

LUCENE / SOLR REVOLUTION / 2017

SEPTEMBER 12-15, 2017
LAS VEGAS, NV



An Intelligent, Personalized Information Retrieval Environment

Pengchu Zhang, John Herzer

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Sandia National Laboratories

Agenda

- Motivations
- Overview of Information Retrieval Models
- Unsupervised Learning for Clustering
- Supervised Learning for Classification
- Graphic Model for Predicting
- Integrating our models with Fusion
- Questions





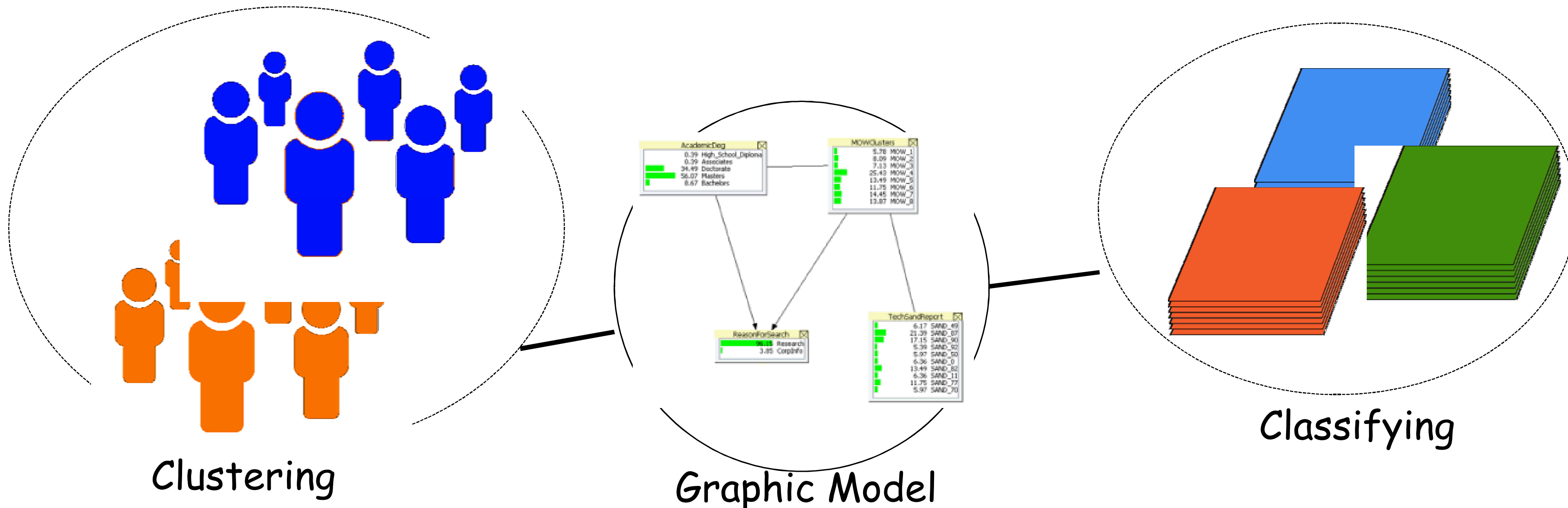
Motivations:

- ◆ Deliver Information to Employees based on their specific needs:
 - Current search engines retrieve information with built-in algorithms solely based on the query;
 - In an organization, members of the work force have different needs for information even though they use the same/similar queries, e.g.:
 - *Managers/Leaders want to know the progress of on-going projects*
 - *Developers search for current/historical technologies*
 - *Financial analysts want to know the health of the budget*
- ◆ Leverage our analytics work along with Fusion's advanced capabilities



Solution: SPIRE

- ◆ The **S**andia **P**ersonalized **I**nformation **R**etrieval **E**nvironment
- ◆ Matches customers with relevant content based on their personal attributes
- ◆ To accomplish this we need to cluster/classify customers, documents and build the probabilistic based predictive model



I. Profile MOWs with Personal Attributes

- Physical Attributes (easy to get but not very useful):
 - Education, years at job, job title...
 - Example: Sandia has about 1000 R&Ds with a job title of "Computer Science & Dev" but they are working in very different areas
- Documents generated by Employees (very useful):
 - Description of job assignments over the years;
 - Publications
 - Patents, awards, self-descriptions...

