CS 276a Problem Set #2

Assigned: Tuesday, November 9, 2004

Due: Thursday, November 18, 2004 by 5:30 p.m.

Review session: Friday, November 12, 2004, 3:15-4:05 p.m. in Gates B01

Delivery: Local students should hand their solutions to Dan or Louis in class or leave them under Professor Manning's door (158). SCPD students should use the SCPD courier system.

Late policy: Refer to the course webpage.

Honor code: Please review the collaboration and honor code policy on the course webpage. Also note: because some questions may be drawn from previous years' problem sets or exams, students are forbidden to consult solution sets from previous years unless we explicitly provide them.

7 questions and 90 points total.

#1. Probabilistic models (15 points)

We are given a corpus consisting of the following two documents. We calculate probabilities based on the corpus as a whole.

"the martian has landed"
"the latin pop sensation ricky martin"

a. Under a unigram probability model, what are P("the") and P("martian")?

b. Under a bigram model what are P("sensation"|"pop") and P("pop"|"the")?

c. What are P("ricky martian") and P("ricky martin") under a unigram probability model? bigram?

d. Consider P("pop martian") and P("pop martin"). Which should be higher? Does a unigram model agree with this judgment? What about a bigram model? If neither one agrees, suggest another probability model that would.

e. What is unigram P("the red martian has landed")? If this is unreasonable, suggest a way to fix it.

f. How can we use the language model to "correct" user queries (à la Google's "Did you mean…") even when none of the words are misspelled? For example, we might want

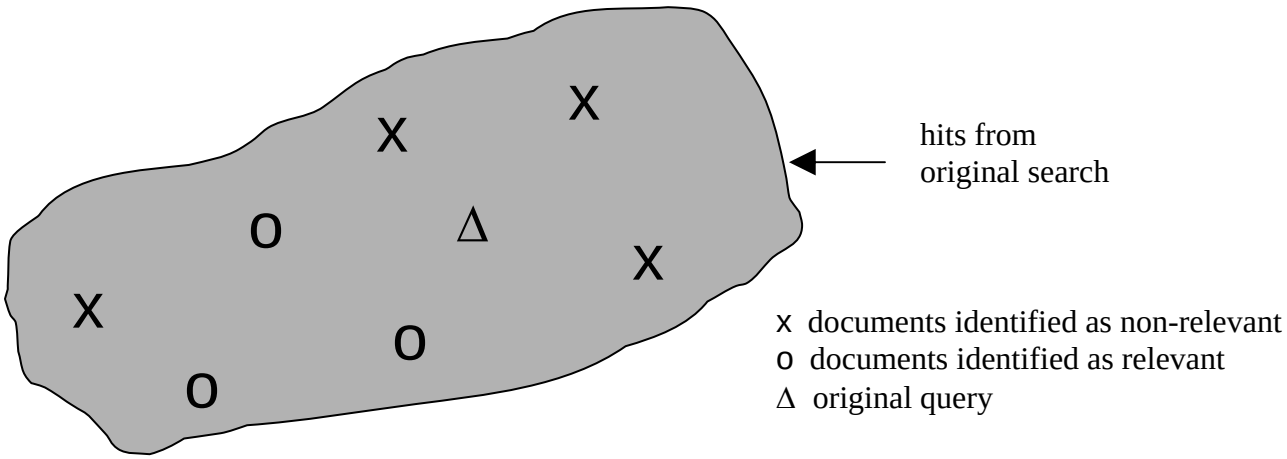
"ricky martian" corrected to "ricky martin". Efficiency does not matter. (Hint: use the lexicon.)

#2. Query expansion (10 points)

Compare the following two techniques for query expansion: synonym expansion using a thesaurus, and automatic query expansion by common terms/phrases in top-ranked documents. Describe advantages and disadvantages of each approach. (You should aim to provide at least 3 clear points of differentiation.)

#3. Rocchio (15 points)

If our document vectors are normalized, then they live on the unit hypersphere, and we can reasonably represent a small patch of the unit hypersphere as a flat surface (just as with maps in geography!). The below picture shows such a depiction of an initial query vector, the best hits, and whether they were judged relevant or irrelevant.



a. Consider now applying Rocchio's algorithm. For fixed alpha = 10 and zero gamma, consider increasing beta. Redraw the figure above and draw a curve showing the position of the reformulated query for different values of beta. Use the values beta = 0, 10, 100 and label those points on the curve. The points drawn don't have to be in precisely the right place, but they need to be qualitatively correct.

b. Suppose now that gamma is set to the same value as beta. Where will the reformulated query be for beta = gamma = 0, 10, 100?

c. One way to construct the Rocchio relevance feedback query is as the optimal separator between relevant and non-relevant documents. (Note that this is the theoretical optimal

version of the Rocchio query and is different from the "practical" one assumed in parts a and b.)

Optimality is defined as follows. Let $r_1,...,r_m$ be the length-normalized vectors for the relevant documents, and $n_1,...,n_k$ be the length-normalized vectors for the non-relevant documents. Then the Rocchio-optimal query ρ maximizes the difference between the average correlation of relevant and non-relevant documents:

$$\rho = \arg\max_{v,||v||=1}\left[\frac{1}{m}\sum_{r_i} v \cdot r_i - \frac{1}{k}\sum_{n_i} v \cdot n_i\right]$$

where m is the number of relevant and k is the number of non-relevant documents.

Given this definition, how is ρ computed? Show your work and justify your final answer.

#4. Evaluation (10 points)

For this exercise, we define the precision-recall graph of a result list as the set of (precision/recall) points, where one precision/recall point is computed for each additional returned document. We will initially define the breakeven point as the point where precision equals recall.

a. Can there be more than one breakeven point? If yes, give an example; if not, show why not.

b. Some precision recall graphs do not have a breakeven point as defined above. An alternative definition is: The breakeven point is the point with the smallest difference between precision and recall among all those that have larger precision than recall. Write a rough pseudocode program that computes the breakeven point. You should not need more than a few lines.

#5. Naïve Bayes classifiers (15 points)

A common strategy with building Naïve Bayes classifiers is to add one parameter whereby we modify the classification function so that the the prior term is raised to some power t. The formula is now:

$$c_{NB} = \underset{c_j \in C}{\text{argmax}}\, P(c_j)^t \prod_{i \in positions} P(x_i \mid c_j)$$

a. In general, why might this be a useful thing to do? (Consider the relative impact of the prior class distribution and the multinomial evidence distribution in a Naïve Bayes model.)

b. In particular, suppose a certain multinomial Naïve Bayes classifier generates the following confusion matrix:

actual \ predicted	a	b	c	d
a	50	26	20	12
b	3	8	2	0
c	0	0	4	0
d	0	1	0	1

(A confusion matrix represents the classifications predicted by a classifier versus the correct classifications. For example, in this matrix there were 108 documents from class a, of which the classifier correctly identified 50.)

What changes in performance would you expect in the confusion matrix for different values of t? (Assume the test data has the same distribution as the training data.)

#6. Centroids (10 points)

Is the centroid of a cluster of normalized vectors normalized? If so, write a short proof. If not, provide a counterexample.

#7. K-means (15 points)

a. Consider the k-means algorithm with k = 2. The two initial centroids are chosen uniformly at random without replacement from the set of points. Give an example in a two-dimensional plane with the minimal number of points such that we can end up with two different clusterings. No examples with ties are allowed for this exercise. Give the coordinates of the points and specify both possible clusterings. Use Euclidean distance for distance computations.

b. What are the probabilities of the different clusterings under uniform random choice?