μ_o *is the sample mean direction,* $\overline{x_o}$ *then:* $\sum_{i=1}^{n} g'(\theta_i - \overline{x_0}) / g(\theta_i - \overline{x_0}) = 0$

By definition of $\overline{x_o}$,

$$\sum_{i=1}^{n} \sin(\theta_i - \overline{x_0}) = 0.$$

Since the above two equations are identical for each n, we have

$$g'(\theta_i - \overline{x_0}) / g(\theta_i - \overline{x_0}) = const.\sin(\theta_i - \overline{x_0}).$$

Replacing $\overline{x_0}$ by μ_o, $g(\theta)$ is the VMF pdf.

Analogies to Normal contd.

- Maximum Entropy Characterization:
  Given a fixed mean and variance the Gaussian is the distribution that maximizes the entropy.
- Likewise given a fixed circular variance ρ and mean direction μ_o, the VMF distribution maximizes the entropy.
- Proof:
  - Given in Handout

Analogy to the Normal Distribution- contd.

- Is there a Central limit theorem for Directional data ?

  For data on a line, the CLT says that the Normal is the limiting distribution.

Whereas for directional data, the limiting distribution of the sum of 'n' independent random variables is given by the Uniform Distribution.

"In spite of this, the Uniform Distribution is hardly a contender for modelling directional data" – [4]

What is the Appropriate Distribution to model Directional Date

- Unfortunately there is no distribution for directional data which has all properties analogous to the linear normal distribution. The VMF has some but not all of these desirable properties.
- The wrapped normal distribution is a strong contender to VMF.
- But the VMF provides:

  - simpler ML estimates.
  - tractable distribution in hypothesis testing.

VMF implemented Using the EM framework

- **Frame Work** :
- The probability density of the movMF generative model is given by:

$$f(\mathrm{x}|\Theta) = \sum_{h=1}^{k} \alpha_h f_h(\mathrm{x}|\theta_h),$$

where $\Theta = \{\alpha_1, \cdots, \alpha_k, \theta_1, \cdots, \theta_k\}$ with $\alpha_h \geq 0$, $\sum_{h=1}^{k} \alpha_h = 1$

And $f_h(\mathrm{x}|\theta_h)$ is a single VMF distribution with parameters, $\theta_h = (\mu_h, \kappa_h)$.

Frame Work – contd.

- Let $\mathcal{X} = \{x_1, x_2, x_3 \ldots x_n\}$ be generated by sampling independently from this generative model.
- Let $\mathcal{Z} = \{z_1, \ldots z_n\}$ be the corresponding set of so called hidden variables.
- $Z_i = h$ if x_i was generated following $f_h\ (x\ /\ \theta_h)$
- With the knowledge of the hidden variables, the log-likelihood is given by,

$$\ln P(\mathcal{X}, \mathcal{Z}|\Theta) = \sum_{i=1}^{n} \ln\left(\alpha_{z_i} f_{z_i}(\mathrm{x}_i|\theta_{z_i})\right),$$

Hidden variables are unknown. The above Eq is a random variable dependent on the distribution of Z. This is the complete log likelihood function.

The Maximization Step- Parameter Estimation step

- Assume that $p(h|\mathrm{x}_i, \Theta), \forall h, i,$ of the hidden variables is known.

The Expectation of the complete data log-likelihood is given by:

$$E_p[\ln P(\mathcal{X}, \mathcal{Z}|\Theta)] = \sum_{h=1}^{k}\sum_{i=1}^{n}(\ln \alpha_h)\ p(h|\mathrm{x}_i, \Theta) + \sum_{h=1}^{k}\sum_{i=1}^{n}(\ln f_h(\mathrm{x}_i|\theta_h))\ p(h|\mathrm{x}_i, \Theta).$$

Now Θ is re-estimated.

The expression for α_h is found by the method of Lagrangian multipliers with $\sum_{h=1}^{k}\alpha_h = 1.$

Again the θ_h are estimated with the condition:

$$\mu_h^T \mu_h = 1, \kappa_h \geq 0, \forall h.$$

The Expectation Step- Distribution Estimation

Given (X,θ) we estimate the conditional distribution of Z/(X, θ).