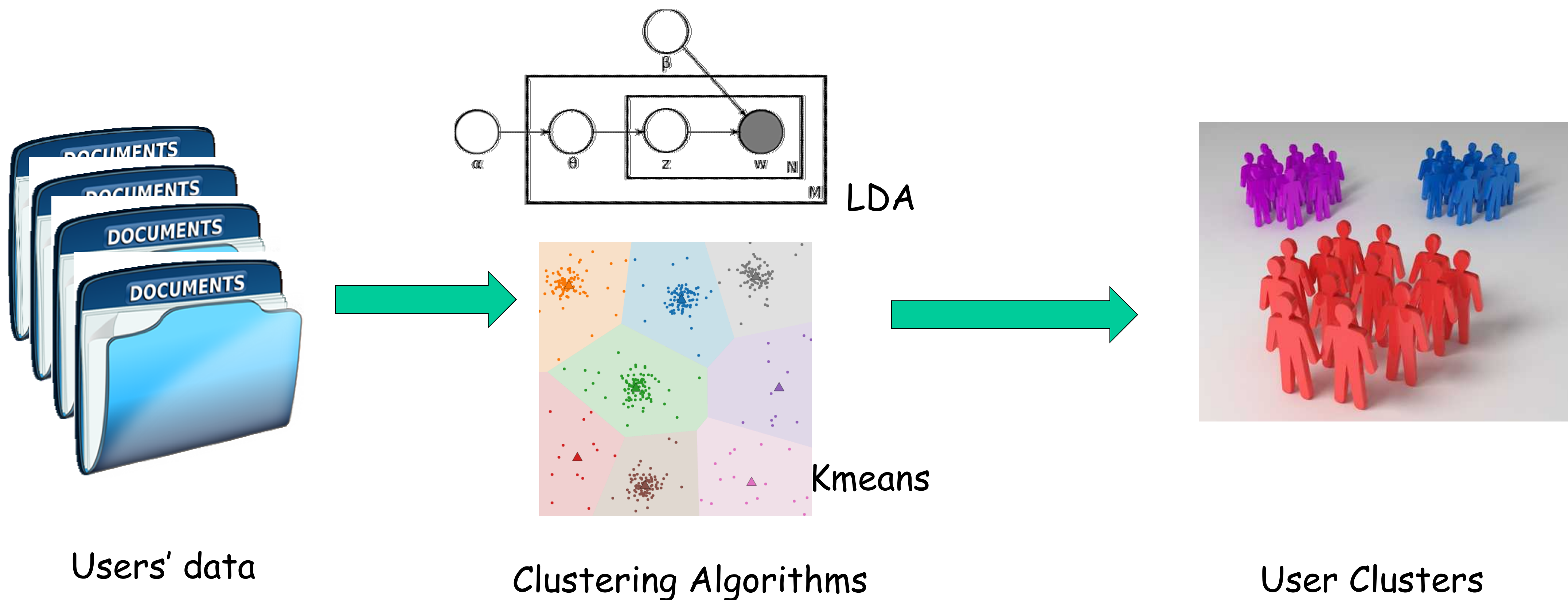


Unsupervised learning



Clustering Users Based on Their Similarities

1. Job description
2. Publications, internal documents...

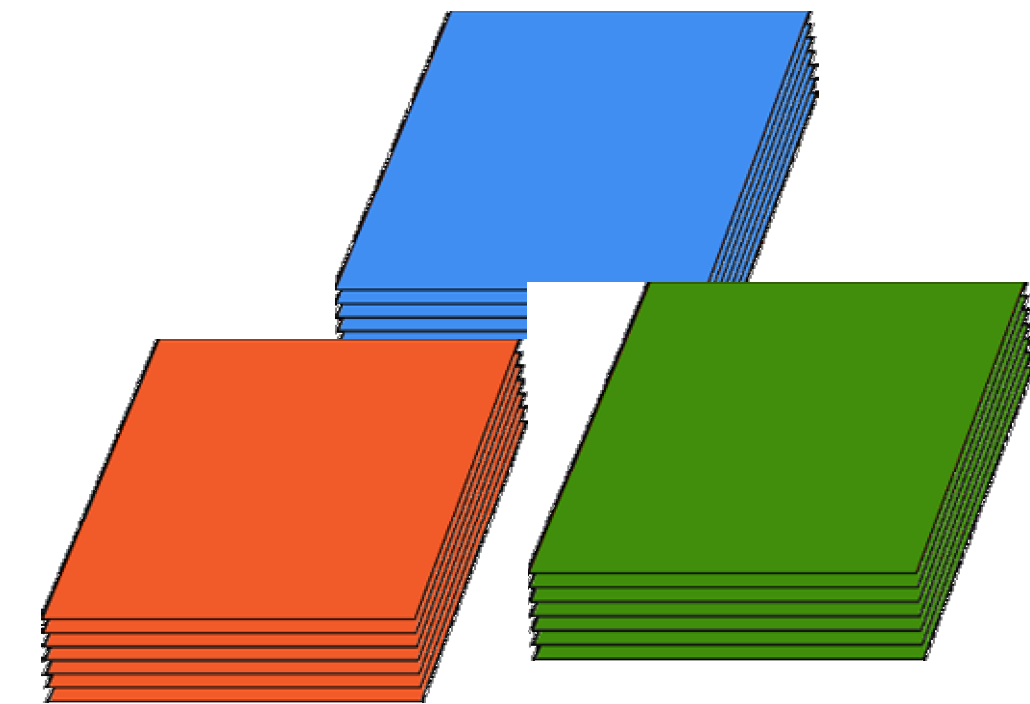


II. Document Classification

- There have been many models for classification, but:
 - Binary model such as sentiment classification: negative or positive
 - Statistical based:
 - *Need a lot of manually selected features as inputs, very expensive*
 - *Hard to scale up*

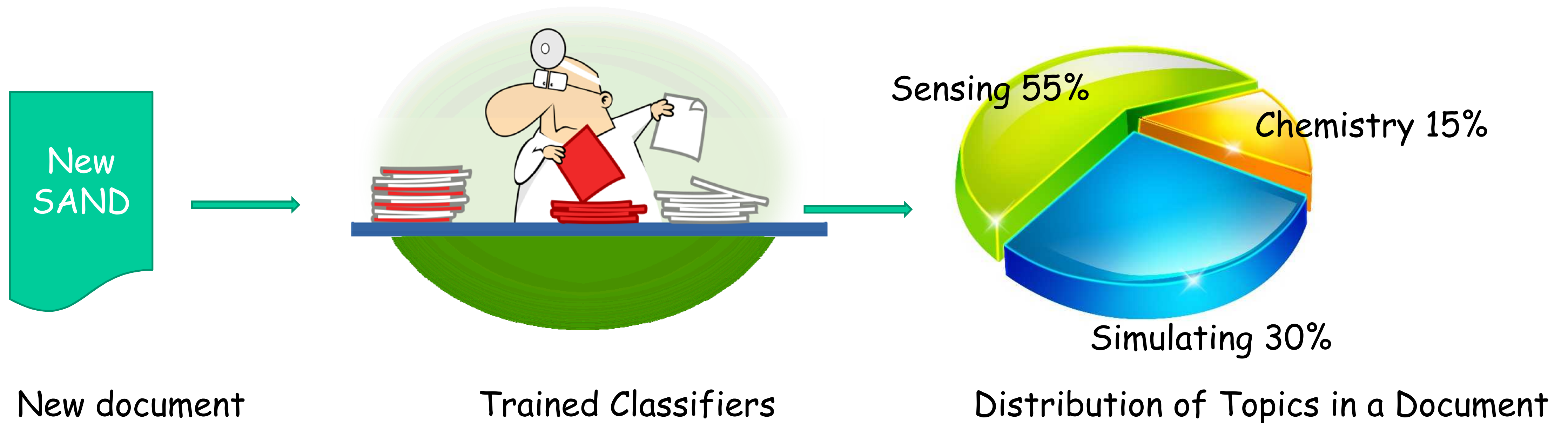


Supervised learning



Build SAND Report Classifiers with Deep Learning

1. Classify documents into proper classes
2. Recognize the document class in various formats
3. Recognize the distribution of possible classes in a document





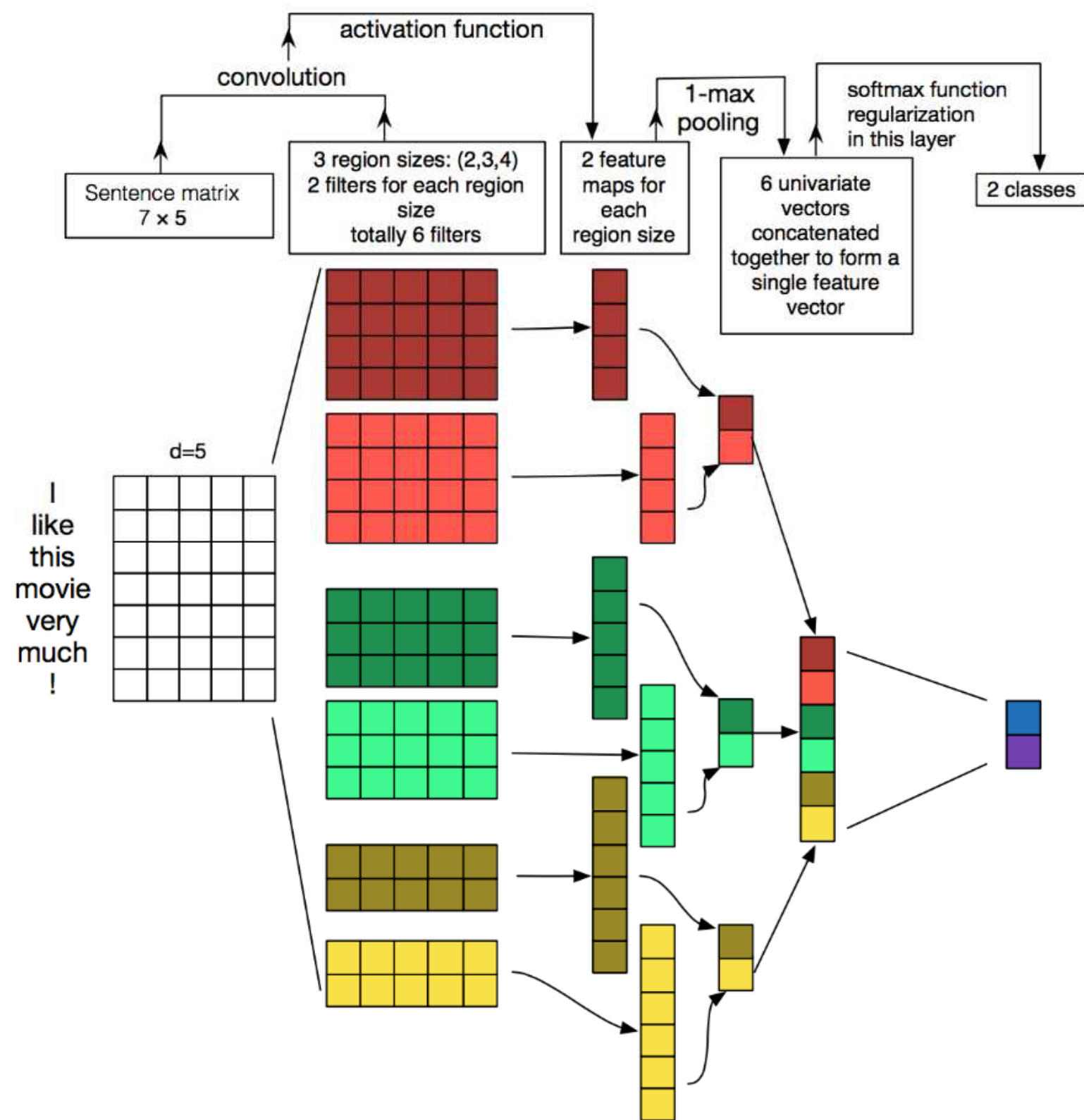
Procedures

- ◆ Collected ~100,000 SAND reports over last 50 years
- ◆ Data cleaned and indexed with Lucene
- ◆ Built "Taxonomy" for Sandia Category Guide (SCG)
- ◆ Selected the highly ranked documents with SCG taxonomy terms/phrases as the training sets using Lucene
- Tokenized the terms in documents with a 200 dimension numerical vector
- ◆ Built a CNN network
 - 3 one dimension, 5 step convolutional layers with 128 feature maps
 - Three layers of maxpooling
 - ReLu and Softmax as the activation functions
 - Loss function = categorical_crossentropy
- ◆ Trained the network with various hyperparameters

