Introduction to

Informa l

Lecture 8: E

This lecture

- How do we know if our results are any good?
  - Evaluating a search engine
    - Benchmarks
    - Precision a

EVALUATING SEARCH ENGINES

Measures for a sea ine

- How fast does it index
  - Number of documents/hour
  - (Average document size)
- How fast doe
  - Latency as a function of inde
- Expressiveness of query lan
  - Ability to express complex information needs
  - Speed on complex queries
- Uncluttered UI
- Is it free?

Measures for a sea ine

- All of the preceding criteria are *measurable*: we can quantify speed/size
  - we can make expressiveness precise

- The key meas

  - What is this?
  - Speed of response/size of in rs

  - But blindingly fast, useless answers won’t make a user happy
- Need a way of quantifying user happiness

Measuring user ha

- Issue: who is the user we are trying to make happy?
  - Depends on the setting
- Web engine:

  - User finds w to the engine
    - Can measur

  - User completes their task – s eans, not end

  - See Russell http://dmrussell. s.com/JCDL-talk-June-2007-short.pdf
- eCommerce site: user finds what they want and buy
  - Is it the end-user, or the eCommerce site, whose happiness we measure?
  - Measure time to purchase, or fraction of searchers who become buyers?

Measuring user ha

- Enterprise (company/govt/academic): Care about "user productivity"
  - How much time do my users save when looking for information?
  - Many other breadth of access, secure access, etc.

Happiness: elusive sure

- Most common proxy: *relevance* of search results
- But how do you measure relevance?
- We will det e, then examine its issues
- Relevance measurement r lements:

1. A benchmark document collection
2. A benchmark suite of queries
3. A usually binary assessment of either Relevant or Nonrelevant for each query and each document
   - Some work on more-than-binary, but not the standard

Evaluating an IR sy

- Note: the **information need** is translated into a **query**
- Relevance is assessed relative to the **information need** *not* the

- E.g., Information need: *I'm information on whether drinking red wine i ective at reducing your risk of heart attacks than white wine.*
- Query: ***wine red white heart attack effective***
- You evaluate whether the doc addresses the information need, not whether it has these words

Standard relevance benchmarks

- TREC - National Institute of Standards and Technology (NIST) has run a large IR test bed for many years

- Reuters and other benchmark doc collections used

- “Retrieval tasks” specified

  - sometimes as queries
- Human experts mark, for each query and for each doc, Relevant or Nonrelevant
  - or at least for subset of docs that some system returned for that query

Unranked retrieval e[illegible]n: Precision and Recall

- **Precision**: fraction of retrieved docs that are relevant = P(relevant|retrieved)

- **Recall**: fraction of relevant docs that are retrieved = P(retrieved|r[illegible]

	Relevant	[illegible]elevant
Retrieved	tp	fp
Not Retrieved	fn	tn

- Precision P = tp/(tp + fp)
- Recall R = tp/(tp + fn)

