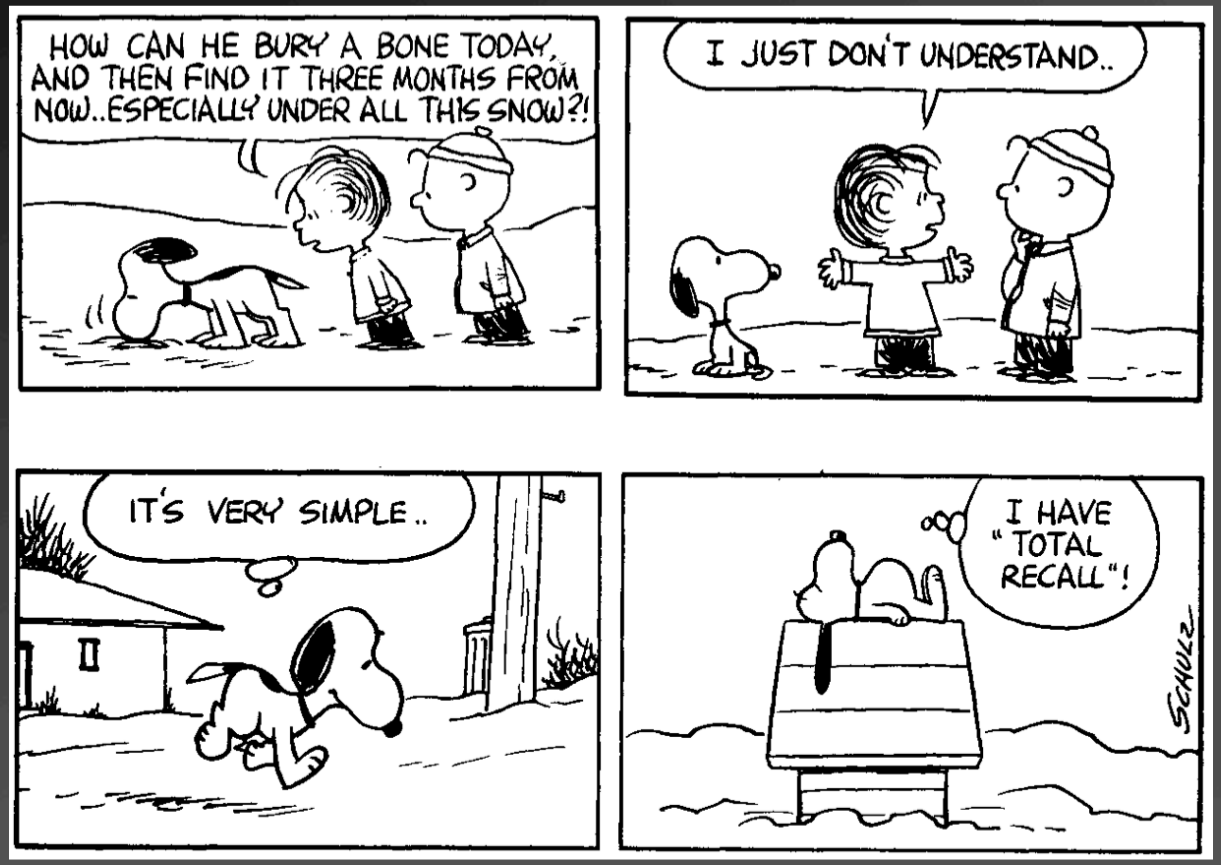
Information Extraction in an OCR Context

Ray E. Pereda
Information Science Research Institute
UNLV Computer Science Department

Information Extraction in an OCR Context

This work was done as part of several teams that included: Dr. Kazem Taghva, Dr. Thomas Nartker, Julie Borsack, Steve Lumos, Allen Condit, Jeffrey Coombs, myself, and others.

Total Recall



Overview

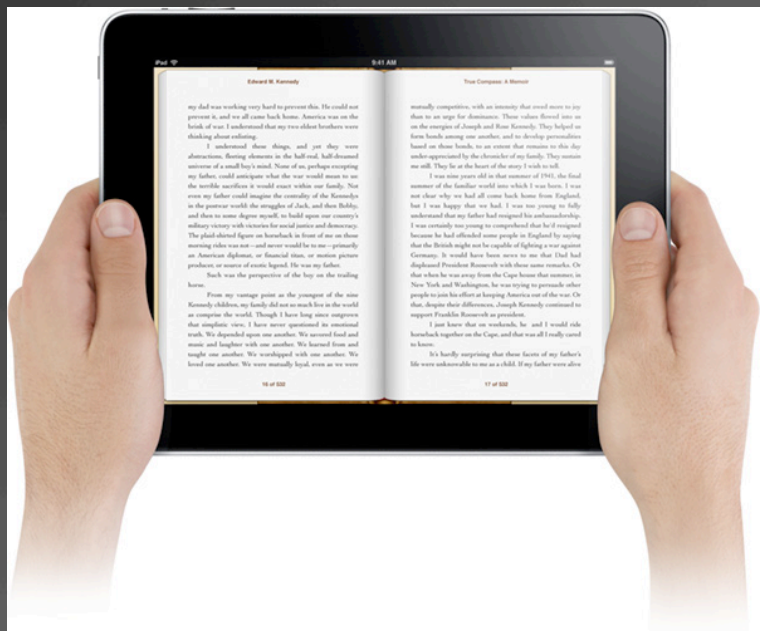
- What is Optical Character Recognition (OCR)?
- What is Information Retrieval (IR)?
  - Three models
  - Three experiments
- What is Information Extraction (IE)?
  - Three models
  - Three experiments
- Compare the effects of OCR on IR and IE

Scanning, OCR, and Digitizing

- Digitizing – converting a physical printed item into an electronic one.
- Reverse the order and it is simply called – printing.

Paperback Book

Book on an iPad

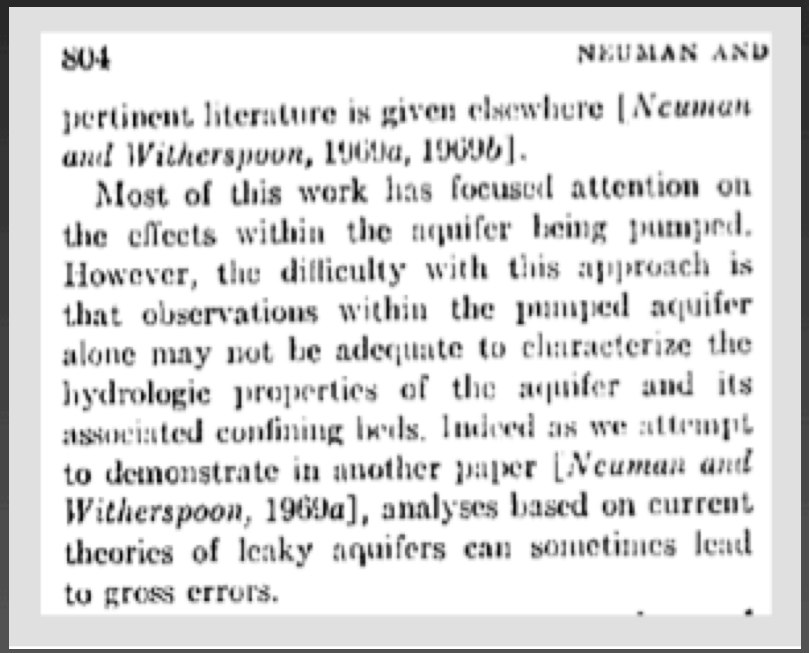


Scanning, OCR, and Digitizing

1. Page Scanning
   - Taking a digital picture of the page.
   - The camera might be a flatbed scanner or phone camera.
2. Block Zoning
   - Dividing up the pages into different logical areas.
     - E.g. The different columns of a news paper.
   - The blocks must be ordered
3. Text Recognition
   - Converting text blocks into a computer encoded characters.
   - OCR is another term for this step.

Scanning, OCR, and Digitizing

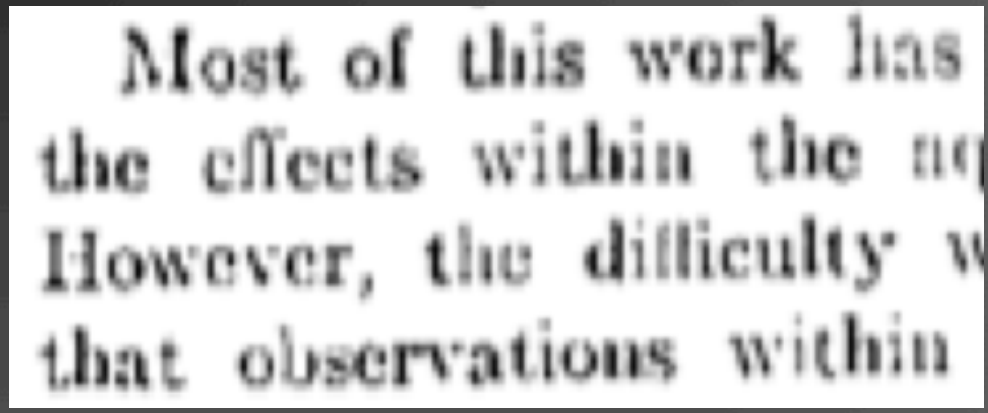
- Example #1 of a poor quality scan.
  - Deteriorated printed page. Notice the broken characters.



804 NEUMAN AND

pertinent literature is given elsewhere [*Neuman and Witherspoon*, 1969*a*, 1969*b*].