CNN for Text Classification

<http://www.wildml.com/2015/11/understanding-convolutional-neural-networks-for-nlp/>

```
print('Training model.')
```



```
# train a 1D convnet with global maxpooling
sequence_input = Input(shape=(MAX_SEQUENCE_LENGTH,),
dtype='int32')
embedded_sequences = embedding_layer(sequence_input)
x = Conv1D(128, 5, activation='relu')(embedded_sequences)
x = MaxPooling1D(5)(x)
x = Dropout(0.5)(x)
x = Conv1D(128, 5, activation='relu')(x)
x = Dropout(0.5)(x)
x = MaxPooling1D(5)(x)
x = Conv1D(128, 5, activation='relu')(x)
x = Dropout(0.5)(x)
x = MaxPooling1D(35)(x)
x = Flatten()(x)
x = Dense(128, activation='relu')(x)
preds = Dense(len(labels_index), activation='softmax')(x)
model = Model(sequence_input, preds)
model.compile(loss='categorical_crossentropy', optimizer='rmsprop',
metrics=['acc'])
```

Modified from Keras example



Example of CNN Classifier Outputs

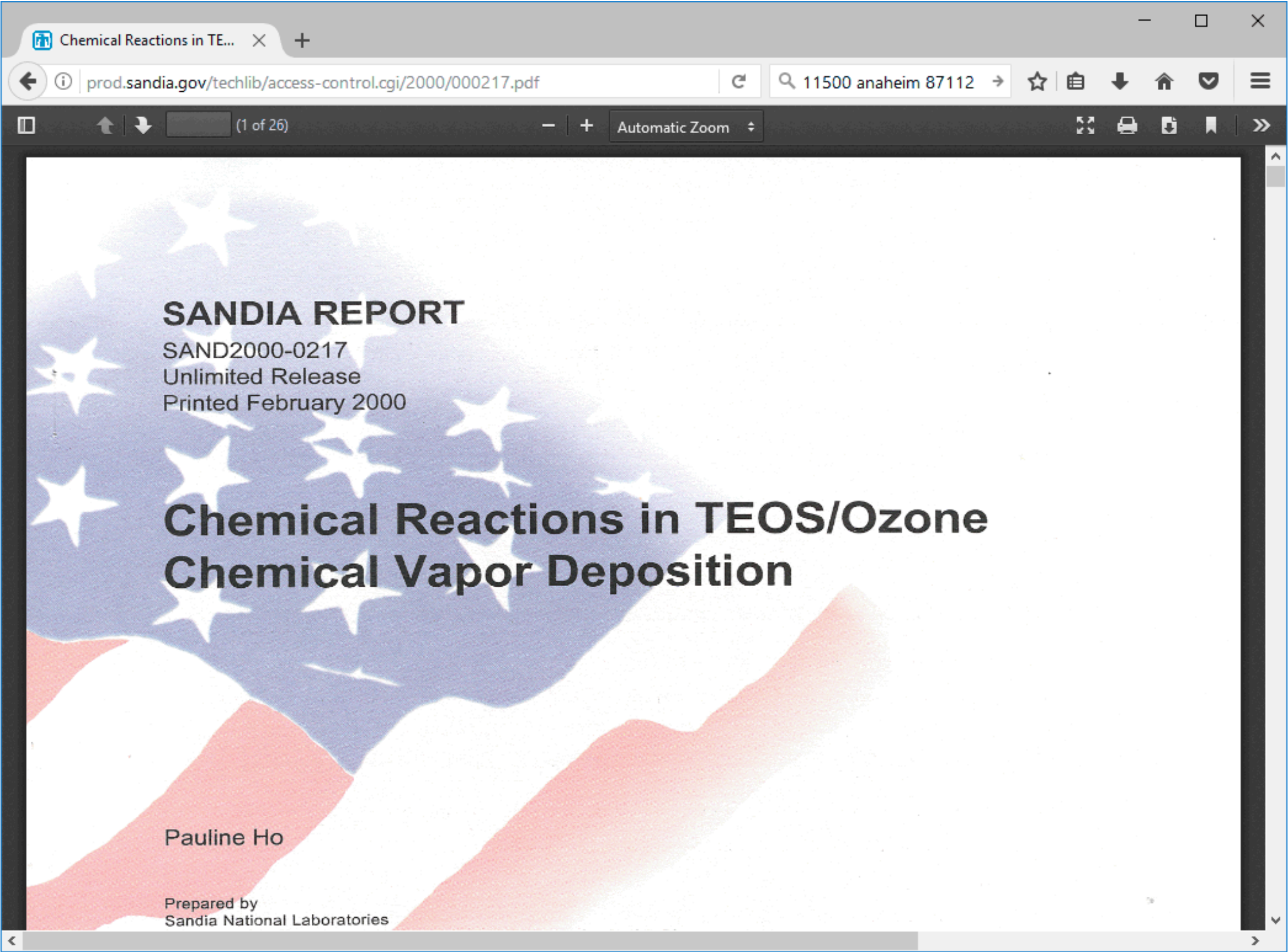
```
prediction = model.predict(data[146:150])
K=2
for p in range(0, prediction.shape[0]):

    a=np.array(prediction[p])
    b=np.argsort(a, -K)[-K:]
    np.set_printoptions(precision=3)
    print(b, np.take(a, b)*100, '%', '\t', titles[p+146], '\t', labels_name[b[0]], '\t', labels_name[b[1]])
```