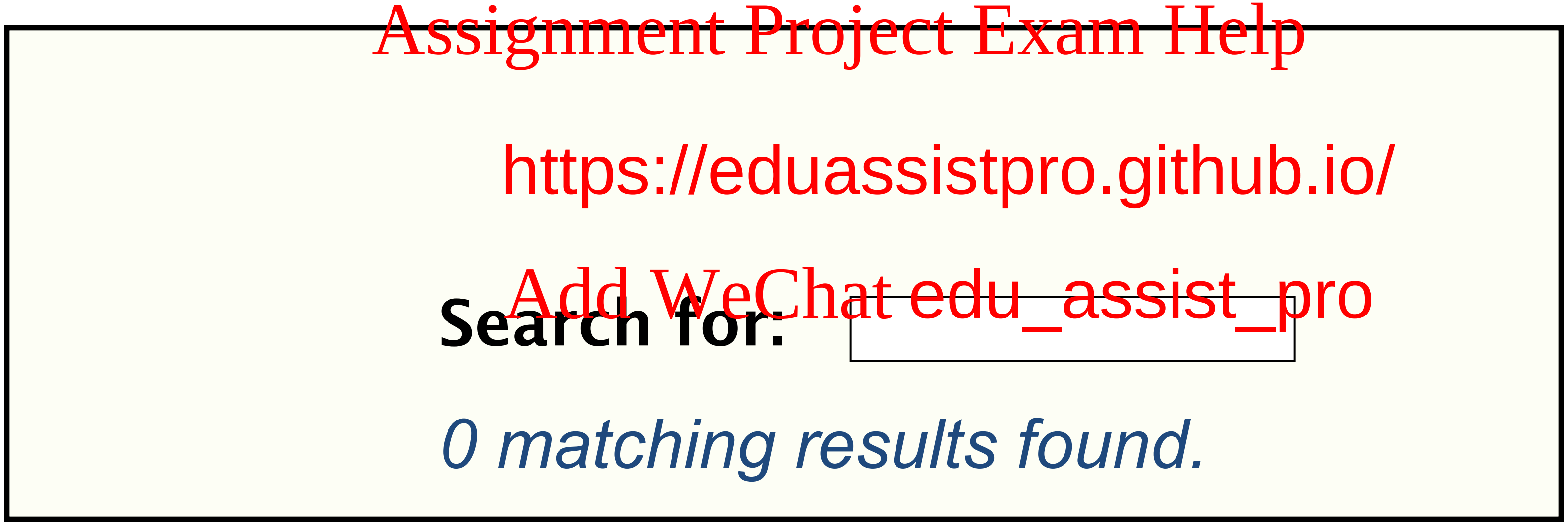
Should we instead us curacy measure for evaluati

- Given a query, an engine classifies each doc as “Relevant” or “Nonrelevant”
- The **accuracy** of an engine: the fraction of these classifications
  - (tp + tn) / (tp + fp + fn + tn)

- **Accuracy** is a commonly us on measure in machine learning classification work
- Why is this not a very useful evaluation measure in IR?

Why not just use a ?

- How to build a 99.9999% accurate search engine on a low budget….



Search for:

0 matching results found.

- People doing information retrieval *want to find something* and have a certain tolerance for junk.

Precision/Recall

- You can get high recall (but low precision) by retrieving all docs for all queries!

- Recall is a no of the number of docs retri

- In a good system, precision decreases as either the number of docs retrieved or recall increases
  - This is not a theorem, but a result with strong empirical confirmation

Difficulties in using on/recall

- Should average over large document collection/query ensembles
- Need human relevance assessments
  - People aren’

- Assessments have to be bi
  - Nuanced assessments?

- Heavily skewed by collection/authorship
  - Results may not translate from one domain to another