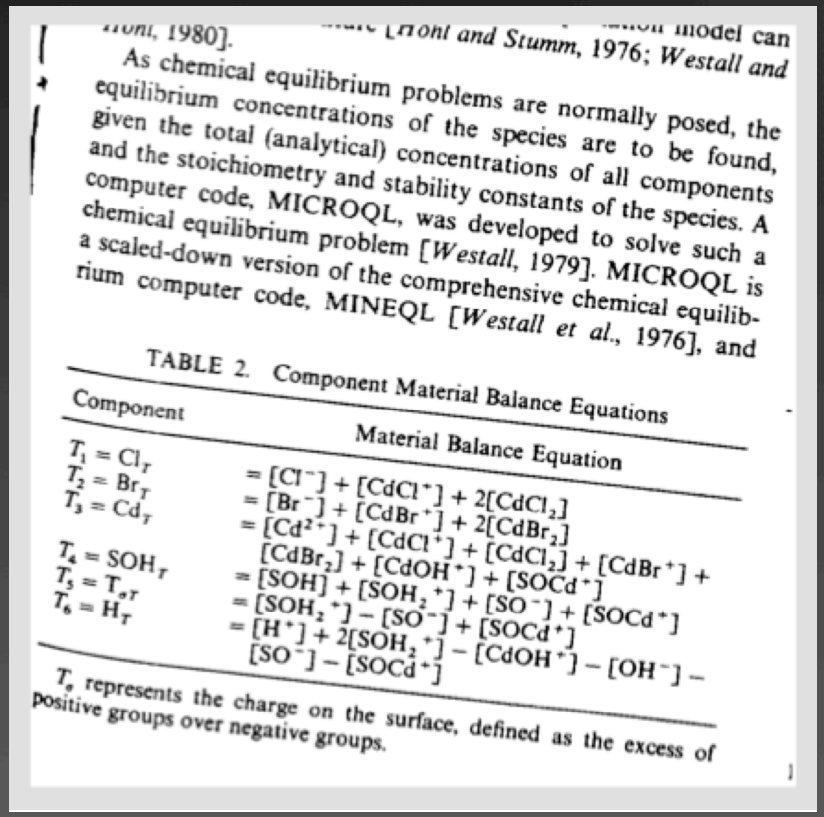
Most of this work has focused attention on the effects within the aquifer being pumped. However, the difficulty with this approach is that observations within the pumped aquifer alone may not be adequate to characterize the hydrologic properties of the aquifer and its associated confining beds. Indeed as we attempt to demonstrate in another paper [*Neuman and Witherspoon*, 1969*a*], analyses based on current theories of leaky aquifers can sometimes lead to gross errors.



Most of this work has
the effects within the
However, the difficulty
that observations within

Scanning, OCR, and Digitizing

- Example #2 of a poor quality scan.



Hohl, 1980]. *[Hohl and Stumm, 1976; Westall and* *model can*

As chemical equilibrium problems are normally posed, the equilibrium concentrations of the species are to be found, given the total (analytical) concentrations of all components and the stoichiometry and stability constants of the species. A computer code, MICROQL, was developed to solve such a chemical equilibrium problem [*Westall*, 1979]. MICROQL is a scaled-down version of the comprehensive chemical equilibrium computer code, MINEQL [*Westall et al.*, 1976], and

TABLE 2. Component Material Balance Equations

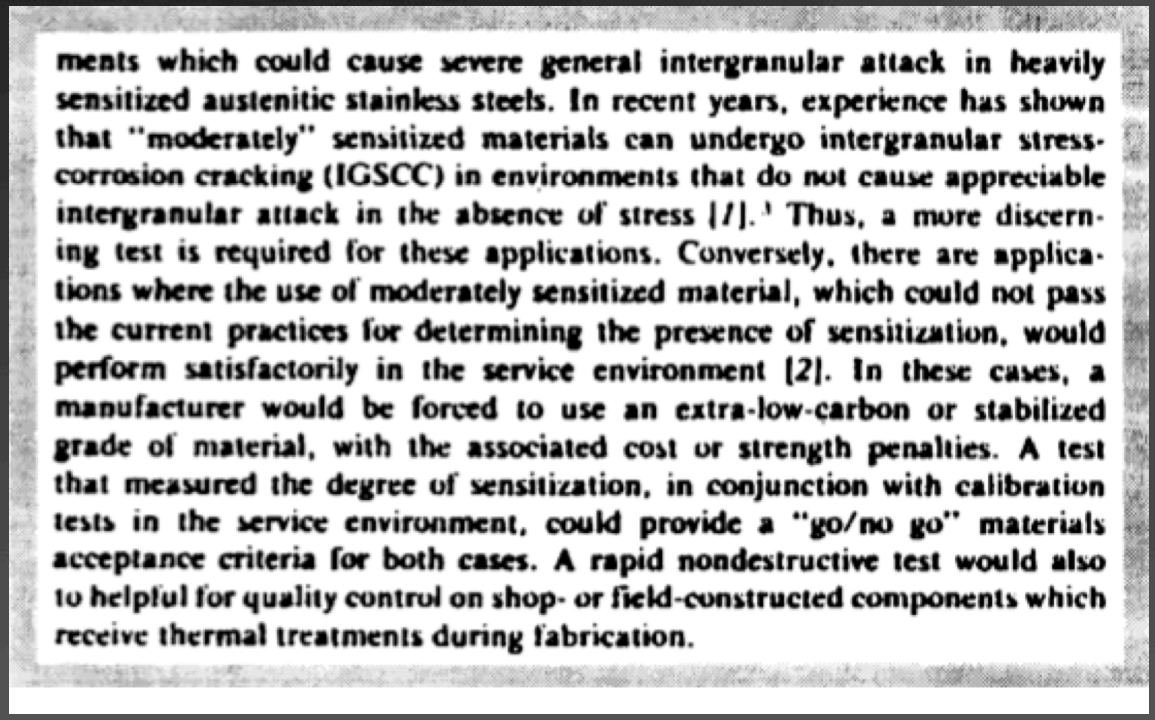
Component	Material Balance Equation
$T_1 = Cl_T$	$= [Cl^-] + [CdCl^+] + 2[CdCl_2]$
$T_2 = Br_T$	$= [Br^-] + [CdBr^+] + 2[CdBr_2]$
$T_3 = Cd_T$	$= [Cd^{2+}] + [CdCl^+] + [CdCl_2] + [CdBr^+] + [CdBr_2] + [CdOH^+] + [SOCd^+]$
$T_4 = SOH_T$	$= [SOH] + [SOH_2^+] + [SO^-] + [SOCd^+]$
$T_5 = T_{\sigma T}$	$= [SOH_2^+] - [SO^-] + [SOCd^+]$
$T_6 = H_T$	$= [H^+] + 2[SOH_2^+] - [CdOH^+] - [OH^-] - [SO^-] - [SOCd^+]$

T_σ represents the charge on the surface, defined as the excess of positive groups over negative groups.

- The page is not squared up.
- Left side of the page is bent back.

Scanning, OCR, and Digitizing

- Example of a typical scanned page



ments which could cause severe general intergranular attack in heavily sensitized austenitic stainless steels. In recent years, experience has shown that "moderately" sensitized materials can undergo intergranular stress-corrosion cracking (IGSCC) in environments that do not cause appreciable intergranular attack in the absence of stress [1]. Thus, a more discerning test is required for these applications. Conversely, there are applications where the use of moderately sensitized material, which could not pass the current practices for determining the presence of sensitization, would perform satisfactorily in the service environment [2]. In these cases, a manufacturer would be forced to use an extra-low-carbon or stabilized grade of material, with the associated cost or strength penalties. A test that measured the degree of sensitization, in conjunction with calibration tests in the service environment, could provide a "go/no go" materials acceptance criteria for both cases. A rapid nondestructive test would also to helpful for quality control on shop- or field-constructed components which receive thermal treatments during fabrication.

Recall and Precision Oxford English Dictionary

- Recall (definition *noun* 3b., 1961)

  "The effectiveness of an information retrieval system, expressed as the proportion of relevant items that are successfully retrieved by a particular search."