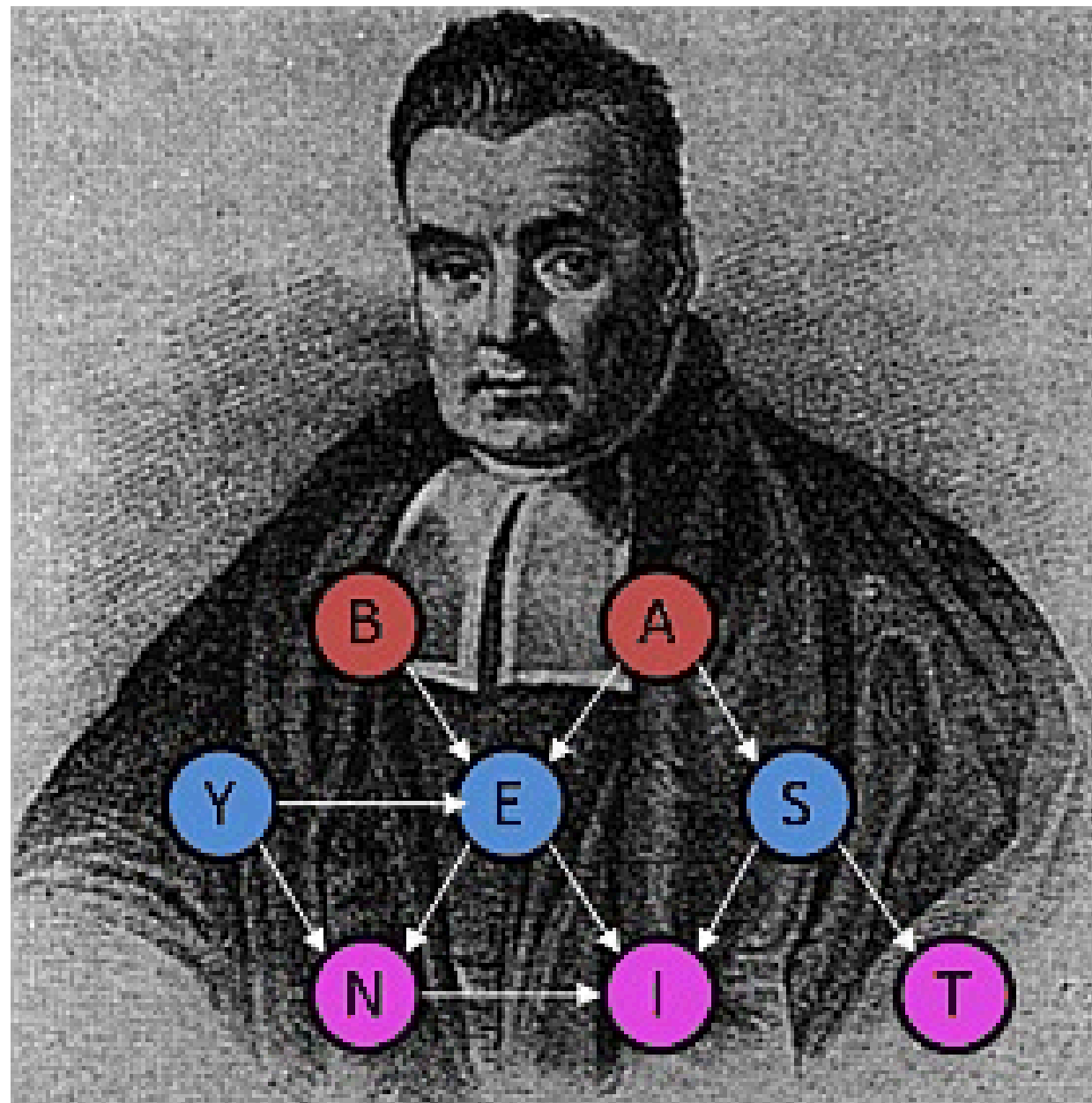
```
[37 1] [ 6.892 79.014] % SAND2000-0217.txt thermodynamics atmospheric science
[30 14] [ 6.755 91.043] % SAND2000-0218.txt particle physics electric and electronic
[16 31] [ 6.668 60.088] % SAND2000-0221.txt fluid mechanics plasma physics
[ 9 29] [ 29.797 52.055] % SAND2000-0222C.txt computer architecture optic
```



[37 1] [6.892 79.014] % SAND2000-0217.txt thermodynamics atmospheric science



III. Building Graphic Model for Prediction



$$P(x|E) = \frac{P(x) * P(E|x)}{P(E)}$$

Prior probability

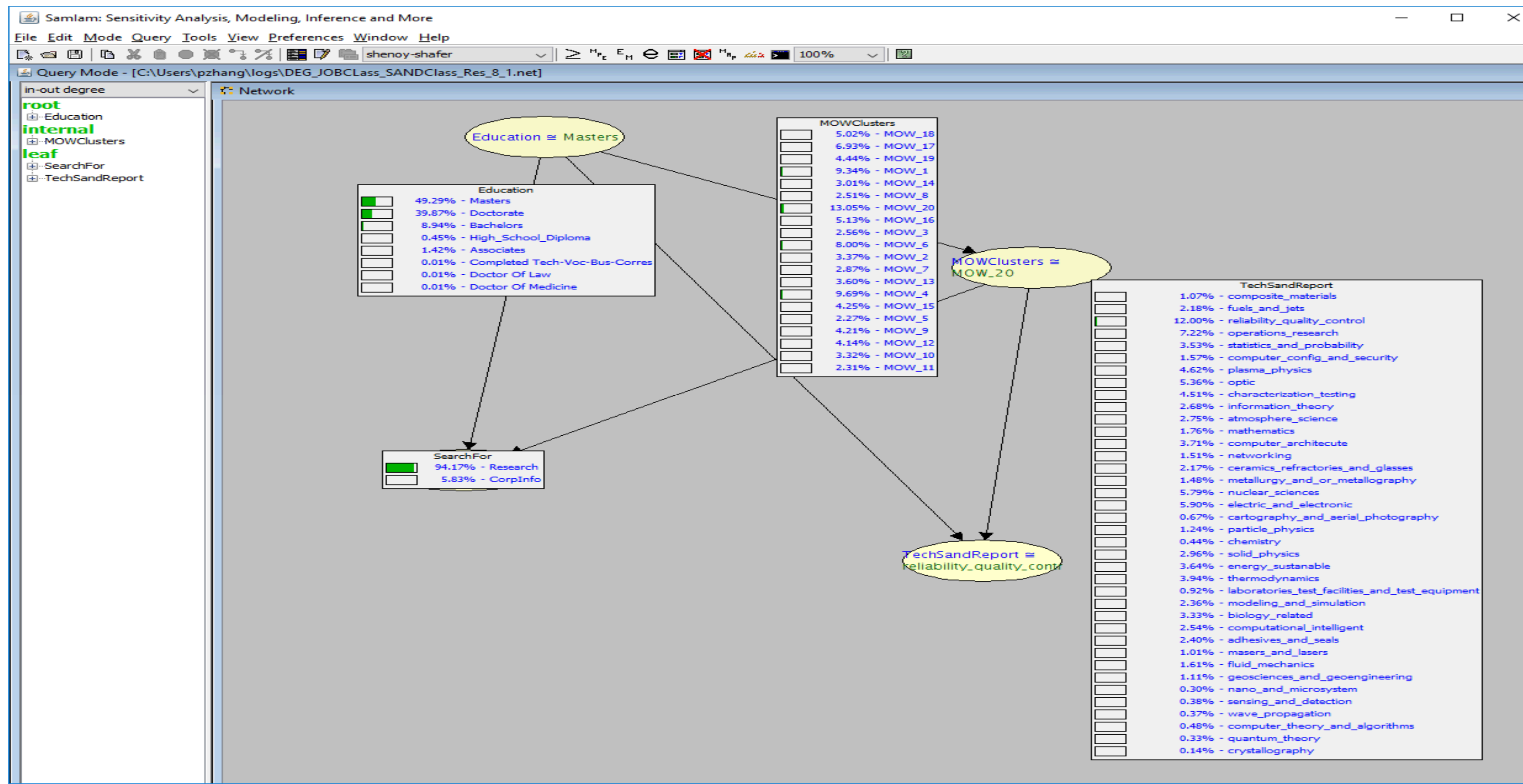
Likelihood of the evidence of 'E'
If the hypothesis 'x' is true

Posterior probability of 'x'
given the evidence 'E'