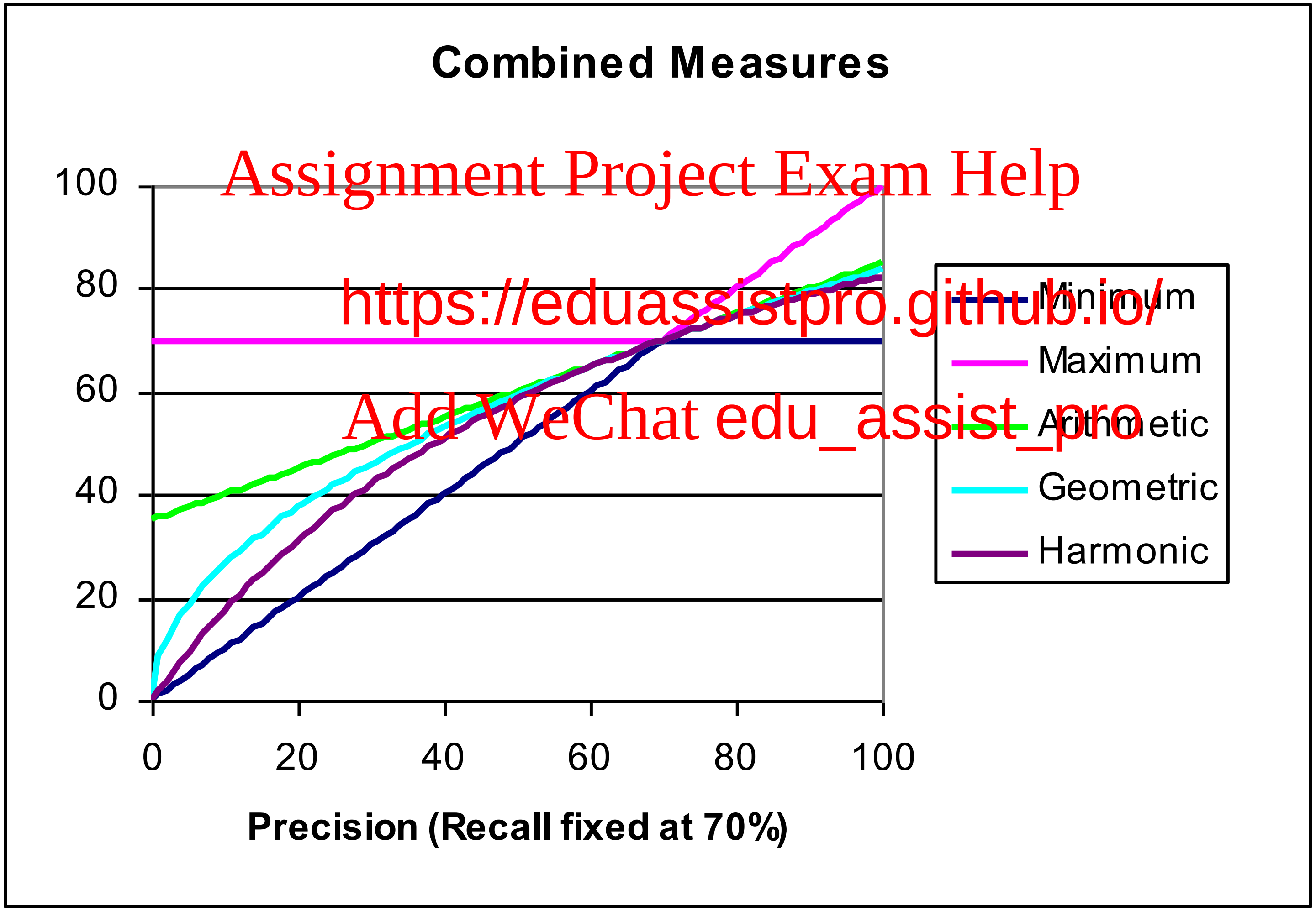
A combined measure

- Combined measure that assesses precision/recall tradeoff is **F measure** (weighted harmonic mean):

$$F = \frac{1}{\alpha\frac{1}{P} + (1-\alpha)\frac{1}{R}} = \frac{(\beta^2+1)PR}{\beta^2 P + R}$$

- People usually use balanced *F₁* measure
  - i.e., with β = 1 or α = ½
- Harmonic mean is a conservative average
  - See CJ van Rijsbergen, *Information Retrieval*