- Soft moVMF : Distribution of the hidden variable is given by:

$$p(h|\mathrm{x}_i, \Theta) = \frac{\alpha_h\ f_h(\mathrm{x}_i|\Theta)}{\sum_{l=1}^{k}\alpha_l\ f_l(\mathrm{x}_i|\Theta)}.$$

- Hard moVMF : Distribution is given by,

$$q(h|\mathrm{x}_i, \Theta) = \begin{cases} 1, & \text{if } h = \underset{h'}{\operatorname{argmax}}\ p(h'|\mathrm{x}_i, \Theta), \\ 0, & \text{otherwise.} \end{cases}$$

Algorithm 1 `soft-movMF`

Input: Set $\mathcal{X}$ of data points on $\mathbb{S}^{d-1}$
Output: A soft clustering of $\mathcal{X}$ over a mixture of k vMF distributions
Initialize all $\alpha_h, \mu_h, \kappa_h,\ h = 1, \cdots, k$
repeat
  {The E (Expectation) step of EM}
  for $i = 1$ to n **do**
    for $h = 1$ to k **do**

$$f_h(\mathbf{x}_i|\theta_h) \leftarrow c_d(\kappa_h)e^{\kappa_h \mu_h^T \mathbf{x}_i}$$

$$p(h|\mathbf{x}_i, \Theta) \leftarrow \frac{\alpha_h f_h(\mathbf{x}_i|\theta_h)}{\sum_{l=1}^{k} \alpha_l f_l(\mathbf{x}_i|\theta_l)}$$

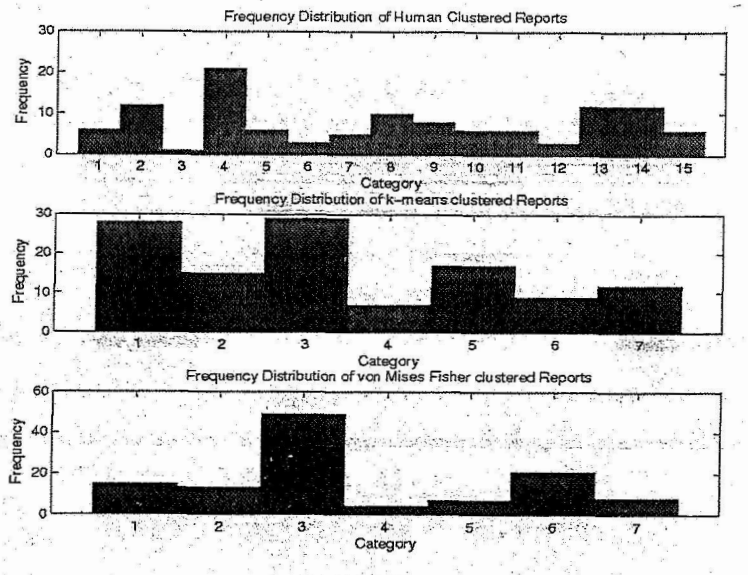
    end for
  end for
  {The M (Maximization) step of EM}
  for $h = 1$ to k **do**

$$\alpha_h \leftarrow \frac{1}{n}\sum_{i=1}^{n} p(h|\mathbf{x}_i, \Theta)$$

$$\mu_h \leftarrow \frac{\sum_{i=1}^{n} \mathbf{x}_i p(h|\mathbf{x}_i, \Theta)}{\left\|\sum_{i=1}^{n} \mathbf{x}_i p(h|\mathbf{x}_i, \Theta)\right\|}$$

$$\kappa_h \leftarrow A_d^{-1}\left(\frac{\left\|\sum_{i=1}^{n} \mathbf{x}_i p(h|\mathbf{x}_i, \Theta)\right\|}{\sum_{i=1}^{n} p(h|\mathbf{x}_i, \Theta)}\right)$$

  end for
until *convergence*



SpKmeans – reduction from soft movMF

- Assumption: Components have infinite concentration parameters.

$$\kappa_h = \kappa \to \infty, \forall h.$$

Thus a point will be assigned to cluster h* if

$$h^* = \operatorname*{argmax}_h \mathbf{x}_i^T \mu_h, \text{ since}$$

$$p(h^*|\mathbf{x}_i, \Theta) = \lim_{\kappa \to \infty} \frac{e^{\kappa\, \mathbf{x}_i^T \mu_{h^*}}}{\sum_{h=1}^{k} e^{\kappa\, \mathbf{x}_i^T \mu_h}} \to 1$$