Prior probability that
the evidence itself is true

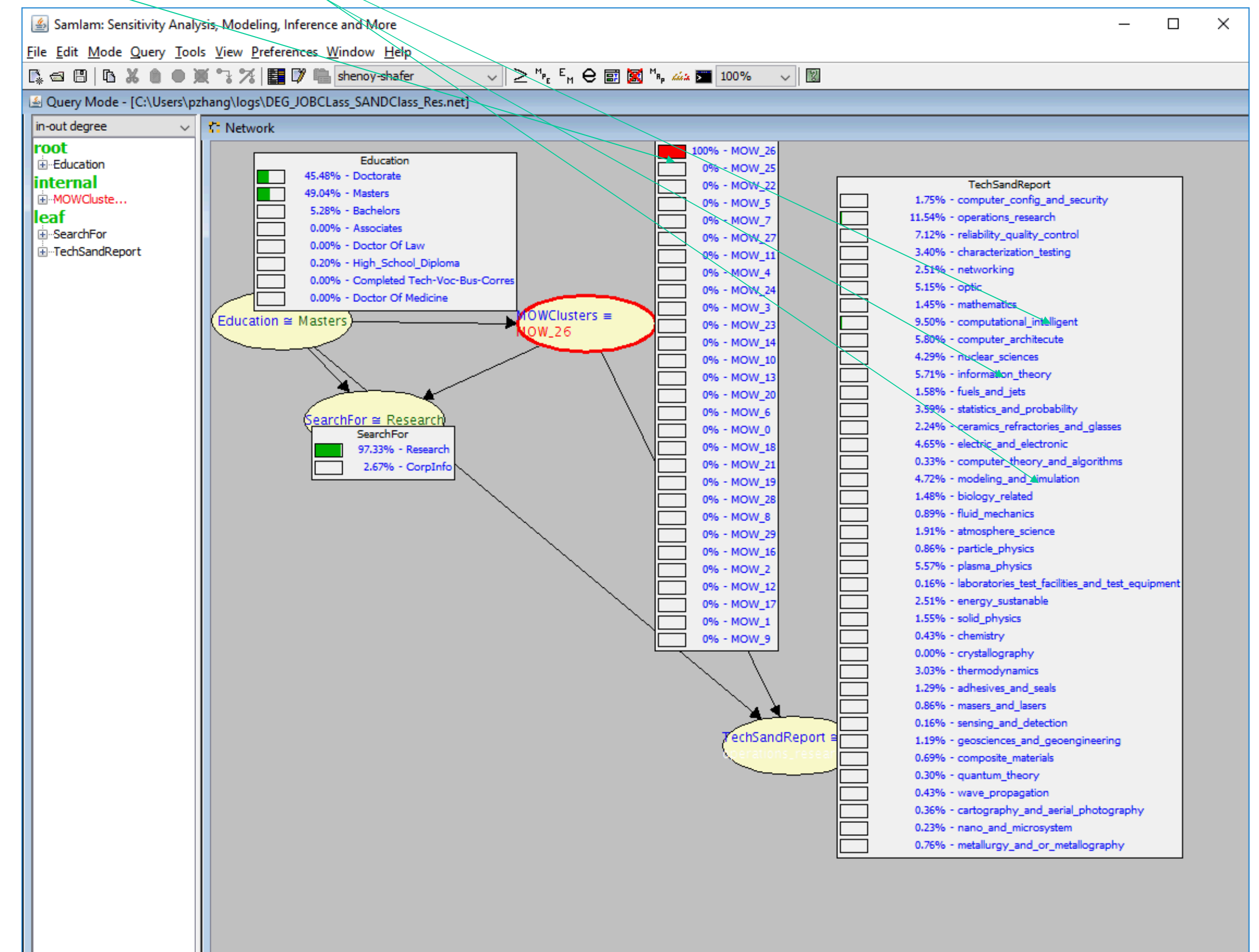
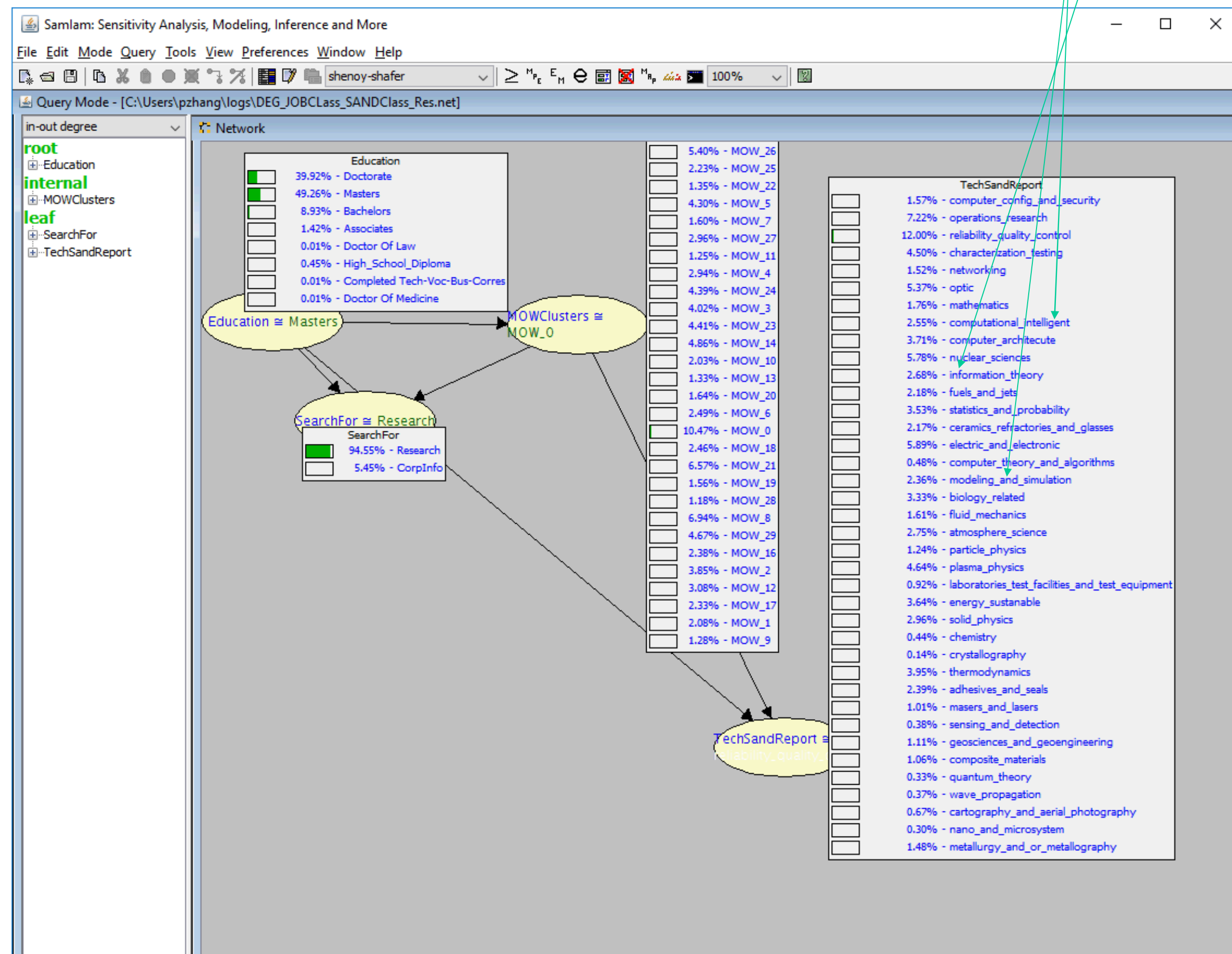
Build Graphic Models With the Conditional Probabilities

1. The behaviors of users on Solr based Enterprise Search Engine
2. Attributes of Clusters of Users
3. Build the conditional probability tables and create the Bayesian Network



Predict Information Needed for a Given MOW

Computational Intelligent from 2.55 to 9.5%
Modeling and Simulation from 2.36 to 4.72%
Information Theory from 2.69 to 5.71%