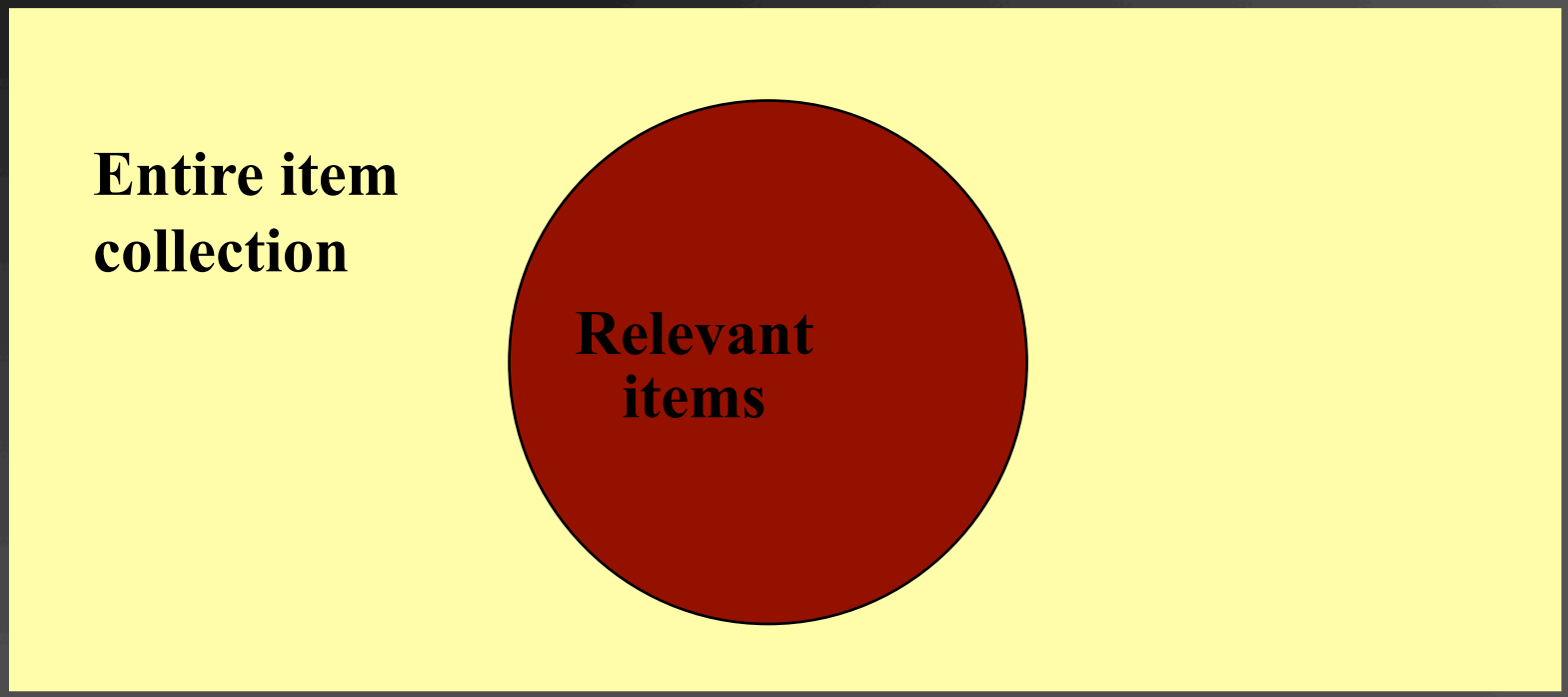
- Precision (definition *noun* 2e., 1965)

  "The accuracy of an information retrieval system, expressed as the proportion of items retrieved by a particular search that are relevant."

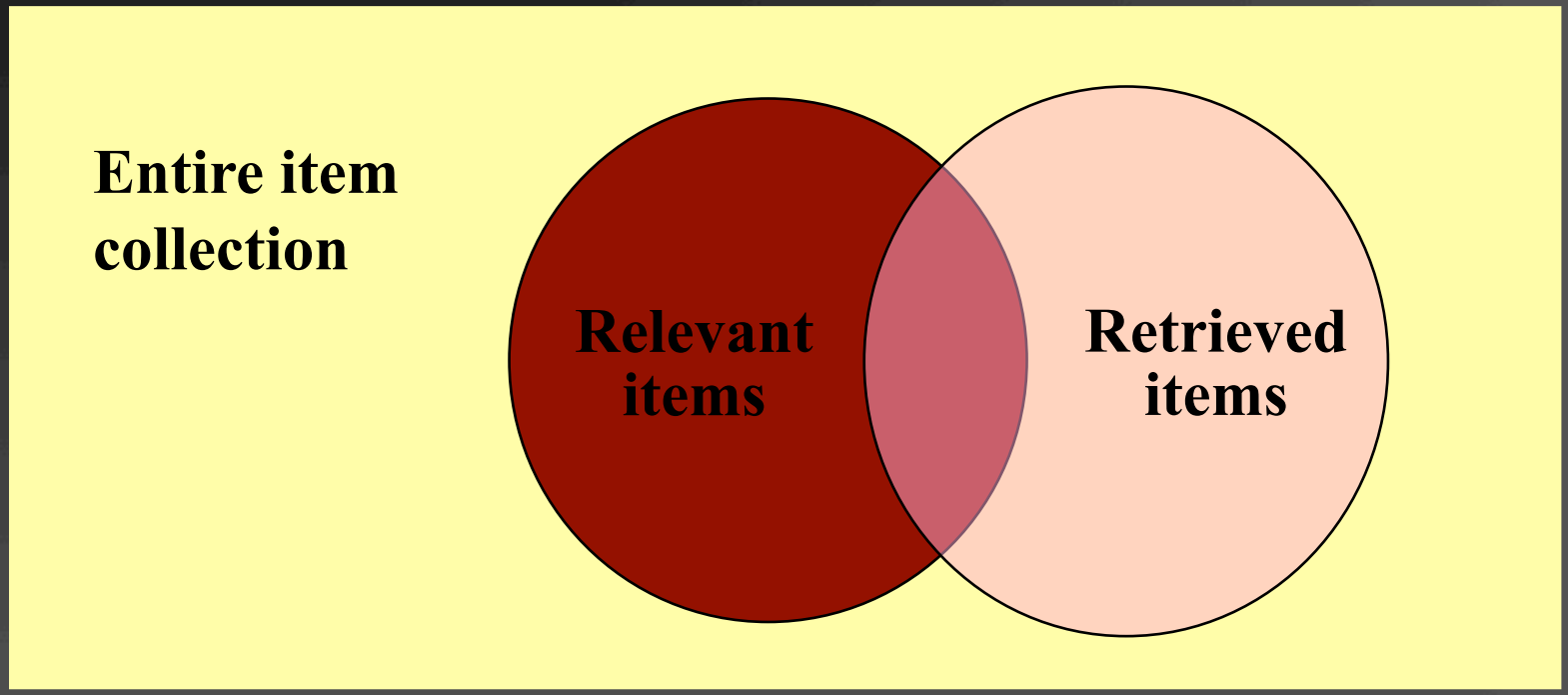
Recall and Precision

Entire item collection

Recall and Precision



Recall and Precision



Recall and Precision

- Positive – program reported a match
- Negative – program reported NO match
- True – the program gave the correct answer
- False – the program gave the wrong answer
- F1 score – the harmonic mean of precision and recall.

$$precision = \frac{TruePositives}{TruePostives + FalesPositives}$$

$$recall = \frac{TruePositives}{TruePositives + FalesNegatives}$$

$$F1 = \frac{2}{\frac{1}{precision} + \frac{1}{recall}}$$

Recall and Precision Intuitively

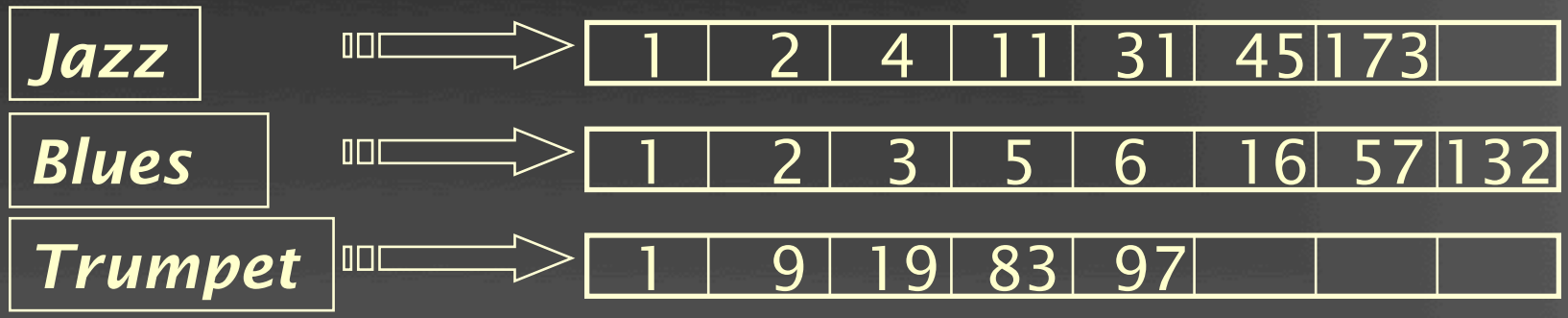
- Recall
  - How much of the target items are retrieved?
- Precision
  - How much junk was mixed with relevant items?
    - E.g. “Of 10 items retrieved, only 7 were relevant, 70% precision”
  - How much of the top-ranked items are relevant?
  - Look at the first “page” of results: what percent of the items are relevant?
  - What is the cut-off point in the list to get x % of the relevant items
    - E.g. “At 80% Recall, 10% Precision”
    - E.g. “At 20% Recall, 75% Precision”
- Recall and Precision are typically inversely related

IR – Boolean Model

- Query is a boolean expression: word1 AND word2
  - E.g. jazz AND trumpet
- An inverted index is a data structure for storing a special mapping.
- The mapping takes a single word and returns a list of documents containing that word.
- To compute the results for a query: use set operations
  - AND is computed with intersection
  - OR is computed with union

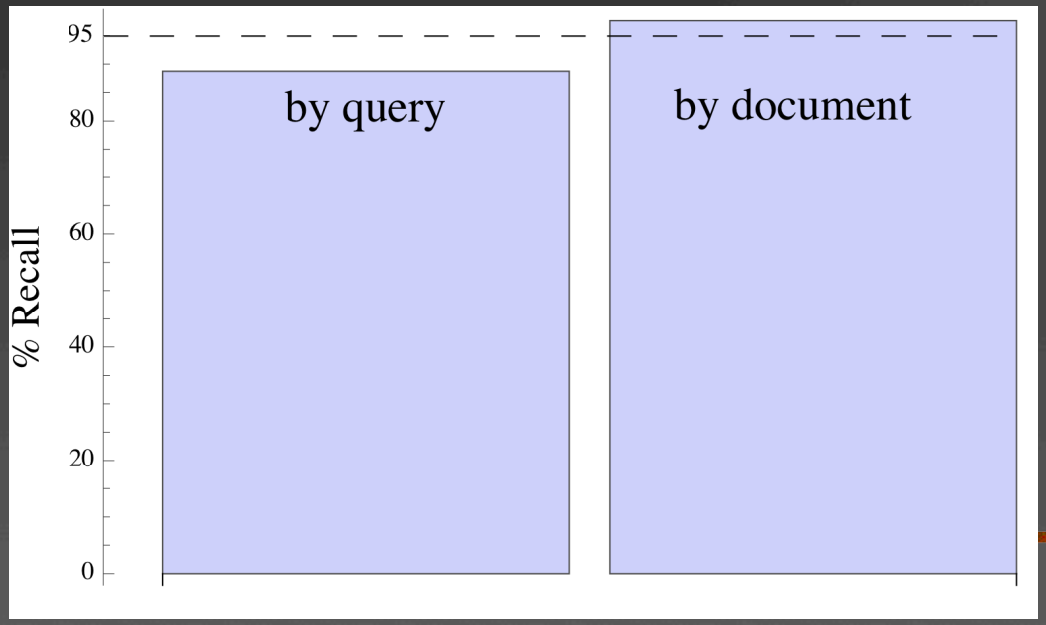
IR – Boolean Model

- For each word *w*,
  store a list of all documents that contain *w*.
- Identify each by a docID, a document serial number.
- These list are called Inverted Files or Posting Files
- Perform boolean set operations on list to answer queries.