And $p(h|\mathbf{x}_i, \Theta) \to 0, \forall h \neq h^*.$

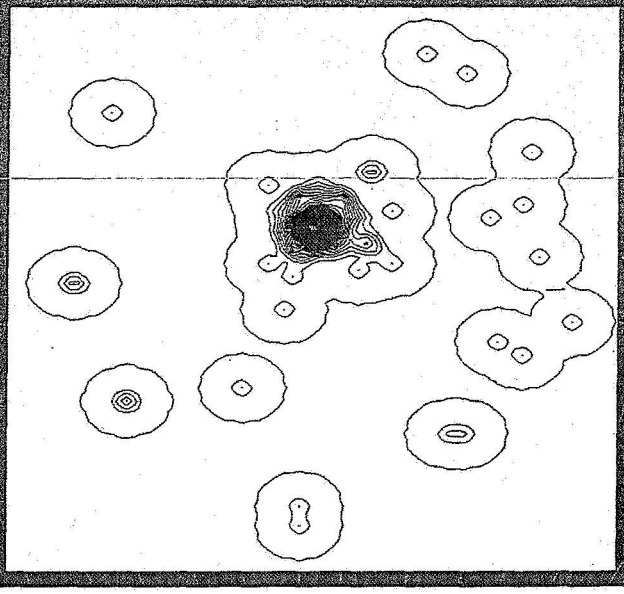
Sammon Mapping

- Project High Dimensional Data on to a two or three dimensional space.
- A set of N data points Z_i are embedded in an l dimensional space
- A new set of N data points are generated such that the following is minimized.

$$\sum_i \sum_j \| d(Z_i, Z_j) - d(Y_i, Y_j) \|^2$$

- d measures the Euclidean distance between real vectors.

A projection of 500 dimension document vectors into two dimensions using Sammon Maps[2]



Spectral Clustering

- Embed document vectors in a possibly high dimensional space using Mercer Kernels.
- Mercer kernel – Measure of similarity
  - Gaussian Kernel

$$\Phi(Z_i)\Phi^T(Z_j) = \exp(-\frac{1}{2\sigma^2}\|Z_i - Z_j\|^2)$$

  - Polynomial Kernel

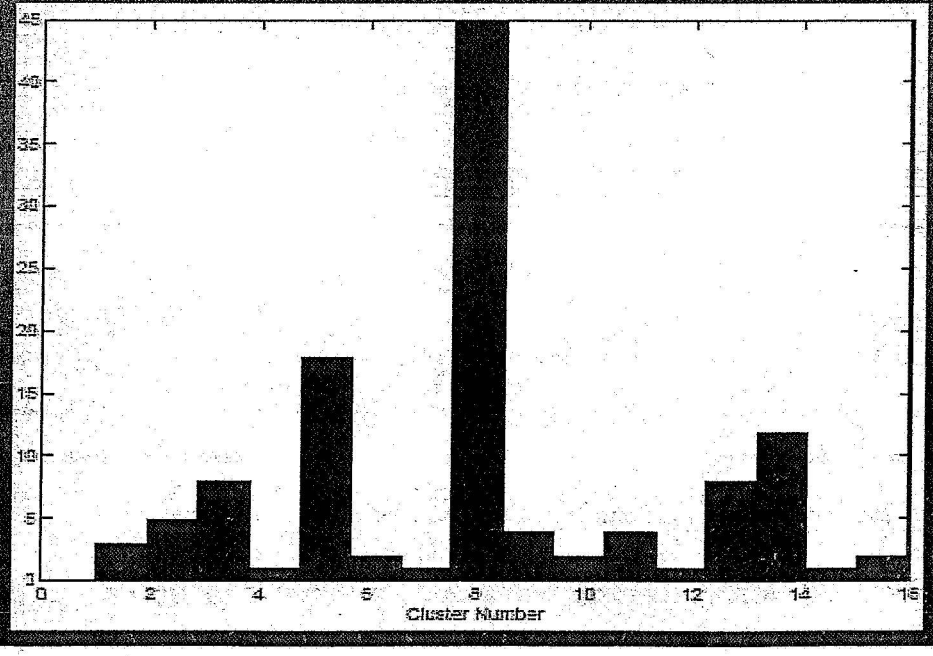
$$K(Z_i, Z_j) = \Phi(Z_i)\Phi^T(Z_j) = < Z_i, Z_j >^p$$

Spectral Clustering Contd.

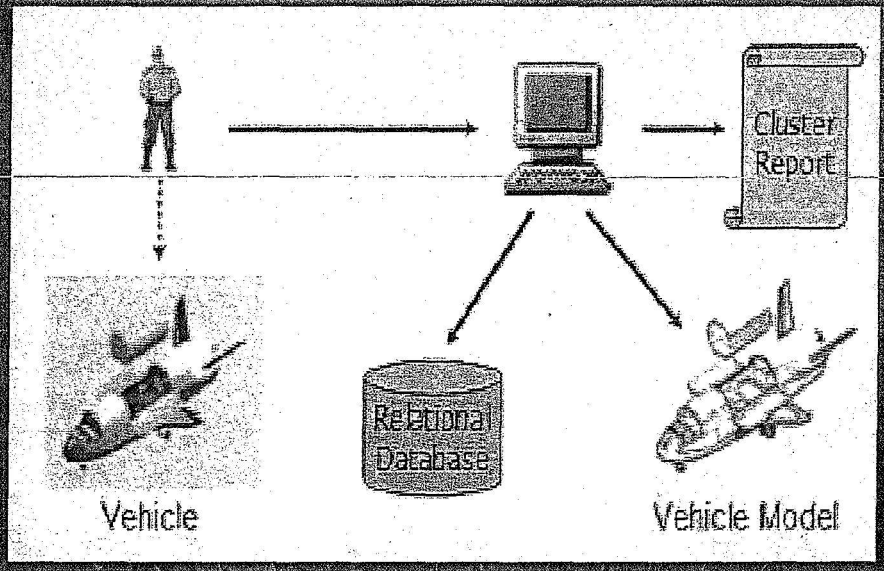
Kernel Matrix:

The *(i,j)* th entry corresponds to the similarity between documents i and j as measured by the kernel function.

