Statistical Natural Language Processing

The Problem

- Humans learn to read, write, speak and understand - with very little training data
- Language is highly ambiguous
  - For example - 455 legal parses for "List the sales of the products produced in 1973 with the products produced in 1972"
  - "Police can't stop drinking"

Possible explanations

- Rationalist
  - Innate knowledge (grammars)
  - Categorical (sentences are either grammatical or non-grammatical)
  - Flaw: grammars 'leak' (language changes, usage is imperfect, etc.)
  - Response: Competence vs. Performance
- Empiricist
  - General operations
    - association, pattern recognition, generalization
  - Probabilistic
  - Models trained on text

Subproblems of (Statistical) NLP

- Part-of-speech (POS) tagging
- Parsing
- Semantic Analysis (pragmatics, discourse)
- Word sense Disambiguation
- Speech synthesis/recognition
- Information Retrieval
- Information Extraction

Word Sense Disambiguation (WSD)

She gave him the slip, then slipped onto the slip, but slipped on a banana and ripped her slip.

- Determine which of several senses of a word is intended, given its context.
- Inputs
  - Target word and senses
  - Context (window of text surrounding target)
- Output: Correct sense

Applications of WSD

- Natural language understanding
  - For interfaces, message understanding, etc.
- Machine translation
  - "Sentence" = *"peine"* or *"phrase"*
- Information retrieval
  - Legal query: eliminate "[royal] court" docs
- Speech synthesis
  - I caught a bass as big as my bass guitar.

Unsupervised Learning for WSD

- David Yarowsky - 1995
- Leverage a small training set using bootstrapping
- Model used is *decision list*

Decision list

- Ordered list of rules
- Score for each rule (LogLikelihood)
- To apply, step through rules in order till a precondition matches

Condition	Prediction	LogL
plant **life**	A (biological)	8.10
assembly *plant*	B (factory)	7.58
animal within +/- 2-10 words	A	6.27

Algorithm

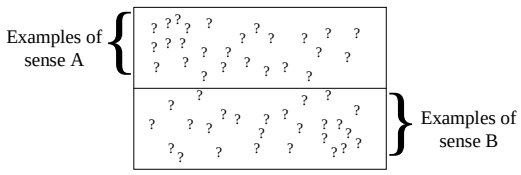
Extract *examples* (target word and context)
Label *training set* of a few examples per sense
Loop
  Learn *decision list* with *training set*
  Classify all *examples* using *decision list*
  Add high-probability examples to *training set*
  Remaining examples are *residual set*
Until *residual set* stabilizes
Return *decision list*

"One sense per discourse"

- Typically, a single sense of a word is used in a given discourse [Gale et al., 1992]
- Use this observation to:
  - Correct misclassifications
  - Add new classifications (bridging)
- Improve accuracy and coverage

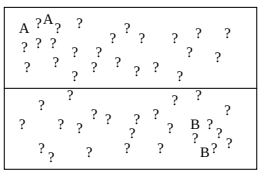
Example

- Initial state



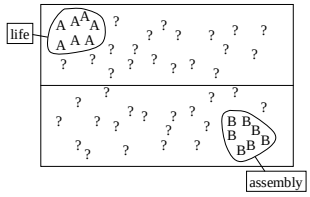
Initialization

- Hand label a few seed examples



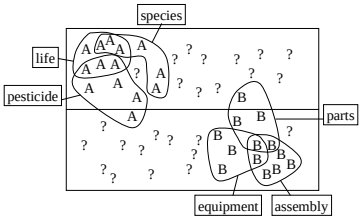
Iteration 1

- Learn decision list with a few rules for prediction with high probability

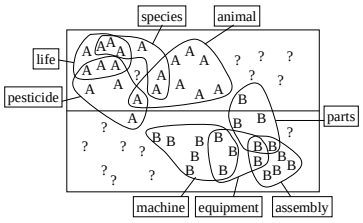


Iteration 2

- Find new rules, label examples

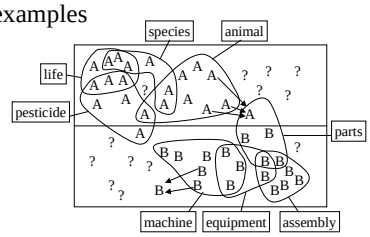


Further iteration



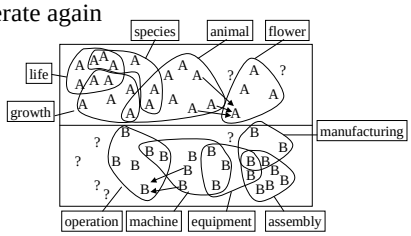
One sense per discourse

- Classify new examples, reclassify incorrect examples



Example

- Apply one-sense-per-discourse heuristic and iterate again



Information Retrieval

- Input
  - Set of documents to be queried
  - Specification of *information need*
- Output
  - (Sub)Set of documents ranked by *relevance*
- Example: web queries