IR – Boolean Model

- Recall by counting number of 100% correct query results, 88.8%
- Recall by counting each correct document across all queries, 97.6%



IR – Probabilistic Model

- Concepts are more abstract than single words
  - Can include phrases
  - Can include a collection of synonyms
- Relies on TF.IDF for estimating the probability that a document is about a concept
- TF.IDF – term frequency × inverse document frequency
- TF – count # of times word occurs in DOC
  - Normalize by dividing by total number words in DOC
- IDF – inverse of count # of times the word occurs in the COLLECTION
  - “Inverse of count” means Log(1/count)
  - Normalizing by × size of whole document collection

IR – Probabilistic Model

f_{ij} = frequency of term t_i in document d_j

$$TF_{ij} = \frac{f_{ij}}{\max_k f_{kj}}$$

n_i = number of docs that mention term i

N = total number of docs

$$IDF_i = \log \frac{N}{n_i}$$

TF.IDF score for document(i)-term(j) pair

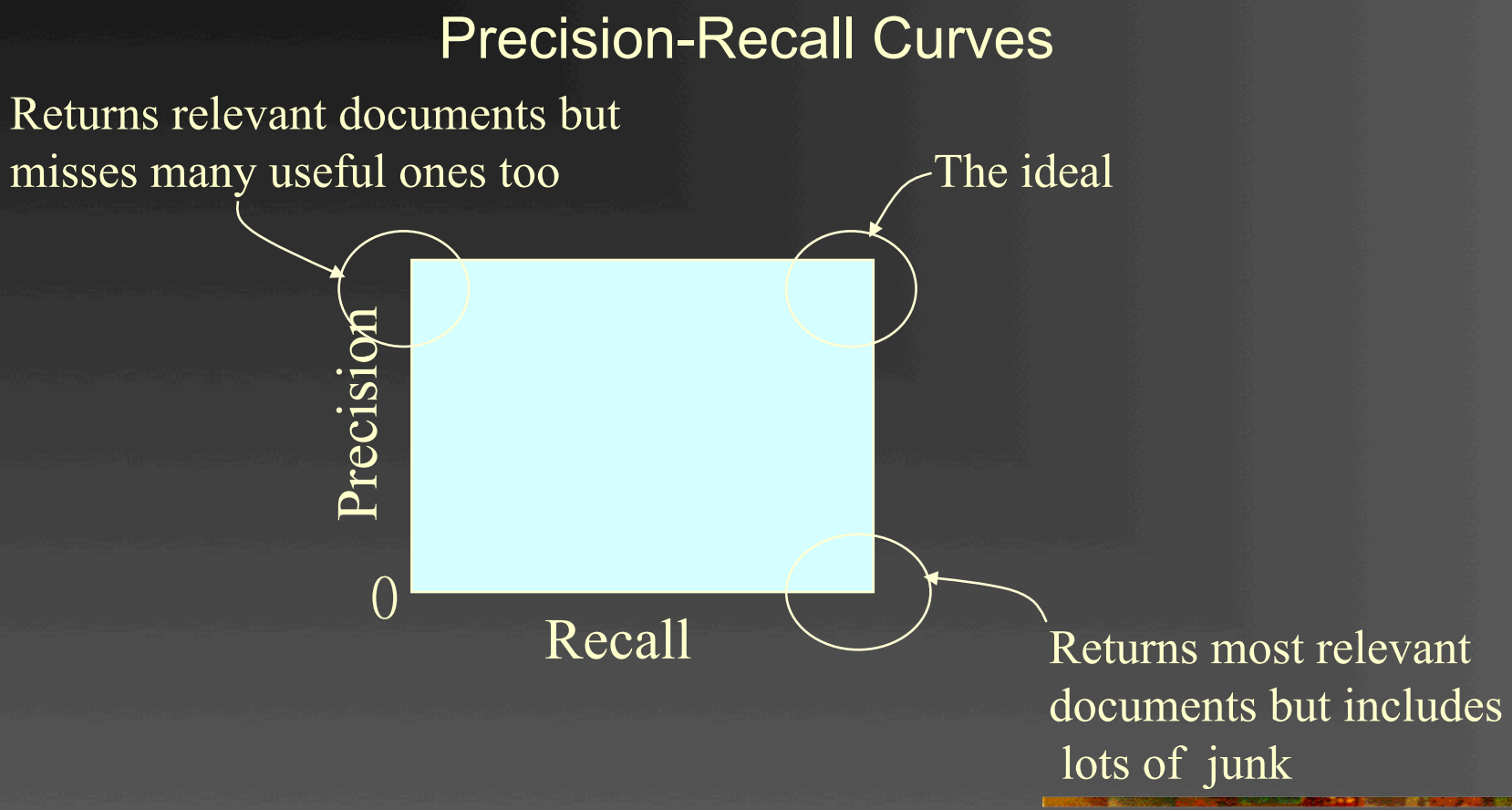
= $TF_{ij} \times IDF_i$

IR – Probabilistic Model

- Precision measured at standard recall points
- Notice that the % change is within 5% for all three OCR collections

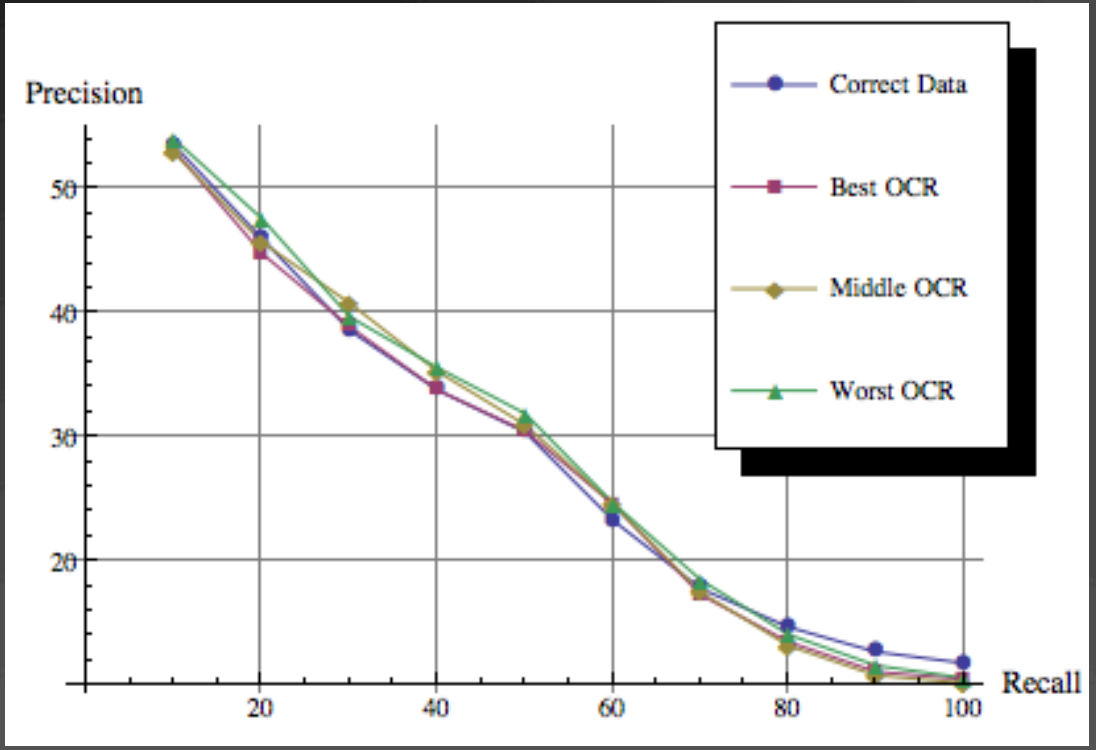
Recall	Correct Data	OCR Best	OCR Middle	OCR Worst
10	53.5	53.1	52.9	54.0
20	46.0	44.8	45.7	47.6
30	38.6	38.9	40.8	39.7
40	33.8	33.8	35.3	35.6
50	30.4	30.5	31.0	31.9
60	23.3	24.6	24.6	24.7
70	17.8	17.3	17.5	18.5
80	14.7	13.5	13.2	14.1
90	12.7	11.1	10.8	11.6
100	11.8	10.5	10.2	10.6
Average	28.3	27.8	28.2	28.8
% Change		-1.6	-0.2	2.0

IR – Probabilistic Model



IR – Probabilistic Model

Precision-Recall Curves



IR – Vector Space Model

$D = (d_1, \cdots, d_t)$ where each weight is non-zero if the term appears in the document (or query).

$$D \bullet Q = (c_D \times c_Q) \sum_{\text{terms}} (a_D \times a_Q) \times (b_D \times b_Q) \times d_i \times q_i$$

- **Term Frequency** Component: considers term frequency with respect to a single document
- **Collection Frequency** Component: considers term frequency with respect to the entire collection.
- **Vector Normalization** Component: treats long and short documents with some equality.