Spectral Clustering - Results



System Architecture- An Example



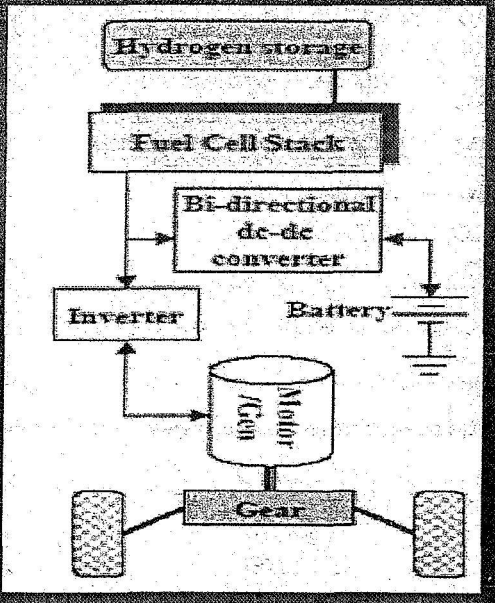
Objectives:

- A streamlined and efficient method for analyzing problem reports.
- Enhance clustering of problem reports to discover recurring anomalies

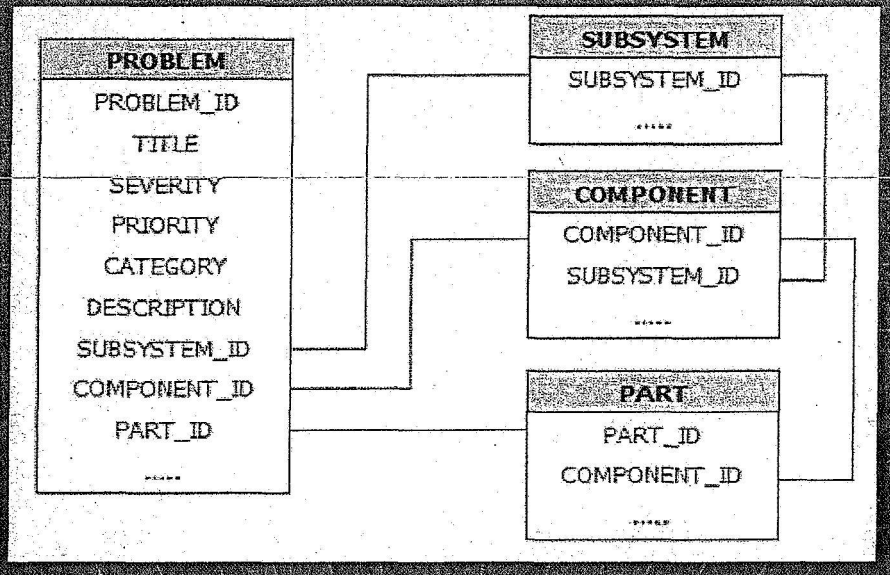
System Architecture contd.

- System Model:
  - An engineering model that defines how parts components and subsystems interact
- Relational Database:
  - Consists of tables for all the parts, subsystems and components.

System Model - Example



Relational Database Framework



References

1. Thorsten Joachims, "*A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization*", March 1996.
2. Ashok N. Srivastava and Brett Zane-Ulman, "*Discovering Recurring Anomalies in Text Reports Regarding Complex Space Systems*"
3. Arindam Banerjee, Inderjit Dhillon, Joydeep Ghosh and Suvrit Sra, "*Generative Model based clustering of Directional Data*".
4. K.V. Mardia and P. Jupp, Directional Statistics, John Wiley and Sons Ltd., 2nd Edition, 2000.
5. Justus H. Piater, "*Mixture Models and Expectation Maximization*".
6. Thomas K. Landauer, Darell Laham and Peter Foltz, "*Learning Human-like Knowledge by Singular Value Decomposition: A Progress Report*"
7. Thomas K. Landauer, Darell Laham and Peter Foltz, "*An Introduction to Latent Semantic Analysis*".
8. Fabrizio Sebastiani, "Text Categorization", Instituto di Scienza e Tecnologie dell' Informazione

THANK YOU!

Questions ?