Example of Recommendation based on the Predictive Model

```
workspaceSpire - Java - modelForLabRDSE/src/gov/sandia/spire4/makeRecommendation.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
  autoLabelClasses
  BuildProfiles
  modelForLabRDSE
    src
      gov.sandia.spire4
        fromUserNameToRecommendedSand.java
        generateTestUrls.java
        Integrator.java
        makeRecommendation.java
        pairClass.java
        ProbabilityQuery_SAND_5_12.java
        readNetFileGetNormalDistribution.java
        recommendDoc.java
        SandProbByMowCluster.java
        utilities.java
  JRE System Library [JavaSE-1.8]
  Referenced Libraries
  data
  lib

modifySCG.java  queryDocsWithSCGs.java  copyFilesToClusterDir.java  labelClusters_CombinedSCG.java  makeRecommendation.java
28      while (true) {
29          System.out.print("> ");
30          String input = br.readLine();
31          System.out.println(input);
32          if (input == EXIT_Command) {
33              return;
34          }
35          username = input;
36          getInterestedSANDClassesForTheUser(username);
37          getSANDsToRecommend();
38          try {
39              List<String> list = recommendation(getSANDReportToTheFrontPage());
39
Problems @ Javadoc Declaration Search Console
makeRecommendation [Java Application] C:\Program Files\Java\jre1.8.0_121\bin\javaw.exe (Jul 31, 2017, 10:59:05 AM)
> tjdrael
tjdrael
Marginal probability tables:|
Interested Sand Classes for User: tjdrael are:
  computational_intelligent
  information_theory
  modeling_and_simulation

For this SAND Report: http://prod.sandia.gov/techlib/access-control.cgi/2016/160621m.pdf belongs to the Class of: modeling_and_simulation
Recommended SAND Report: http://prod.sandia.gov/techlib/access-control.cgi/2017/170582d.pdf
Recommended SAND Report: https://prod.sandia.gov/techlib/auth-required.cgi/2016/1612552c.pdf
Recommended SAND Report: https://prod.sandia.gov/techlib/auth-required.cgi/2016/168706c.pdf
Recommended SAND Report: http://prod.sandia.gov/techlib/access-control.cgi/2016/163773pe.pdf
Recommended SAND Report: http://prod.sandia.gov/techlib/access-control.cgi/2016/163774pe.pdf
For this SAND Report: http://prod.sandia.gov/techlib/access-control.cgi/2016/167809pe.pdf belongs to the Class of: computational_intelligent
Recommended SAND Report: http://prod.sandia.gov/techlib/access-control.cgi/2017/170406.pdf
Recommended SAND Report: https://prod.sandia.gov/techlib/auth-required.cgi/2016/1610652c.pdf
Recommended SAND Report: https://prod.sandia.gov/techlib/auth-required.cgi/2016/1610378c.pdf
Recommended SAND Report: http://prod.sandia.gov/techlib/access-control.cgi/2016/1610426.pdf
Recommended SAND Report: https://prod.sandia.gov/techlib/auth-required.cgi/2016/164354c.pdf
For this SAND Report: http://prod.sandia.gov/techlib/access-control.cgi/2005/056137.pdf belongs to the Class of: information_theory
Recommended SAND Report: http://prod.sandia.gov/techlib/access-control.cgi/2017/170402pe.pdf
Recommended SAND Report: http://prod.sandia.gov/techlib/access-control.cgi/2016/168558a.pdf
Recommended SAND Report: https://prod.sandia.gov/techlib/auth-required.cgi/2016/165208t.pdf
Recommended SAND Report: http://prod.sandia.gov/techlib/access-control.cgi/2016/164712pe.pdf
Recommended SAND Report: https://prod.sandia.gov/techlib/auth-required.cgi/2016/167943c.pdf
>
```



Implementation of Personalized Recommendations

The screenshot shows a web application interface for Sandia National Laboratories. The browser tabs include 'Dashboard', 'phrase2vec-dev', 'Review and Approval', 'Techweb - Home', and 'Search Results'. The address bar shows the URL: `infod.sandia.gov/lpberg/sites/archive/SEARCH/RelatedSAND/#g=1&p=search_results`. The search bar contains the text 'radar'. The left sidebar has a 'PAGES' section with 'Search Results' and 'Related SAND documents'. The main content area shows 'Showing 1 - 25 of 6292 results'. A list of results is displayed, including 'Synthetic Aperture Radar (SAR) External Homepage', 'TTU Advanced Doppler Radar Sandia Energy', 'Tonopah Test Range', and 'Radar Remote Sensing - Technology Symposium'. A yellow callout box with an arrow points to the 'Related SAND reports' link in the first result, containing the text: 'Single, subtle link. Consider playing with different names: "Related SAND", "Related SAND docs"'. The sidebar also lists various categories with counts: SAND Reports (3749), Sandia External Web (687), SharePoint (653), Sandia News (597), Sandia Internal Web (591), Sandia Videos (10), Corporate Forms (2), and Health and Benefits (1).



Implementation of Personalized Recommendations

The screenshot displays a web application interface for Sandia National Laboratories, specifically the 'Related SAND documents' page. The browser's address bar shows the URL: `infod.sandia.gov/lpberg/sites/archive/SEARCH/RelatedSAND/#g=1&p=related_sand_documents`. The page features a sidebar with navigation options: PAGES, NOTES, and CONSOLE. The main content area includes a search bar, a list of tabs (Sandia, FileNet), and a section titled 'SAND reports related to' which lists several documents with their titles, URLs, and brief descriptions. Annotations with arrows point to specific elements on the page:

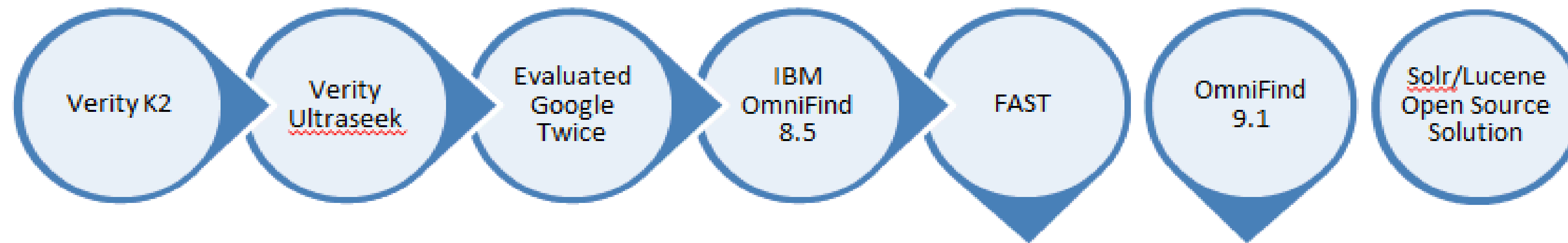
- A yellow box labeled 'search terms are cleared' points to the search bar.
- A yellow box labeled 'tabs remain to allow a fresh search' points to the 'Sandia' and 'FileNet' tabs.
- A yellow box labeled 'Simple header to give context to the page' points to the 'SAND reports related to' section header.
- A yellow box labeled 'Do you have access to the document titles? That might be more helpful here.' points to the document titles in the list.

The documents listed are:


- SAND2010-4288P**
<https://prod.sandia.gov/techlib/auth-required.cgi/2010/104288p.pdf>
SAND2010-4288 Demonstration of Ground Surveillance **Radar** Sandia is a multiprogram laboratory operated by Sandia Corporation
- SAND2012-6492P**
<https://prod.sandia.gov/techlib/auth-required.cgi/2012/126492p.pdf>
SAND2012-6492 **Radar** Image
- SAND2017-1817D**
<http://prod.sandia.gov/techlib/access-control.cgi/2017/171817d.pdf>
of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000. SAND No. 2017-XXXXP A Synthetic Aperture **Radar**