IR – Vector Space Model

The SMART notation for weighting schemes:

xxx.xxx

First three for document, second three for query.

- N – none
- A – augmented
- L – logarithm
- P – probabilistic
- S – square
- T – inverse doc freq or sum
- C – cosine

IR – Vector Space Model

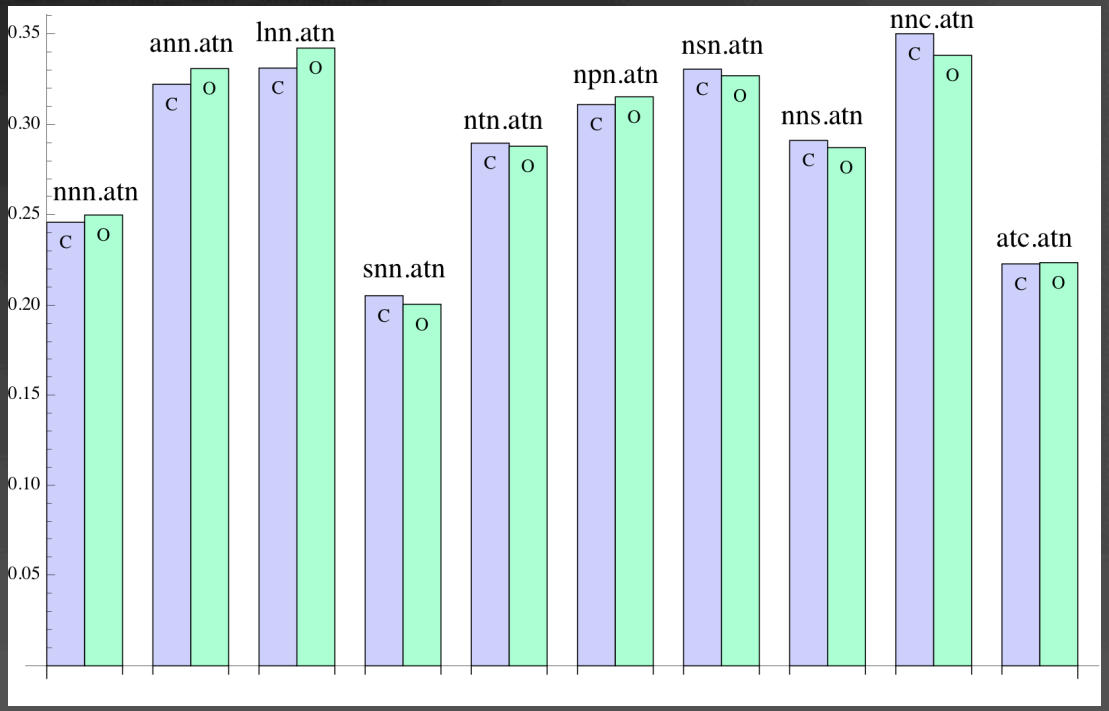
Experimental Results

Average Precision for Correct and OCR Collections

Weighting	Correct Data	OCR Data	% Difference
nnn.atn	0.2457	0.2497	1.63
ann.atn	0.3222	0.3308	2.67
lnn.atn	0.3311	0.3421	3.32
snn.atn	0.2053	0.2006	-2.29
ntn.atn	0.2898	0.2881	-0.59
npn.atn	0.3110	0.3153	1.38
nsn.atn	0.3305	0.3269	-1.09
nns.atn	0.2913	0.2873	-1.37
nnc.atn	0.3500	0.3381	-3.40
atc.atn	0.2228	0.2235	0.31

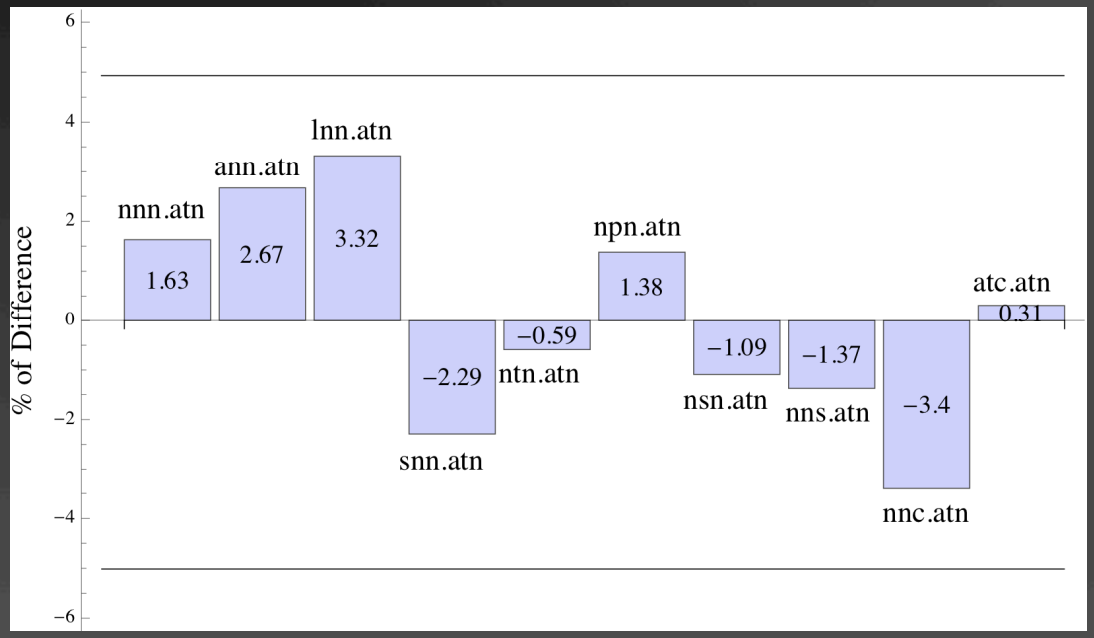
IR – Vector Space Model

Average Precision for Clean (c) and OCR (o) Collections



IR – Vector Space Model

% Change in Average Precision for Correct-OCR



IR – Effects of OCR

- Measured the difference between results on OCRed collection and clean collection

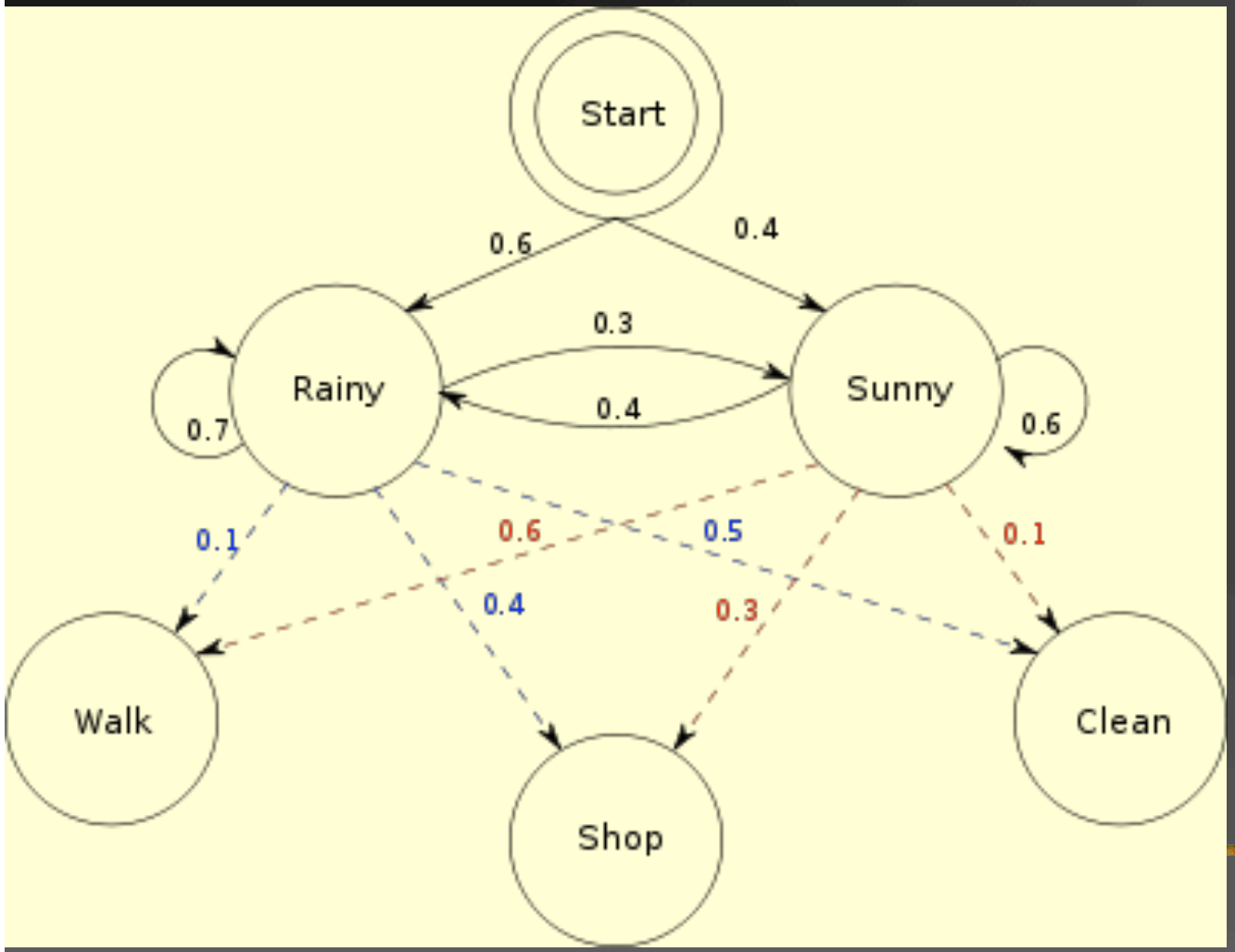
- Boolean Model: < 5 % Change Recall
- Probabilistic Model: < 5 % Change Average Precision
- Vector Space Model: < 5 % Change Average Precision

