CSCI 5582
Artificial Intelligence

Lecture 26

Jim Martin

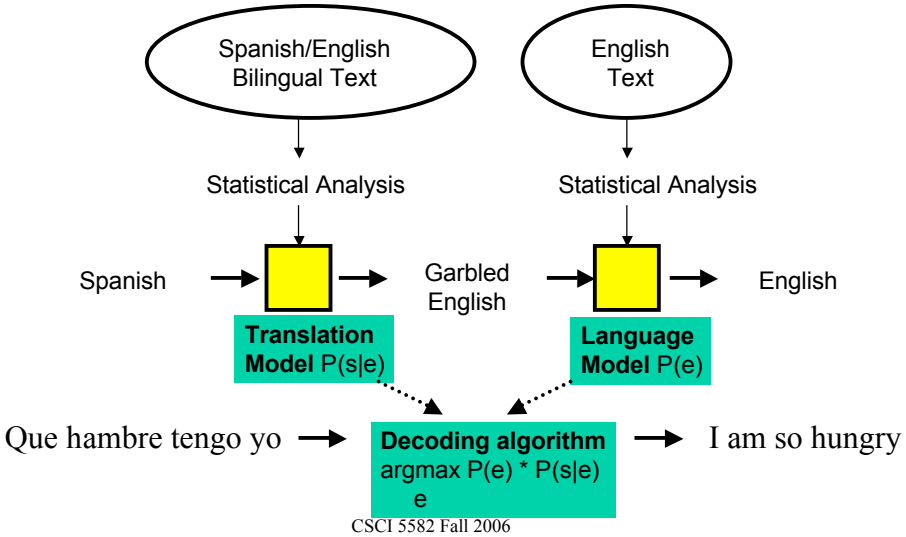
Today 12/12

- Machine Translation
  - Review
  - Automatic Evaluation
- Question Answering

Readings

- Chapters 22 and 23 in Russell and Norvig for language stuff in general
- Chapter 24 of Jurafsky and Martin for MT material

Statistical MT Systems



Four Problems for Statistical MT

- Language model
  - Given an English string e, assigns P(e) by the usual methods we've been using sequence modeling.
- Translation model
  - Given a pair of strings <f,e>, assigns P(f | e) again by making the usual markov assumptions
- Training
  - Getting the numbers needed for the models
- Decoding algorithm
  - Given a language model, a translation model, and a new sentence f … find translation e maximizing P(e) * P(f | e)

Remember though that what we really need is argmax P(e|f)

Evaluation

- There are 2 dimensions along which MT systems can be evaluated
  - Fluency
    - How good is the output text as an example of the target language
  - Fidelity
    - How well does the output text convey the source text
      - Information content and style

Evaluating MT: Human tests for **fluency**

- Rating tests: Give human raters a scale (1 to 5) and ask them to rate
  - For distinct scales for
    - Clarity, Naturalness, Style
  - Check for specific problems
    - Cohesion (Lexical chains, anaphora, ellipsis)
      - Hand-checking for cohesion.
    - Well-formedness
      - 5-point scale of syntactic correctness

Evaluating MT: Human tests for **fidelity**

- Adequacy
  - Does it convey the information in the original?
  - Ask raters to rate on a scale
    - Bilingual raters: give them source and target sentence, ask how much information is preserved
    - Monolingual raters: give them target + a good human translation

Evaluating MT: Human tests for **fidelity**

- Informativeness
  - Task based: is there enough info to do some task?

Evaluating MT: Problems

- Asking humans to judge sentences on a 5-point scale for 10 factors takes time and $$$ (weeks or months!)
- Need a metric that can be run every time the algorithm is altered.
- It's OK if it isn't perfect, just needs to correlate with the human metrics, which can still be run periodically.