The Evolution of Enterprise Search at Sandia



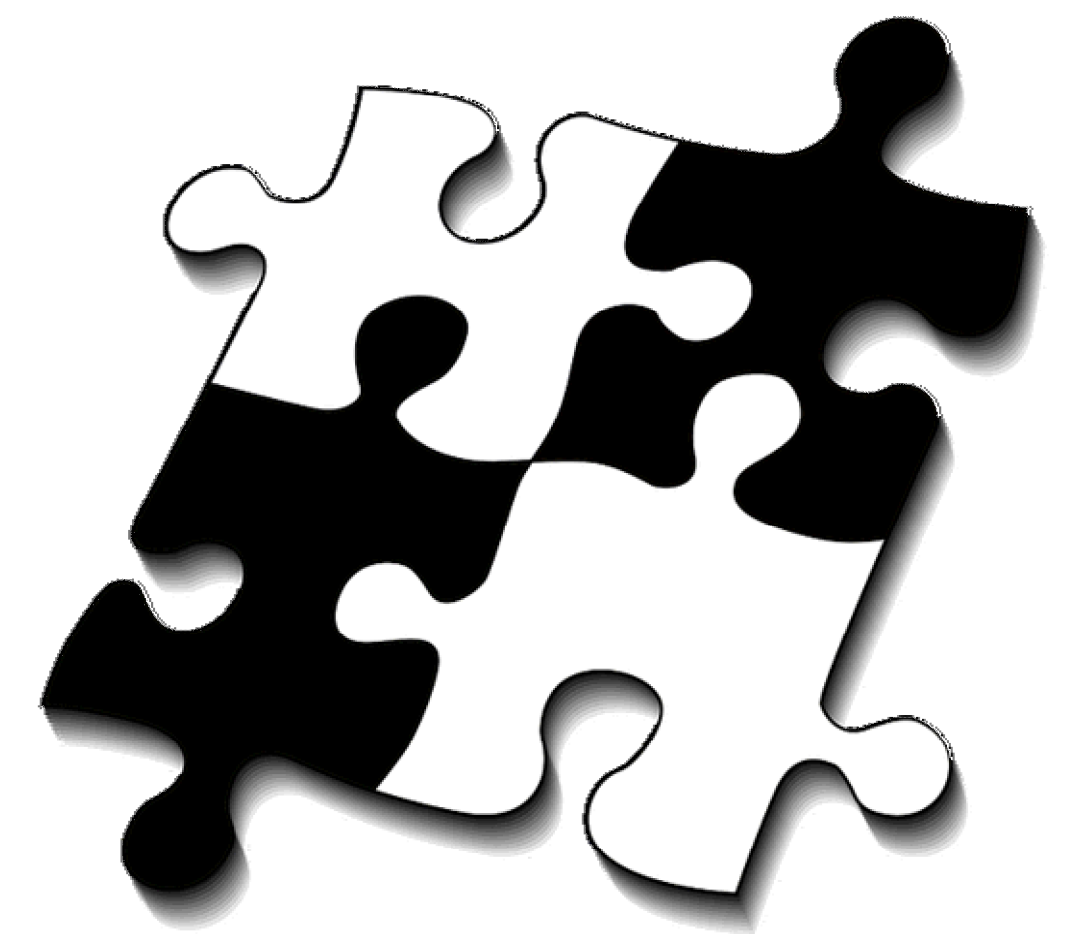
- We've gone through a number of search engine migrations
- Bad experiences with several COTS search engines
- As a result, selected Apache Solr, an open source platform
- Built our own search engine on top of Solr
- Developed custom client, crawlers, click tracking, click popularity boosting, faceting, and BestBets
- Selected Fusion based on a build vs buy decision process



Integrating our Analytics Capabilities with Fusion

Our department has ongoing analytics efforts including:

- Personalized recommendations using graphics models (SPIRE)
- Click popularity boosting (SVM in Matlab)
- Neural Network Recommender (using TensorFlow)
- PLUG recommender (collaborative filtering)
- Trending Query Terms





Handling overlapping capabilities

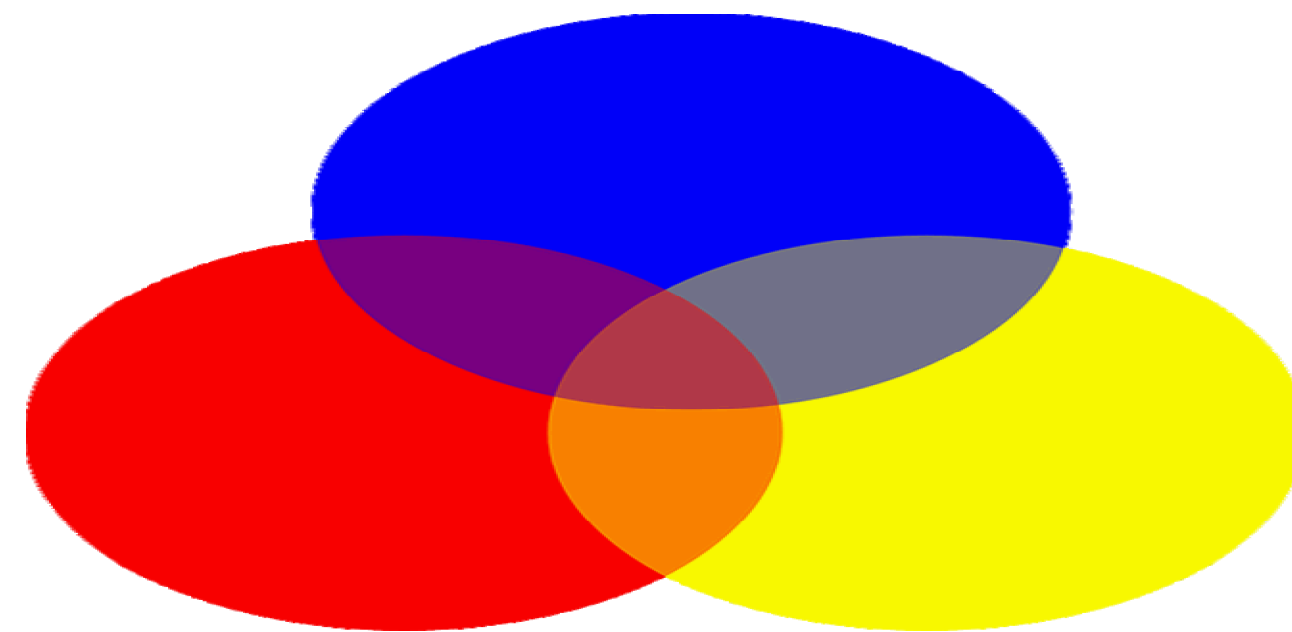
Our goal is to use as much capability from Fusion as possible

In areas where Fusion capabilities duplicate ours,

determine which capabilities perform best in our environment

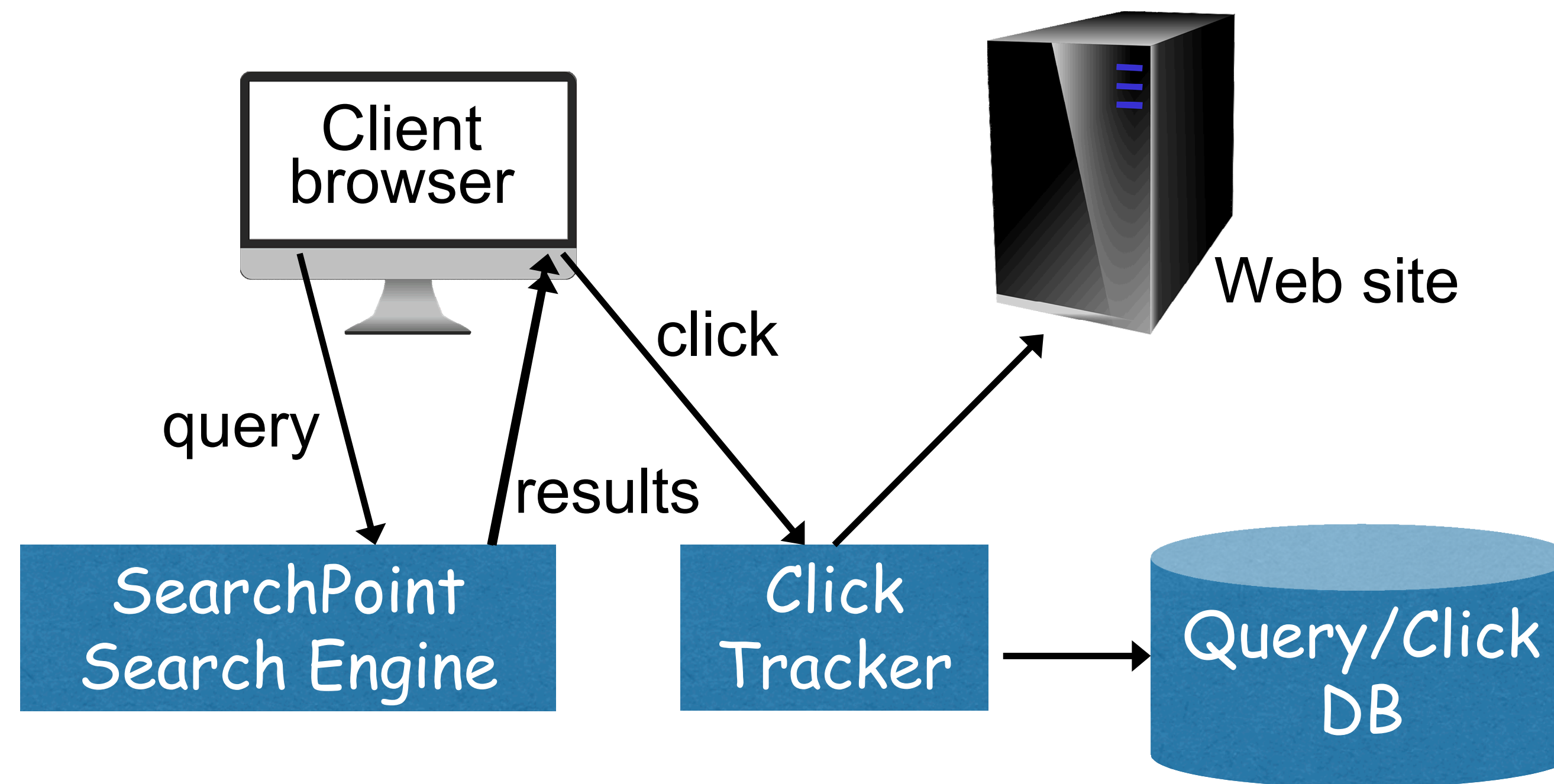
- Use our Search Engine Configuration Evaluator (SECE)
- Utilize QA people to manually compare features