- Next: turn our attention to IE – information extraction

IE – Information Extraction

- Freedom of Information Act (1966)
  - disclosure of previously unreleased information and documents controlled by the United States Government.
- Privacy Act of 1974
  - Limits the release of personally identifiable information about individuals stored in federal information systems.
- Solution #1: Manual review of documents
  - blot out just the “private information”
- Solution #2: Automated Assisted Review of Documents
  - Information Extraction to the rescue

IE – Hidden Markov Models



An Overview

many more clear details on the web

Read Wikipedia:

- Markov Model
- Hidden Markov Model
- Viterbi Algorithm

IE – Hidden Markov Models

The sequence from a roll of a die (dice means two “die”) is:
1156631211315331652413416525241336314631412445315 6

You know the casino dealer is using a loaded dice.

die roll sequence (below, OBSERVABLE always):
3323645635366111112265143663412441111111111412231 44
FFFFFFFFFFFFLLLLFFFFFFFFFFFFFFFLLLLLLLLLFFFFFFF
fair-loaded sequence (above, HIDDEN but not while training):
F = fair die L = loaded die

- Suppose you study many of the sequence pairs like the one above.
- Given a sequence of just the OBSERVABLE sequence you could guess the HIDDEN sequence!

IE – Hidden Markov Models

Observable sequence (below #1):

1. car stopped at 4505 Maryland Las Vegas, NV 89154 before the getaway
2. W W W Num Sname CityName State ZipLike W W W
3. B B B A A A A A B B B

Hidden Address Sequence (above #3):

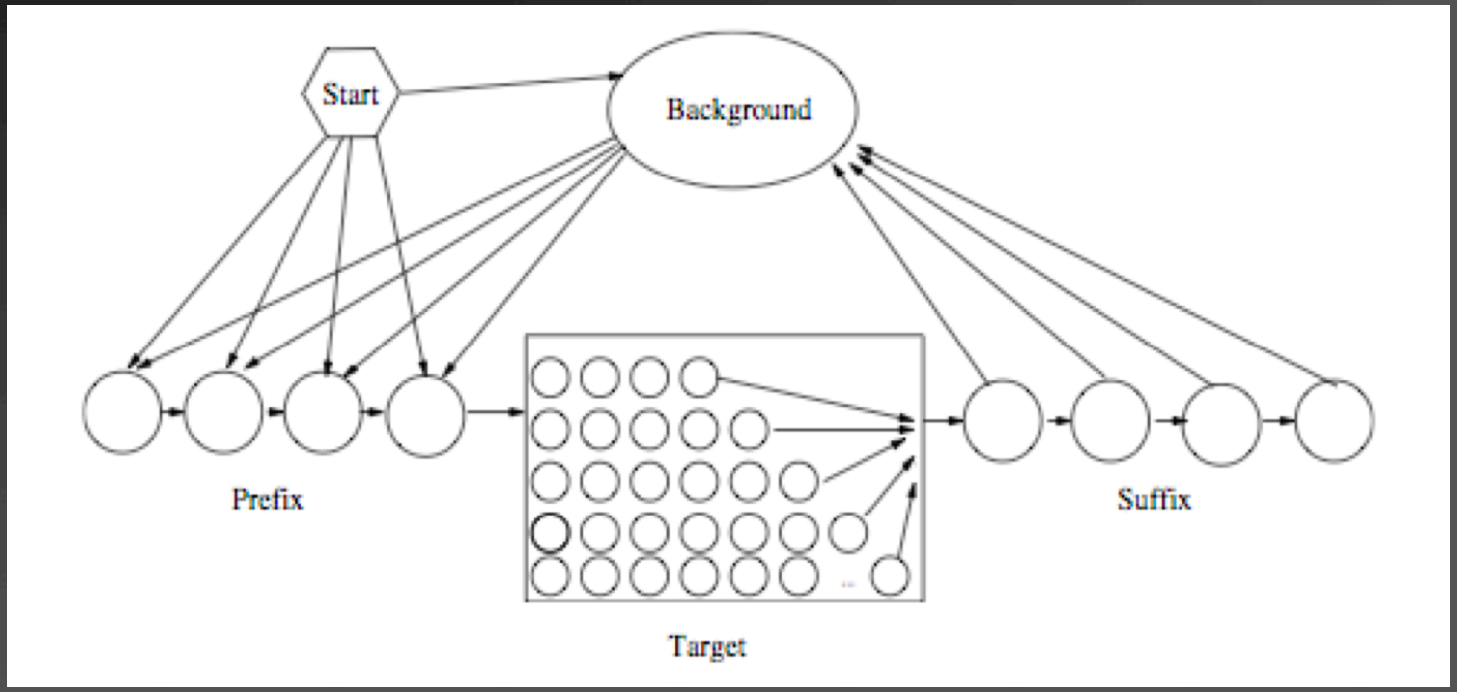
B = Background, Num = number, CityName, State, ZipLike…

A = Address, “this is part of an address”

- Train on text with hidden address sequences to learn the structure of postal addresses as expressed in free flowing OCR text.
- Should be able to identify dates even if they have an error or two.

IE – Hidden Markov Models

Freitag and McCallum



IE – Hidden Markov Models

HMM Hidden Symbols used for Address Extraction

symbol	*examples*
`comma`	final comma (,)
`colon`	final colon (:)
`ziplike`	89783, 89123-2319
`phonelike`	(555) 555-5555, 555-555-5555, 555-5555
`purenumber`	5, 5555
`containsnumber`	n5k
`mailterm`	mail,P.O.,address,apt
`roadname`	street, road, avenue, ave
`statename`	nevada, nv
`cityname`	las vegas, oak ridge
`pname`	joe, smith
`startcap`	Yucca, Mountain
`default`	yucca, tree, mountain

IE – Hidden Markov Models

Precision, Recall, and F1 Score

EXPERIMENT	TP	FP	FN	TN	PRECISION	RECALL	F1
DEFAULT	104	4	9	495	0.963	0.920	0.941
CLEAN	108	16	5	483	0.871	0.956	0.911
SHRINKAGE	108	28	5	471	0.794	0.956	0.867

IE – Hidden Markov Models

Precision and Recall

Precision and Recall Results on 612 Clean Text Documents

Per-Document	
tp: 108	fp: 16
fn: 5	tn: 483
precision: 87.1%	recall: 95.6%

Precision and Recall Results on 612 OCR Text Documents

Per-Document	
tp: 104	fp: 4
fn: 9	tn: 483
precision: 96.3%	recall: 92.0%

IE – Shallow *Precise* Parsing

- Focused on one piece of personal information
  - Is there a birthday mentioned here? On this line of text?
- Relation: birthday(name, date)
- Subproblem 1: find all the date patterns from examples
  - 1968-3-15
  - January 15, 1968
  - 3/15/1968
  - 3-15-1968
  - 4th of July, 1968
  - 15 September, 1968
  - 09-18-21
  - 09/18/21
  - l5-JUN-53
- Arrived at about 60 different date variations

IE –*Precise* Shallow Parsing

- Subproblem 2: find all the mentions of birthdates
- Look at tiny set of a dozen sample documents
- Preparing the test documents takes the team weeks
- In the meantime, look for extraction patterns on the web for extraction patterns too.
  - George Washington (born February 22,1732),
  - George Washington Tuesday, February 22;
  - George Washington's Birthday is February 22.
  - GEORGE WASHINGTON, b. February 22,
  - George Washington was actually born February 22, 1732
  - I learned about George Washington. He was born February 22, 1732.
  - George Washington ~Date of Birth~ February 22, 1732
  - George Washington, the first president of the United States, was born on February 22, 1732.