Automatic evaluation

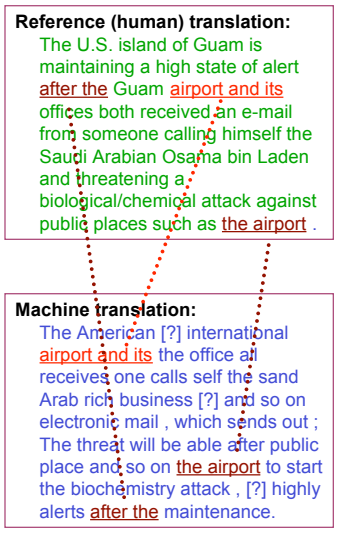
- Assume we have one or more human translations of the source passage
- Compare the automatic translation to these human translations using some simple metric
  - BLEU score

BiLingual Evaluation Understudy (BLEU)

- Automatic scoring
- Requires human reference translations
- Approach:
  - Produce corpus of high-quality human translations
  - Judge "closeness" numerically by comparing n-gram matches between candidate translations and 1 or more reference translations

Slide from Bonnie Dorr

BLEU Evaluation Metric



N-gram precision
(score is between 0 & 1)
- What percentage of machine n-grams can be found in the reference translation?

BLEU Evaluation Metric

- Two problems (ways to *game*) that metric…
  1. Repeat a high frequency n-gram over and over
     "of the of the of the of the"
  2. Don't say much at all
     "the"

BLEU Evaluation Metric

- Tweaks to N-Gram precision
  - Counting N-Grams by type, not token
    - "of the" only gets looked at once
  - Brevity penalty

BLEU Evaluation Metric

Reference (human) translation:
The U.S. island of Guam is maintaining a high state of alert after the Guam airport and its offices both received an e-mail from someone calling himself the Saudi Arabian Osama bin Laden and threatening a biological/chemical attack against public places such as the airport .

Machine translation:
The American [?] international airport and its the office all receives one calls self the sand Arab rich business [?] and so on electronic mail , which sends out ; The threat will be able after public place and so on the airport to start the biochemistry attack , [?] highly alerts after the maintenance.

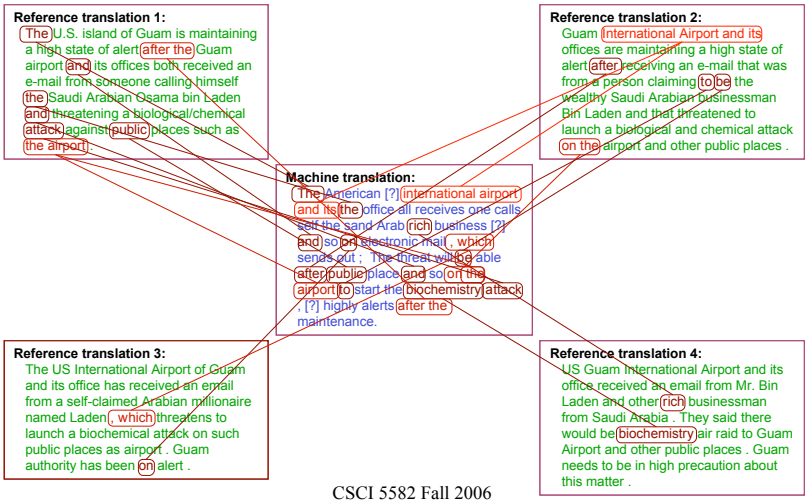
Slide from Bonnie Dorr

- BLEU4 formula
  (counts n-grams up to length 4)

exp (1.0 * log p1 +
0.5 * log p2 +
0.25 * log p3 +
0.125 * log p4 –
max(words-in-reference / words-in-machine – 1,
0)

p1 = 1-gram precision
P2 = 2-gram precision
P3 = 3-gram precision
P4 = 4-gram precision

Multiple Reference Translations



BLEU in Action

枪手被警方击毙。 **(Foreign Original)**

the gunman was shot to death by the police . **(Reference Translation)**

the gunman was police kill .	**#1**
wounded police jaya of	**#2**
the gunman was shot dead by the police .	**#3**
the gunman arrested by police kill .	**#4**
the gunmen were killed .	**#5**
the gunman was shot to death by the police .	**#6**
gunmen were killed by police ?SUB>0 ?SUB>0	**#7**
al by the police .	**#8**
the ringer is killed by the police .	**#9**
police killed the gunman .	**#10**

Slide from Bonnie Dorr

BLEU in Action

枪手被警方击毙。 (Foreign Original)

the gunman was shot to death by the police . (Reference Translation)

the gunman was police kill .	#1
wounded police jaya of	#2
the gunman was shot dead by the police .	#3
the gunman arrested by police kill .	#4
the gunmen were killed .	#5
the gunman was shot to death by the police .	#6
gunmen were killed by police ?SUB>0 ?SUB>0	#7
al by the police .	#8
the ringer is killed by the police .	#9
police killed the gunman .	#10

green = 4-gram match (good!)
red = word not matched (bad!)