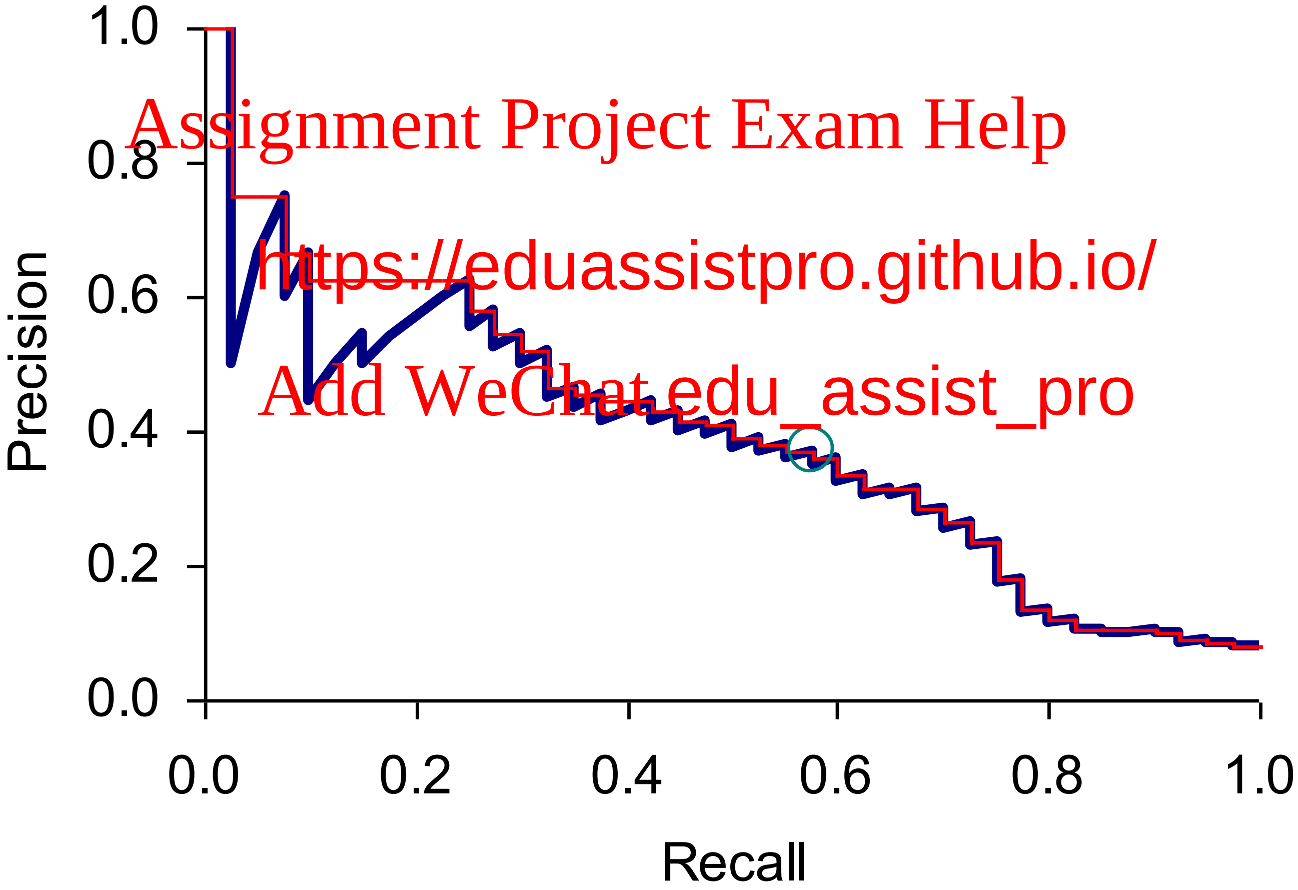
F_1 and other avera



Evaluating ranked

- Evaluation of ranked results:
  - The system can return any number of results
  - By taking various numbers of the top returned documents (levels of rec duce a *precision-recall curve*

A precision-recall c



Averaging over qu

- A precision-recall graph for one query isn't a very sensible thing to look at
- You need to average performance over a whole bunch of quer
- But there's a technical issu
  - Precision-recall calculations pl ints on the graph
  - How do you determine a value (interpolate) between the points?