Vector space model

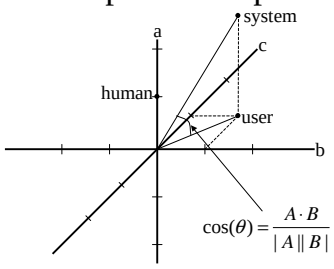
- Represent corpus as a matrix [Salton et. al]
  - Words are rows
  - Documents are columns
  - Cell *i,j* contains the number of times word *i* appears in document *j*
  - Similarity between two words is the cosine of the angle between the vectors representing those words

Vector space example

a: System and human system engineering testing of EPS
b: A survey of user opinion of computer system response time
c: The EPS user interface management system
d: Human machine interface for ABC computer applications
e: Relation of user perceived response time to error measurement
f: The generation of random, binary, ordered trees
g: The intersection graph of paths in trees
h: Graph minors IV: Widths of trees and well-quasi-ordering
i: Graph minors: A survey

	a	b	c	d	e	f	g	h	i
Human	1	0	0	1	0	0	0	0	0
User	0	1	1	1	0	0	0	0	0
System	2	1	1	1	0	0	0	0	0
Interface	0	0	1	0	1	0	0	0	0
Computer	0	1	0	0	0	0	0	0	0
Response	0	1	0	0	1	0	0	0	0
Time	0	1	0	0	1	0	0	0	0
EPS	1	0	1	0	0	0	0	0	0
Survey	0	1	0	0	0	0	0	0	1
Trees	0	0	0	0	0	1	1	1	0
Graph	0	0	0	0	0	0	1	1	1
Minors	0	0	0	0	0	0	0	1	1

Vector space example cont.



Using Co-occurrence Information

- Major problems with word-matching
  - Synonymy (one meaning, many words)
  - Polysemy (one word, many meanings)
- Solutions
  - Concept search
  - Query expansion
  - Clustering

almost ⇨ Latent Semantic Indexing

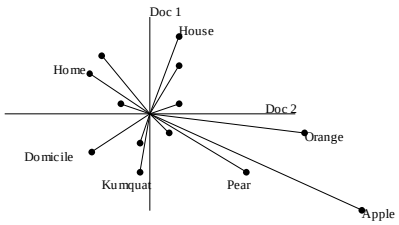
Latent Semantic Indexing is ...

- Latent
  - Captures underlying structure of corpus
- Semantic
  - Groups words by "conceptual" similarity
- Cool
  - Lots of neat applications
- Not Silver Bullet
  - Not really semantic, just MDS, expensive

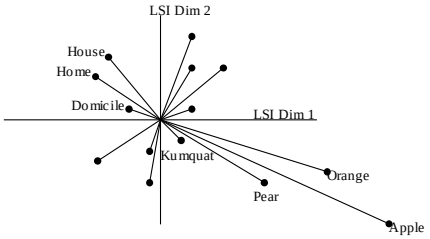
What does LSI do

- Restructures vector space so that co-occurrences are mapped together
- Captures transitive co-occurrence relations
- Application of dimensional reduction to term-document matrix
  - Throw out da noise, bring in da regularities

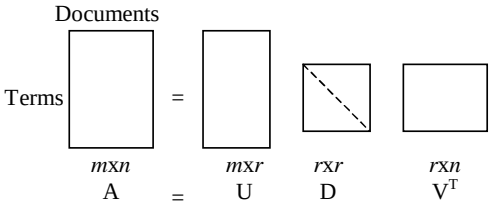
Document vector space



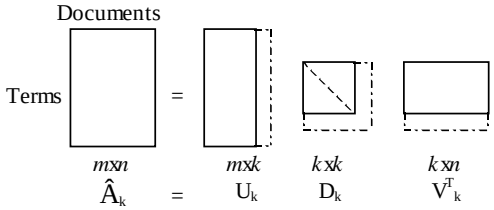
Semantic Space



Singular Value Decomposition



Term-Document Matrix Approximation



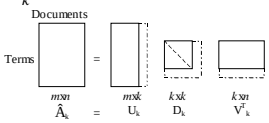
Properties of Â

- Best least-squares approximation of A given only k dimensions
- Terms and documents which were similar in A are more similar in Â
- This measure of similarity is transitive

So what can we do with this?

