

Answer Models for Statistical Question Answering Systems



Andrés Corrada-Emmanuel
University of Massachusetts
Center for Intelligent Information Retrieval
February 13, 2004



Introduction

- What are Question Answer systems?
 - Automatic computer algorithms that return
 - Documents (Google searches, standard IR)
 - Passages (UMASS, CNLP AIDE QA system)
 - Exact answers (Language Computer Corporation)
- QA as passage retrieval (Open Domain Factoid Questions)

When was the telegraph invented? ->

telegraph, or sounder, invented by Samuel Morse
in **1860** and manufactured by Charles L. Williams.

</P>

<P>

In 1875 Williams leased his Boston workshop to a fellow named Alexander Graham Bell, who with Williams' technician, Thomas Watson, invented



Overview

1. The query likelihood algorithm
 - Bayes' Theorem
 - Approximations (unigrams, smoothing)
2. The TREC QA Passage sub task
 - Corpus
 - Current performance of systems
 - Automatic evaluation of 'answers'
3. Answer models
 - Answers have patterns
 - Putting answer models into the priors
 - Performance



Query Likelihood Ranking Algorithm

- We want to think probabilistically about passage/document retrieval (Ponte and Croft, 1998) so we want $P(\text{answer} \mid \text{question})$.
- This quantity is straightforward to calculate but it is noisy in practice. So we use Bayes' theorem to invert the relation:
 - $P(a \mid q) = P(a) P(q|a) / P(q)$
 - $P(a \mid q)$ rank equivalent to $P(q \mid a)$ if we have a uniform prior.



Query Likelihood Ranking Algorithm

- We calculate the query likelihood, $P(q | a)$ by the simplest statistical language model, unigrams
- That is, we treat pieces of text as if they were bags of words. To find the probability of seeing a word, we reach into the bag and pick out words. The probability or likelihood of producing a question/query is the product of probabilities of seeing the individual words in the piece of text:

$$P(q | a) \approx \prod_{i=1}^n P_a(q_i)$$



Query Likelihood Ranking Algorithm

- This formula is not enough. It produces zero probability for certain words. This acts like a sinkhole that does not allow us to rank the passages.
- In other words, there is a huge difference between zero and a very small probability. And a badly estimated small probability is much better than zero probability. So we 'smooth' the probabilities:

$$P_a(q_i) \approx \lambda \left(\frac{\# q_i, a}{|a|} \right) + (1 - \lambda) \left(\frac{\# q_i, C}{|C|} \right)$$



The TREC QA passages sub-task

- The Text REtrieval Conference (TREC) sponsored by NIST and DARPA has included a QA track since 1999 (TREC-8).
- The bulk of the questions utilized in the evaluations have been of the 'factoid' type.
- Starting in TREC 2002, the main task became finding 'exact answers'. But in response to concerns by myself and others that an 'exact' task does not allow showcasing of technological advances in passage retrieval systems, a passage sub-task was added in 2003.



The TREC QA passages sub-task

- The AQUAINT Corpus of English News Text has been used since 2002. It contains about 1 million documents with 3 GB of text. It consists of documents from three sources
 - AP newswire 1998-2000
 - New York Times newswire 1998-2000
 - Xinhua News Agency English newswire 1996-2000
- The main metric used for performance is number of correct answers at rank 1.
- The highest performing system came from LCC with 68.5% and UMASS ranked fourth out of 11 systems with 20.1%



The TREC QA passages sub-task

- Research on QA systems hampered relative to traditional IR research.
 - No list of 'relevant' passages.
- We approximate ground truth with answer patterns:
 - 1515 Theodore? (Seuss)?Geisel
 - 1516 cardio\s?-?\s?pulm(o|i)nary resuscitation
- Performance is measured by Mean Reciprocal Rank (MRR).
 - Correct answer at position 1 -> MRR = 1.0
 - Correct answer at position 2 -> MRR = 0.5
 - Etc.



Answer Models

- Clearly, query-likelihood is not enough.
- Factoid type questions have answers that can, sometimes, be expressed in the form of answer templates (Sabboutin and Sabboutin, 2001; Ravichandran and Hovy, 2002)
 - When was <NAME> born?
 - <NAME> was born in <BIRTHDATE>
 - <NAME> (<BIRTHDATE> -
- A statistical QA system can incorporate knowledge about these patterns by relaxing the assumption of uniform priors.



Answer Models

- Instead of $P(q|a)$, rank by $P(a)*P(q|a)$
- How do we calculate $P(a)$?
 - Use training answers to train bigram language model that allows you to estimate $P(a)$.
 - Abstract the training answers by automatically identifying and replacing with generic tokens entities such as <PERSON>, <LOCATION>, <ORGANIZATION>, <DATE>, etc.
 - Models trained for different 'classes' of questions



Performance on TREC 2002

- Answer models tested on a subset of TREC 2002
500 questions (385 out of the 500)
 - Date (110) (<Date>, <Year>, <Number>)
 - Person (91) (<Person>)
 - Geo-Political Entity (GPE) (88) (<Location>, <Organization>)
 - Definition (51) (?)
 - Quantity (45) (<Number>, <Percent>)



Performance on TREC 2002

- Baseline performance (250 byte passages)

| Dataset | Rank 1 correct | MRR(5) | MRR(20) |
|-----------------|----------------|--------|---------|
| Full set | 25.6% | 31.1% | 32.6% |
| 385 together | 24.4% | 31.1% | 32.6% |
| 385 tuned apart | 26.2% | 32.3% | 33.7% |



Performance on TREC 2002

- Ten fold validation experiment (Rank 1 correct)

| q. type | QL | QL + AM |
|------------|-------|---------------|
| Date | 20.9% | *30.9% (0.60) |
| Person | 34.1% | 34.1% (0.10) |
| GPE | 34.1% | 36.4% (0.35) |
| Definition | 19.6% | 21.6% (0.40) |
| Quantity | 15.6% | 26.7% (0.70) |
| All | 26.2% | 31.2% |



Conclusions

- ❑ Answer models perform well on Date and Quantity categories.
- ❑ Person and GPE were expected to perform well but show a small or no improvement. Why?
- ❑ We can improve performance on the TREC passage sub-task from ~25% to ~30% by classifying questions beforehand and using Date and Quantity answer models.
- ❑ Approach needs to be tested across corpora (TREC 2001 -> TREC 2002).