IE – *Precise* Shallow Parsing

- Trim context and abstract variables from particulars
  - aName was born on aDate
  - aName's Birthday aDate
  - aName (aDate)
  - aName, born on aDate
  - aName, (aDate-aDate2)
  - aName, born aDate
  - aName: aDate
  - aName's aDate birthday
  - aName was born aDate
  - On aDate, aName was born.
  - aName's Birthday (on aDate)
- Relax the requirement for matching personal name, match purely based on relative location to other parts of pattern

IE – *Precise* Shallow Parsing

Experimental Results (per document)

Precision and Recall Results on 1075 Clean Text Documents

Per-Document		Per-Hit	
tp: 67	fp: 3	tp: 215	fp: 32
fn: 9	tn: 997	fn: 91	tn:
precision: 95.71%	recall: 88.16%	precision: 87.04%	recall: 70.26%

Precision and Recall Results on 1075 OCRed Documents

Per-Document		Per-Hit	
tp: 64	fp: 3	tp: 180	fp: 1173
fn: 12	tn: 997	fn: 128	tn:
precision: 95.52%	recall: 84.12%	precision: 13.30%	recall: 58.44%

IE – *Fuzzy* Shallow Parsing

Some examples from clean text

- 1870 -1924
- (c. 581 - November, 644)
- (ca. 570/571 Mecca[مَكَهْ [/] مَكَةَ] – June 8, 632)

Some examples with OCR errors

- Lnuis Pasteur (I■822- I■895)
- (t9zg – 1953)
- t3lt4ltsl @a. 5701571 Mecca[eS. llli&'] - June 8,632

IE – *Fuzzy* Shallow Parsing

Some examples of a second kind with errors

- born: Jan 01, 1751
- he was born -on Feb. 11, 1947
- Born: August l6, 1769
- Mendel was bom in Hyncice, Moravia on22 Julyll22in what
- tsirth: c1400 in Mainz, Germany
- b, 18 November 1787; d. 10 JulY 1851

IE – *Fuzzy* Shallow Parsing

- Pattern 1, clean and OCR documents:
  - [0-9]+.{0,40}[-].{0,40}[0-9]+
- For pattern 2,
  - clean documents:
    - Pattern 2 a: [0-9]+ AND \b(birth|born)\b
    - Pattern 2 b: \b[.] [0-9]+
  - OCR documents:
    - Pattern 2 a: [0-9]+ AND \b(birth|born){-1+2#2~2}\b
    - Pattern 2 b: \b[.,] [0-9]+

IE – *Fuzzy* Shallow Parsing

Experimental Results (per hit)

Precision and Recall Results on 50 Clean Text Documents

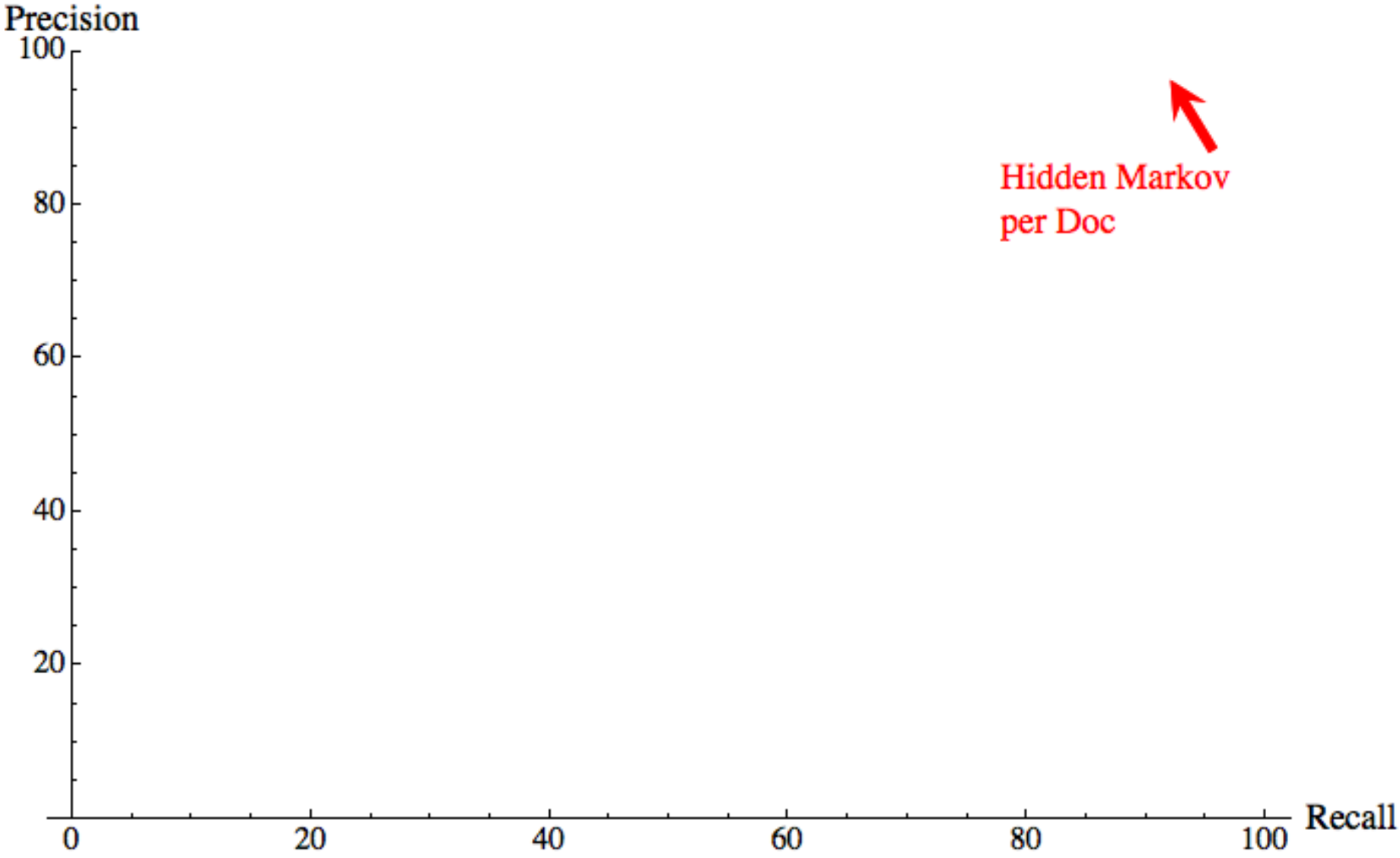
Pattern 1		Pattern 2	
tp: 37	fp: 16	tp: 41	fp: 1
fn: 9	tn: 10442	fn: 0	tn: 10543
precision: 69.81%	recall: 100.00%	precision: 97.62%	recall: 100.00%

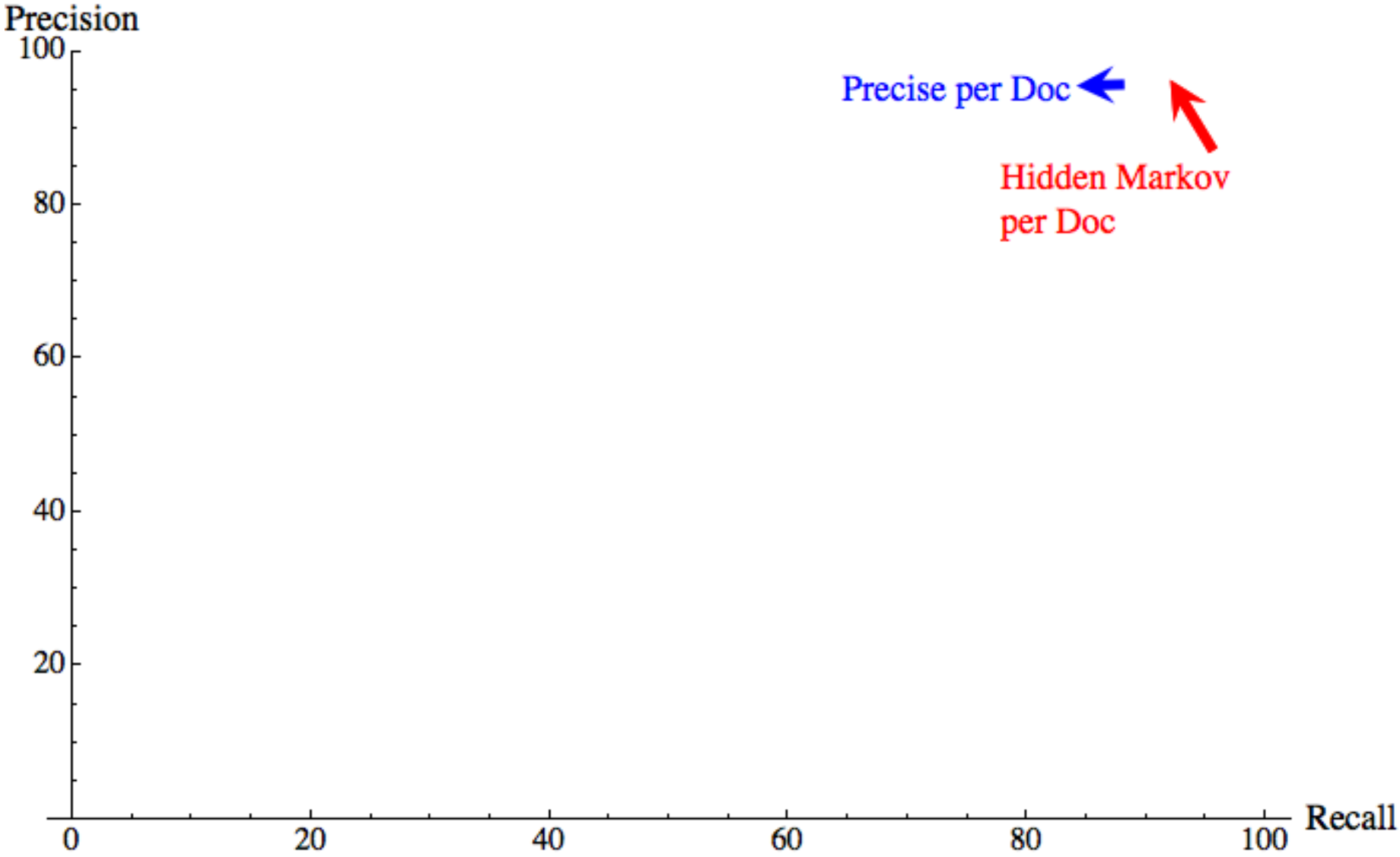
Precision and Recall Results on 50 OCRed Documents

