Text Retrieval

- Introduction
- Text Retrieval Tasks
- Evaluation
- The Vector Space Model
- Advanced Techniques

Readings

- Manning Chapter: Text Retrieval (Selections)
- Vorhees & Harman (Bulkpack)

Text Retrieval: Overview

- Goal:
  - Use a collection of documents to generate a response to a query.
- Prototypical example:
  - Web search engine

Text Retrieval Tasks

- Ad-hoc retrieval
  - Given: Large fixed document set,
  - Goal: Find documents that answer queries.
- Filtering
  - Given: Fixed query
  - Goal: Decide whether documents answer that query.
- Question-Answering
  - Given: Large fixed document set
  - Goal: Find short passages (~ 1 sentence) that answer questions.

Text Retrieval Tasks: Classification

There are many different kinds of text retrieval.

- **Variables for Classification:**
  - **Is the query known ahead of time?**
  - **Is the corpus known ahead of time?**
  - **What kind of query is used?**
  - **What kind of corpus is used?**
  - **What kind of response should be generated?**

Text Retrieval Tasks: Classification (Cont'd)

- **Example classifications:**

Task	Known ahead of time?	Corpus	Query	Response
Ad-Hoc	*Corpus*		*Short – Medium*	*Document*
Question Answering	*Corpus*		*Short*	*Sentence*
Filtering	*Query*		*Long*	*Document*
Cross-Language	*Corpus*	*Multilingual*	*Short*	*Document*
Speech	*Corpus*	*Speech*	*Short – Medium*	*Document*

TREC

- **Text REtrieval Conference**
  - **Annual workshop to foster reasearch in text retrieval**
  - **50+ groups compete for high performance.**
- **16 *Tracks* = types of text retrieval**
  - **7 tracks were active in TREC-9**

Ad-Hoc	Database Merging	High Precision
Routing	Filtering	Very Large Corpus
Interactive	Chinese	Query
Spanish	NLP	Cross-Language
Confusion	Speech	Web

Evaluation

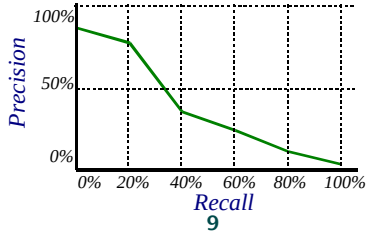
- **Precision/Recall: Review**

	Target	¬Target
Selected	True positive	False positive
¬Selected	False negative	True negative

- **Precision** = $\frac{tp}{(tp+fp)}$
  - **What proportion of selected items are correct?**
- **Recall** = $\frac{tp}{(tp+fn)}$
  - **What proportion of target items are selected?**

Evaluation: Precision/Recall Graphs

- Text retrieval generates multiple responses
- Consider the first *n* responses for various *n*
  - *n*=0: precision=100%; recall=0%
  - n=N: precision=0%; recall=100%
- Graph precision vs. recall at various *cutoffs*



Evaluation: Summary Measures

- Uninterpolated average precision
  - *P* = positions at which we got a true positive
  - average precision = $\underset{p \in P}{\text{Avg}}$(precision at cutoff *p*)
- F Measure
  - Weighted harmonic mean of precision and recall.

$$\frac{1}{\alpha \frac{1}{P} + (1-\alpha)\frac{1}{R}}$$

Evaluation: Test Set Construction

- Which documents are relevant?
  - Need human evaluation
  - New evaluation for each query
- Need a *complete set* of relevant documents
  - Can't find recall without a complete set
- But there are too many documents to evaluate!

Evaluation: Pooling

- TREC: Construct a test set that is "approximately correct"
  - Assume that all releveant documents are returned by at least one text retrieval system
- Manually evaluate the top 100 documents returned by each system.
- No bias against any competing system.
- Small bias against systems that did not compete.

The Vector Space Model

Represent each document as a sparse vector $\vec{x}$

- **Each dimension corresponds to a single vocabulary item:**
  - **1 if the vocabulary item is in the document**
  - **0 if the vocabulary item is not in the document**

$d = document\,(multiset\,of\,words)$

$V = w_1, w_2, \ldots, w_n = vocabulary$

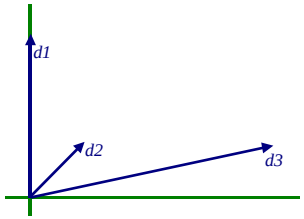
$w_i = word$

$\vec{x} = \langle x_1, x_2, \ldots, x_n \rangle$

$x_i = \begin{Bmatrix} 1 : if\ w_i \in d \\ 0 : if\ w_i \notin d \end{Bmatrix}$

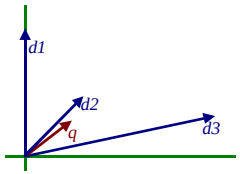
Comparing Documents

- **Two documents are similar if they have similar term vectors.**



Comparing Documents & Queries

- **Treat the query as a very small document.**
- **Construct a vector representation of the query.**



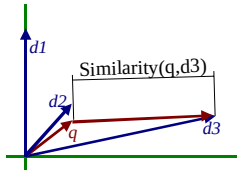
- **Query vectors are typically shorter than document vectors.**

Comparing Term Vectors

What makes term vectors similar?