If our own capability works better, integrate it in with Fusion



Case Study: Integrating click tracking and Signals

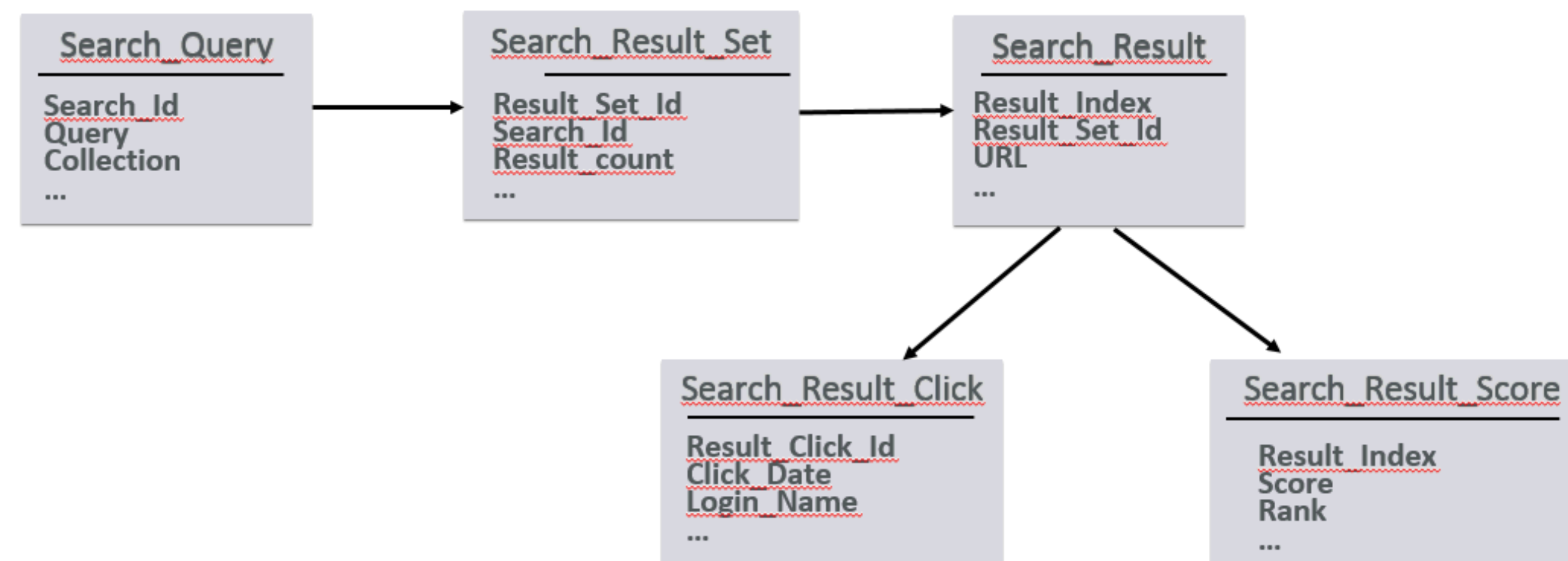
- The click tracking module stores click data in an Oracle DB
- Serves as primary source of data for many of our analytic models




Case Study: Integrating click tracking and Signals

Our custom click tracking database tracks:

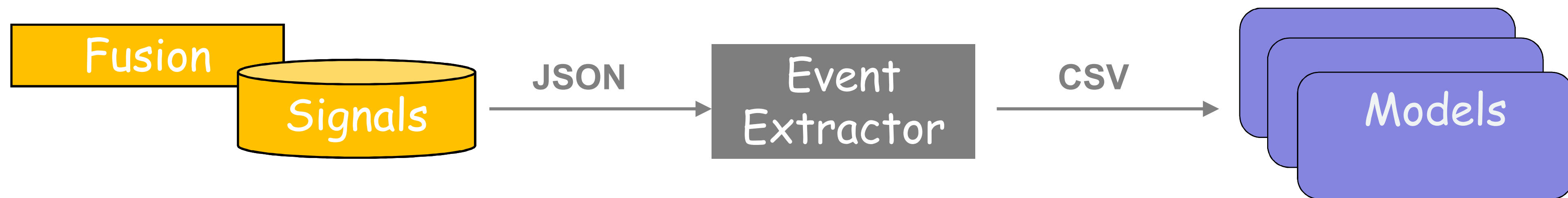
- Queries
- Results returned per query
- Result clicks
- Rank, Score ...





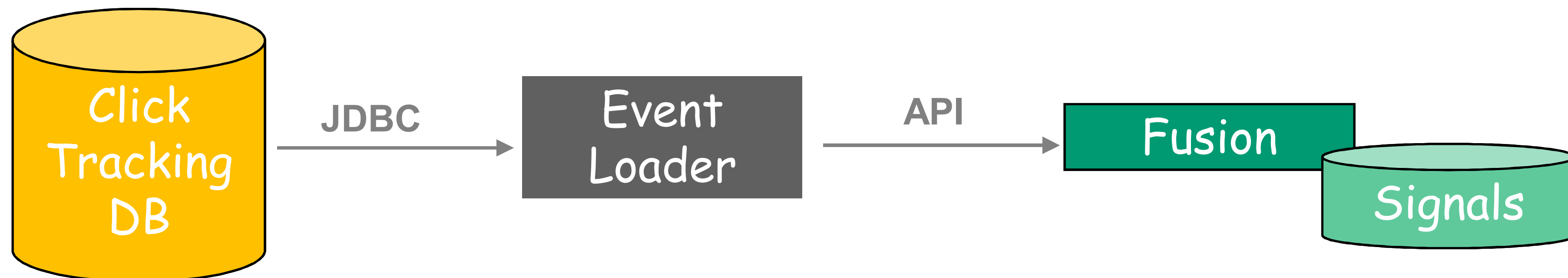
Integration option: Rely on Signals for event tracking

- Utilize the Fusion APIs to store click events directly as signals
- Maintain detailed and aggregated signals in Solr collections
- Set up jobs to export signals as needed by our local models
- May entail additional server costs for long term storage of signals



Integration option: Maintain source events in our DB

- Import events into Fusion as signals
- Aggregate signals and delete the detailed signal events
- Fusion models can utilize the imported signals directly
- Local models can still use the click tracking DB





Conclusion

- Application of combined machine learning and probabilistic models enables an enterprise search engine to intelligently satisfy users personal information needs
- Fusion's flexible environment provides a number of integration possibilities
- Using several integration points, we can leverage the analytics work we've been doing in the Fusion environment



Questions



LUCENE / SOLR REVOLUTION / 2017

SEPTEMBER 12-15, 2017
LAS VEGAS, NV



Thank You

Sandia National Laboratories