LSI Tricks and Tips

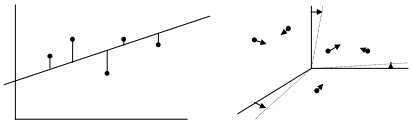
- Use $\hat{A}$ to query using standard cosine measure
- Use $U_k \cdot D_k$ for term similarity
- Use $D_k \cdot V_k^T$ for document similarity



Least Squares Approximation

Why does it work? 1st Attempt

- Â is best least-squares approximation to A using just k dimensions



Least Squares cont.

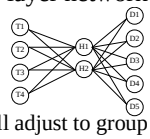
- Why does this work
- Are these the regularities we want to capture
- Why approximate at all? (hint: overfitting)

Not very convincing

Neural Network Explanation

Why does it work? 2nd Attempt

- Consider fully connected 3 layer network
  - First layer is terms
  - Middle layer has *k* units
  - Last layer is documents
  - Weights on hidden layer will adjust to group terms that appear in similar documents and documents containing similar terms
  - This is analogous to the SVD matrices



Linear Algebra Explanation

Why does it work? 3rd Attempt

- Rank of a matrix is a measure of how much information it contains
- Rows which are linear combinations of each other can be removed
- In this case, some singular values will be 0

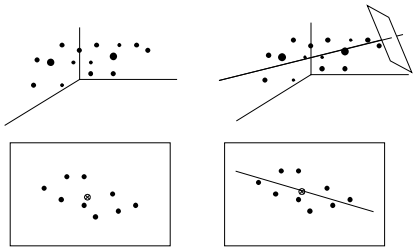
Eigenvalues cont.

- Consider vectors of terms X, Y and Z
  - X = [1 1 0 0 1 0 ...]
  - Y = [0 0 1 1 0 0 ...]
  - Z = [1 1 2 2 0 1 ...]
  - $Z \approx X + 2Y$
  - Some singular value of A is low
  - By forcing that singular value to 0, we merge X, Y and Z

Dimensionality Reduction

- Dimensionality reduction takes high-dimensional data and re-expresses it in a lower dimension - avoids overfitting
- Principle Components Analysis
  - If you were only allowed 1 line to represent all the data, what would it be
    - The one that explains the greatest variance
  - Recur

PCA cont.



Applications

- Information Retrieval
  - Improve retrieval
  - Cross-language retrieval
  - Document routing/filtering
  - Measuring text coherence
- Cognitive Science
  - Learning synonyms
  - Subject matter knowledge
  - Word sorting behavior
  - Lexical priming
- Education
  - Essay grading
  - Text selection

Standard Vector Space Retrieval in LSI Space

- Improves recall at expense of precision
- Compared to term-document vector space, SMART and Vorhees [Deerwester et al. 1990]
  - LSI did best on MED dataset
  - SMART did best on CISI dataset
  - but LSI was comparable to SMART when stemming was added

Cross Language Retrieval

- Train on multilingual corpora using "combined" documents
- Add in single language documents
- Query in LSI space

  [Landauer & Littman 1990] French & English

  [Landauer, Littman & Stornetta 1992] Japanese & English

  [Young 1994] Greek & English

  [Dumais, Landauer & Littman 1996] Comparisons between LSI, no-LSI and Machine Translation

Document Routing/Filtering

- Match reviewers with papers to be reviewed based on reviewers' publications [Dumais & Nielsen 1992]
- Select papers for researchers to read based on other papers they liked [Foltz & Dumais 1992]

LSI goes to college

- Train LSI on encyclopedia articles
- Test against TOEFL synonym test
- Results comparable to (non-native) college applicants

[Landauer & Dumais 1996]

- Train on introductory Psychology texts
- Receive passing grade on multiple-choice questions (but did worse than students)

[Landauer, Foltz & Laham 1998]

Essay Grading

- Several techniques
  - Use essay (or sentences from essay) to query into textbook or database of graded essays
  - Grade based on cosine from text or closest graded essay
  - More consistent than expert human graders [Landauer, Laham & Foltz 1998]
  - Is that good?

Routing Meets Education

- Run LSA on a bunch of texts at different levels of sophistication
- Have student write short essay about topic
- Use essay as query to select most appropriate text for student

[Wolfe, Schreiner, Rehder, Laham Foltz, Kintch and Landauer 1998]

Measuring Text Coherence

- Use LSI to compute cosine of each sentence with following one [Foltz, Kintch & Landauer 1998]
- Correlates highly with established methods
- Can indicate where coherence breaks down
- Can be used to measure how semantic content changes across a text (discourse segmentation?)