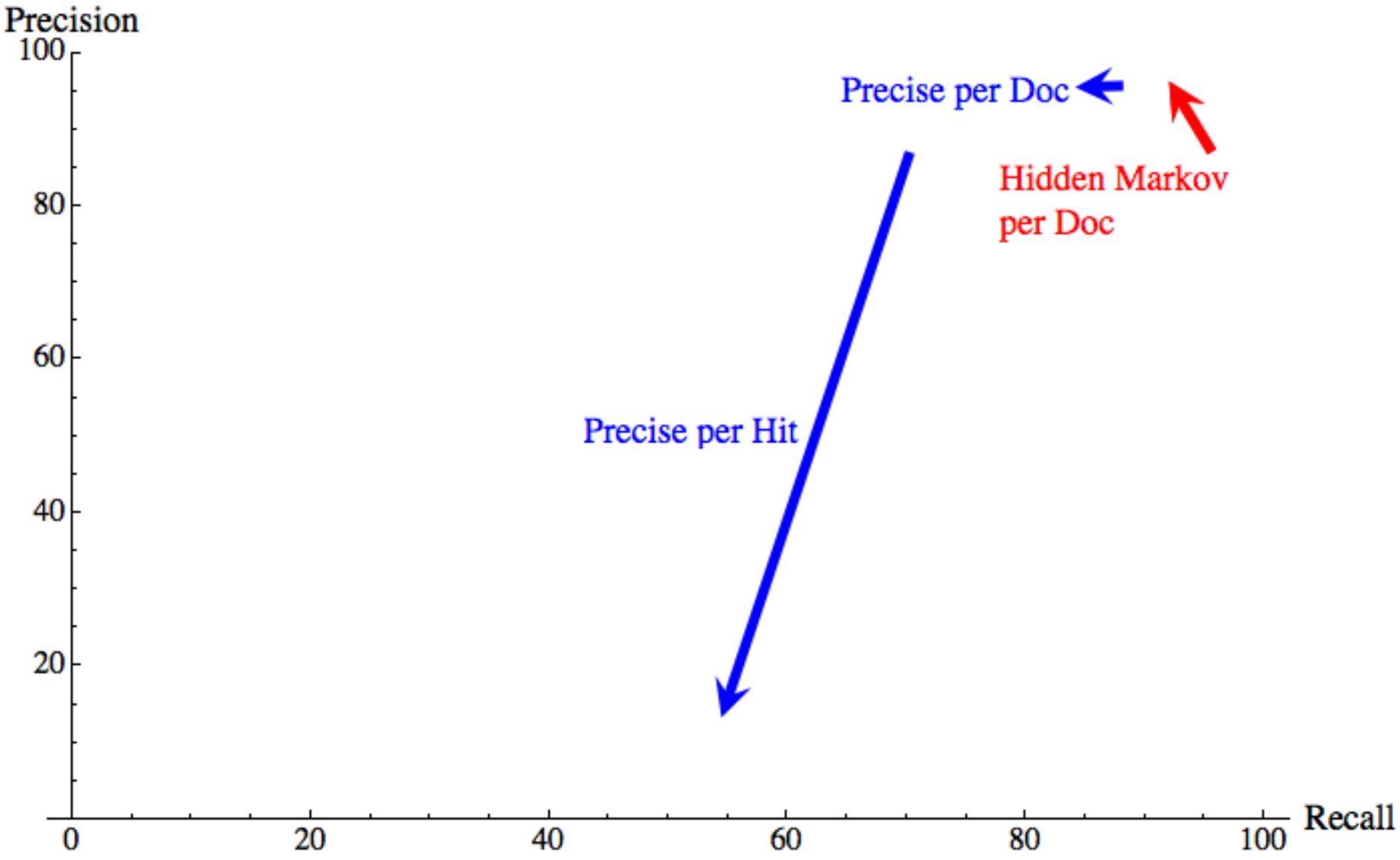
Pattern 1		Pattern 2	
tp: 35	fp: 29	tp: 36	fp: 109
fn: 0	tn: 3109	fn: 3	tn: 3025
precision: 54.69%	recall: 100.00%	precision: 24.83%	recall: 92.31%

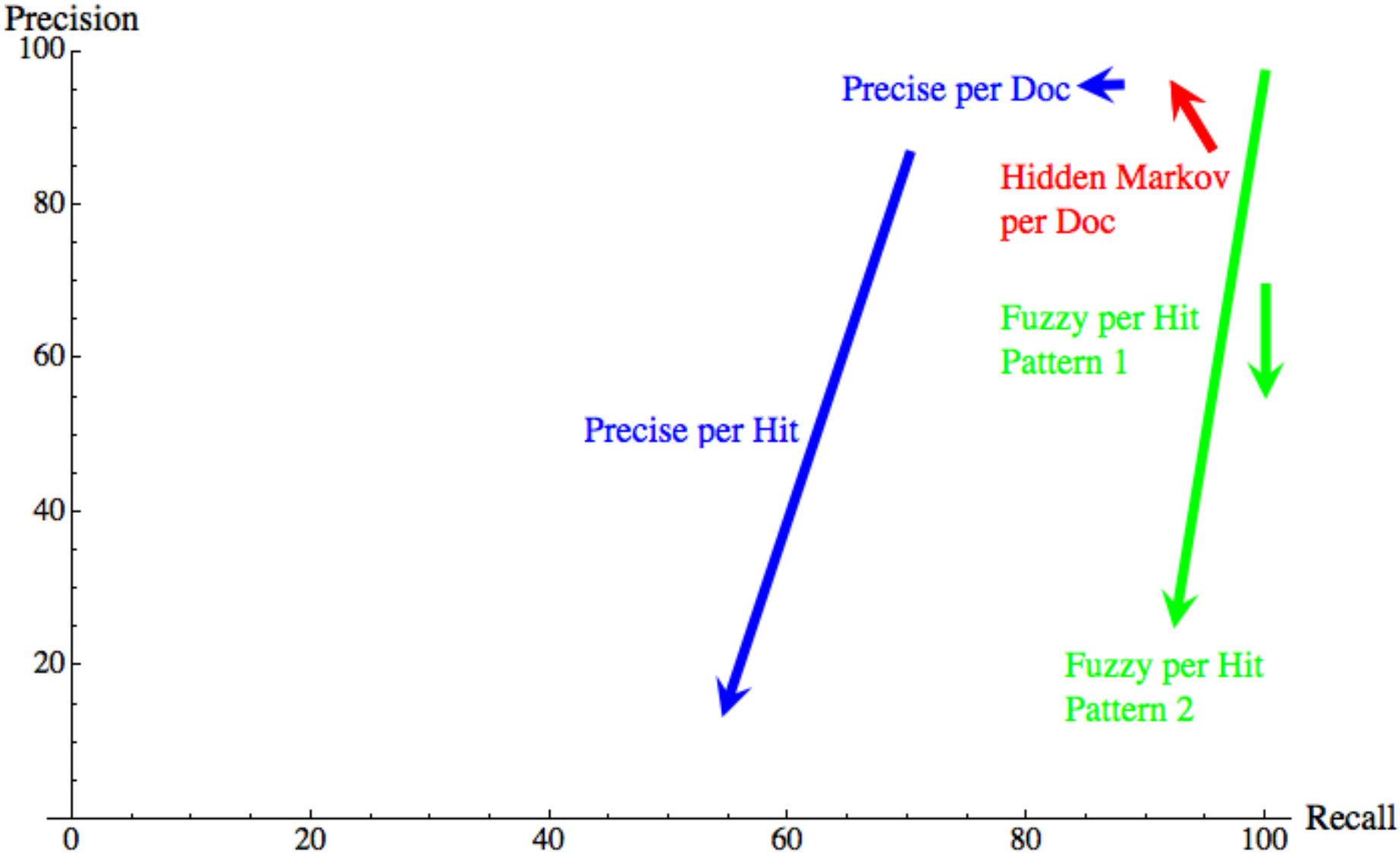
IE – Effects of OCR

- Three Models:
  - Hidden Markov
  - Precise Shallow Parsing
  - Fuzzy Shallow Parsing
- What is the precision & recall?
  - Before OCR errors, on clean text
  - After OCR errors, on dirty or OCRed text


Precision
100
80
60
40
20
0
20
40
60
80
100
Recall
Hidden Markov
per Doc


Precision
100
80
60
40
20
0
20
40
60
80
100
Recall
Precise per Doc
Hidden Markov
per Doc


Precision
100
80
60
40
20
0
20
40
60
80
100
Recall
Precise per Doc
Hidden Markov
per Doc
Precise per Hit


Precision
100
80
60
40
20
0
20
40
60
80
100
Recall
Precise per Doc
Hidden Markov
per Doc
Fuzzy per Hit
Pattern 1
Precise per Hit
Fuzzy per Hit
Pattern 2

Conclusion

IR Model	Significant	Not Significant
Boolean		✔
Probabilistic		✔
Vector Space		✔

IE Model	Significant	Not Significant
Hidden Markov	✔	
Shallow Precise	✔	
Shallow Fuzzy	✔	

- IR need not be specialized for OCR context
- IE needs to be specialized for OCR context

Conclusion

- Can IR be adapted for an OCR context?
  - Shallow Fuzzy Parsing: yes
- Can shallow fuzzy parsing techniques be generalized?
  - Probably Yes.
- First: Solve Real-World problems.
  - OCR is weird. Hard to guess the errors.
- Second: Formalize the steps
  - Pattern synthesis is subtle.
  - It becomes clear after much staring at mountains of data.

Questions

- Lots of big residual questions:
  - (1) How do you best measure OCR effects?
  - (2) Is there a preferred IE technique?
  - (3) How general-purpose can IE systems be?
  - (4) …

More question?
RayPereda@gmail.com