Interpolated precis

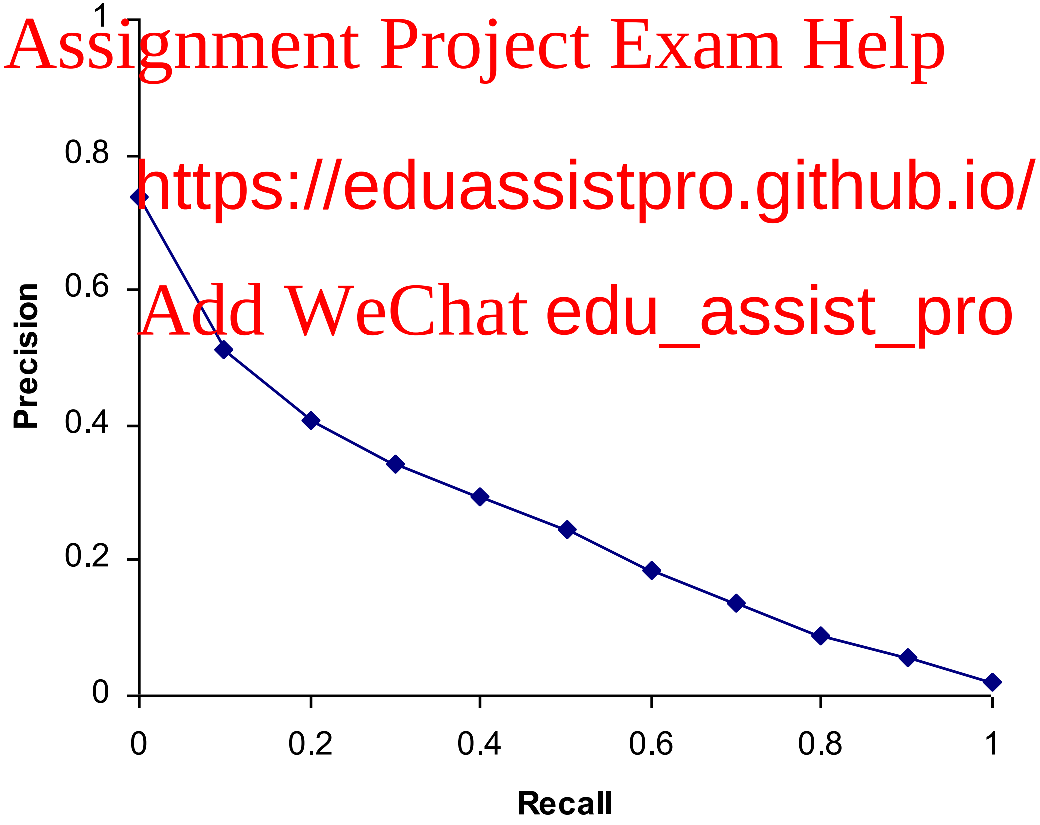
- Idea: If locally precision increases with increasing recall, then you should get to count that…
- So you max of precisions to right of value

Evaluation

- Graphs are good, but people want summary measures!
  - Precision at fixed retrieval level
    - Precision-at-*k*: Precision of top *k* results
    - Perhaps arch: all people want are good mat sults pages
    - But: averages badly and has arameter of *k*
  - 11-point interpolated aver n
    - The standard measure in the early TREC competitions: you take the precision at 11 levels of recall varying from 0 to 1 by tenths of the documents, using interpolation (the value for 0 is always interpolated!), and average them
    - Evaluates performance at all recall levels

Typical (good) 11 point precisions

- SabIR/Cornell 8A1 11pt precision from TREC 8 (1999)



Yet more evaluati ures…

- Mean average precision (MAP)
  - Average of the precision value obtained for the top *k* documents, each time a relevant doc is retrieved
  - Avoids inter ll levels
  - MAP for que ave.
    - Macro-averaging: each query c

- R-precision
  - If have known (though perhaps incomplete) set of relevant documents of size *Rel,* then calculate precision of top *Rel* docs returned
  - Perfect system could score 1.0.

Variance

- For a test collection, it is usual that a system does crummily on some information needs (e.g., MAP = 0.1) and excellently on others (e.g., MAP = 0.7)

- Indeed, it is u e variance in performance ross queries is much greater than the vari ferent systems on the same query.

- That is, there are easy information needs and hard ones!

CREATING TEST COLLECTIONS FOR IR EVALUATION

Test Collections

From document collections to test collections

- Still need
  - Test queries
  - Relevance assessments
- Test queries
  - Must be germane to docs av
  - Best designed by domain ex
  - Random query terms generally not a good idea
- Relevance assessments
  - Human judges, time-consuming
  - Are human panels perfect?

Unit of Evaluation

- We can compute precision, recall, F, and ROC curve for different units.
- Possible units
  - Documents (
  - Facts (used in some TREC evalua
  - Entities (e.g., car companies)
- May produce different results. Why?

Kappa measure for inter-judge (dis)agreement

- Kappa measure
  - Agreement measure among judges
  - Designed fo
  - Corrects for
- Kappa = [P(A) – P(E)] / [1 – P
- P(A) – proportion of time judg
- P(E) – what agreement would be by chance
- Kappa = 0 for chance agreement, 1 for total agreement.

Kappa Measure: E

P(A)? P(E)?