- **Attempt 1: Term vectors are similar if their difference is small.**

$similarity_1(\vec{x}_1, \vec{x}_2) = |\vec{x}_1 - \vec{x}_2|$



- **Does document length matter?**
- **We only care about the *relative frequency* of each term.**

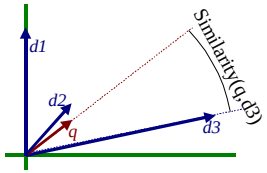
Comparing Term Vectors Cont'd: The Cosine Measure

What makes term vectors similar?

- **Attempt 2: Term vectors are similar if the angle between them is small.**

$$similarity_2(\vec{x}_1, \vec{x}_2) = \cos(\vec{x}_1, \vec{x}_2)$$

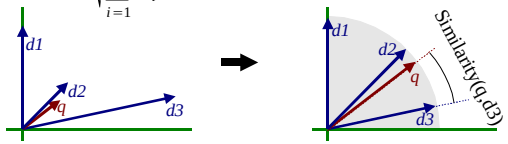
$$\cos(\vec{x}_1, \vec{x}_2) = \frac{\vec{x}_1 \cdot \vec{x}_2}{|x_1| \times |x_2|}$$



Normalizing Term Vectors

- **Cosines are easier to compute if we first *normalize* all document vectors.**

$$\vec{x}' = \frac{\vec{x}}{|\vec{x}|} = \frac{\vec{x}}{\sqrt{\sum_{i=1}^{n} x_i^2}}$$



$$similarity_2(\vec{x}_1, \vec{x}_2) = \vec{x}_1' \cdot \vec{x}_2'$$

Term Weighting

- **Some terms are more informative than others.**
- **Can we get a better simularity measure if we use different vectors?**

$$\vec{x} = \langle x_1, x_2, \ldots, x_n \rangle$$

- **Change our definition of x_i**

$$x_i = \begin{Bmatrix} 1 : if\ w_i \in d \\ 0 : if\ w_i \notin d \end{Bmatrix}$$

- **Increase x_i for more informative terms.**

Term Weighting (Continued) Term Frequency

- **Term frequency:** $tf_i = number\ of\ occurances\ of\ w_i \in d$
- **Higher term frequency ➔ term is more relevant to the document.**
- **Weight a term proportionally to its term frequency?** $x_i = tf_i$
- **But a word appearing 3 times as often is not 3 times as relevant: use smoothing.**

$$x_i = \begin{Bmatrix} 1 + \log(tf_i) : if\ w_i \in d \\ 0 : if\ w_i \notin d \end{Bmatrix} \quad \textbf{or} \quad x_i = \sqrt{tf_i}$$

Term Weighting (Continued) Inverse Document Frequency

- Document Frequency:

$$df_i = number\ of\ documents\ containing\ w_i$$

- Higher document frequency ➡ term is less informative.
- Weight a term inversely to its document frequency.

$$x_i = \begin{Bmatrix} \left(1+\log(tf_i)\right)\log\left(\frac{N}{df_i}\right) : if\ w_i \in d \\ 0 : if\ w_i \notin d \end{Bmatrix}$$

- (use smoothing)

Latent Semantic Indexing

- Vector model assumes that term contributions to document similarity are *independant*.
- Many terms are not independant.
  - "star chart" is similar to "astral map"
- Project document vectors into a new vector space.
  - In the new vector space, terms that are semantically similar are close to each other.
  - "Latent" semantic dimensions

Using NLP Techniques

- Traditional wisdom: linguistic knowledge does not improve text retrieval performance.
  - Especially true for ad-hoc retrieval.
- But:
  - Linguistic knowledge useful for other tasks (e.g., question answering, cross-language text retrieval)
  - Clearly a *temporary* fact: it is impossible to get 100% performance without linguistic knowledge.

Using NLP Techniques: Words

- Add new terms:
  - Find terms with collocation analysis
  - Synonym expansion: WordNet
- Make terms more precise:
  - Word sense disambiguation
- Merge terms with the same meaning:
  - Stemming
- Clean up a "messy" corpus:
  - Language modelling
  - Spell-check

Using NLP Techniques: Syntax

- **Find new terms:**
  - Entity extraction
  - Relationship extraction
- **Narrow the search space:**
  - Question parsing
    - Determine what type of answer we expect
  - Entity tagging
    - Tag entities with types (e.g., company, person, date)
  - Relationship extraction
    - Search for partial relationships (e.g., "John likes *?x*")