	Judge 2: Relevant	Judge 2: Nonrelevant
Judge 1: Relevant	300	20
Judge 1: Nonrelevant	10	70

Total assessment:400

- P(A) = 370/400 = 0.9250
- P(nonrelevant) = (10+20+70+70)/800 = 0.2125
- P(relevant) = (10+20+300+300)/800 = 0.7875
- P(E) = 0.2125^2 + 0.7875^2 = 0.6653
- Kappa = (0.9250 – 0.6653)/(1-0.6653) = 0.7759

Using pooled marginals

Kappa Example

- P(A) = 370/400 = 0.9250
- P(nonrelevant) = (10+20+70+70)/800 = 0.2125
- P(relevant) = (7875
- P(E) = 0.2125^
- Kappa = (0.9250 – 0.6653)/(1- 759

- Kappa > 0.8 = good agreement
- 0.67 < Kappa < 0.8 -> “tentative conclusions” (Carletta ’96)
- Depends on purpose of study
- For >2 judges: average pairwise kappas

TREC

- TREC Ad Hoc task from first 8 TRECs is standard IR task
  - 50 detailed information needs a year
  - Human evaluation of pooled results returned
  - More recently , HARD
- A TREC query (TR

<top>

<num> Number: 225

<desc> Description:

What is the main function of the Federal Emergency Management Agency (FEMA) and the funding level provided to meet emergencies? Also, what resources are available to FEMA such as people, equipment, facilities?

</top>

Standard relevance benchmarks: Others

- GOV2
  - Another TREC/NIST collection
  - 25 million web pages
  - Largest collection that is easily available
  - But still 3 ord n what Google/Yahoo
- NTCIR
  - East Asian language and cross-language information retrieval
- Cross Language Evaluation Forum (CLEF)
  - This evaluation series has concentrated on European languages and cross-language information retrieval.
- Many others

Interjudge Agreem EC 3

Impact of Inter-jud ement

- Impact on absolute performance measure can be significant (0.32 vs 0.39).
- Little impact o tems or relative performance
- Suppose we want to know if a s better than algorithm B
- A standard information retrieval experiment will give us a reliable answer to this question.

Critique of pure rel

- Relevance vs Marginal Relevance
  - A document can be redundant even if it is highly relevant
  - Duplicates
  - The same inf ources
  - Marginal rel e of utility for the user.

- Using facts/entities as evaluation units more directly measures true relevance.
- But harder to create evaluation set

$$MMR \stackrel{\mathrm{def}}{=} Arg \max_{D_i \in R \setminus S} \left[\lambda (Sim_1(D_i, Q) - (1-\lambda) \max_{D_j \in S} Sim_2(D_i, D_j)) \right]$$

Can we avoid hum ment?

- No
- Makes experimental work hard
  - Especially on a large scale
- In some very se proxies
  - E.g.: for approximate vector val, we can compare the cosine distance c he closest docs to those found by an approximate retrieval algorithm
- But once we have test collections, we can reuse them (so long as we don't overtrain too badly)

Evaluation at large [illegible] engines

- Search engines have test collections of queries and hand-ranked results
- Recall is difficult to measure on the web
- Search engines of [illegible] .g., k = 10
- . . . or measures t[illegible] ting rank 1 right than for getting rank 10 right.
    - **NDCG** (Normalized Cumulative Disc[illegible]
- Search engines also use non-relevance-based measures.
    - Clickthrough on first result
        - Not very reliable if you look at a single clickthrough … but pretty reliable in the aggregate.
    - Studies of user behavior in the lab
    - A/B testing

A/B testing

- Purpose: Test a single innovation
- Prerequisite: You have a large search engine up and running.
- Have most users use old system
- Divert a small proportion of traffic (e.g., 1%) to the new system that inclu
- Evaluate with an kthrough on first result
- Now we can directly see if the i es improve user happiness.
- Probably the evaluation methodology that large search engines trust most
- In principle less powerful than doing a multivariate regression analysis, but easier to understand

RESULTS PRESENTATION

Result Summaries

- Having ranked the documents matching a query, we wish to present a results list
- Most commonly, a list of the document titles plus a short summary

Resources for this l

- IIR 8
- MIR Chapter 3
- MG 4.5
- Carbonell and se of MMR, diversity-based reranking f ing documents and producing summaries.