Slide from Bonnie Dorr

Bleu Comparison

Chinese-English Translation Example:

Candidate 1: It is a guide to action which ensures that the military always obeys the commands of the party.

Candidate 2: It is to insure the troops forever hearing the activity guidebook that party direct.

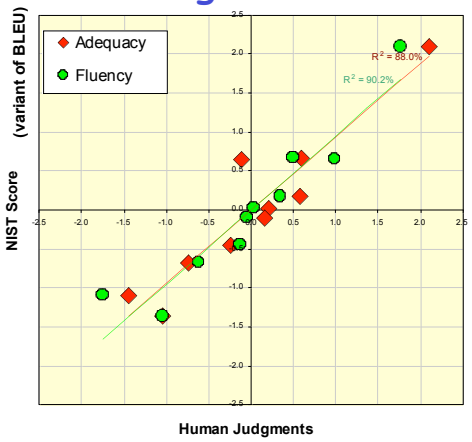
Reference 1: It is a guide to action that ensures that the military will forever heed Party commands.

Reference 2: It is the guiding principle which guarantees the military forces always being under the command of the Party.

Reference 3: It is the practical guide for the army always to heed the directions of the party.

Slide from Bonnie Dorr

BLEU Tends to Predict Human Judgments



Current Results

These results are not to be construed, or represented as endorsements of any participant's system or commercial product, or as official findings on the part of NIST or the U.S. Government. Note that the results submitted by developers of commercial MT products were generally from research systems, not commercially available products. Since MT-06 was an evaluation of research algorithms, the MT-06 test design required local implementation by each participant. As such, participants were only required to submit their translation system output to NIST for uniform scoring and analysis. The systems themselves were not independently evaluated by NIST.

Current Results

NIST data set BLEU-4 Score

Site ID	Language	Overall	Newswire	Newsgroup	Broadcast News
google	Arabic	0.4569	0.5060	0.3727	0.4076
google	Chinese	0.3615	0.3725	0.2926	0.3859

Unlimited Data Track (Train on NIST Data + whatever else)

- Chinese performance significantly worse than Arabic across all the best participants.

Break

- Final is the 18th (Monday). Right here.
- Next class is a review class
  - Come prepared with questions
  - Even better email me your questions ahead of time so I can figure out an answer.
- I am still going to send out a new test set for the last HW evaluation

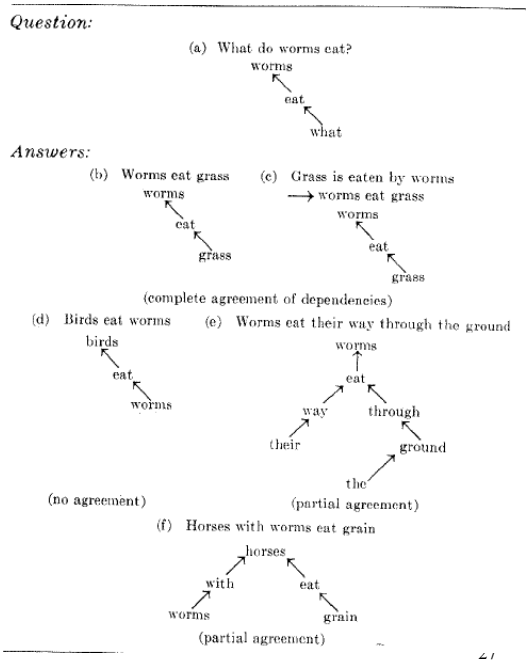
Question-Answering from the Web

- The notion of getting computers to give reasonable answers to questions has been around for quite awhile
- Three kinds of systems
  1) Finding answers in text collections
  2) Interfaces to relational databases
  3) Mixed initiative dialog systems

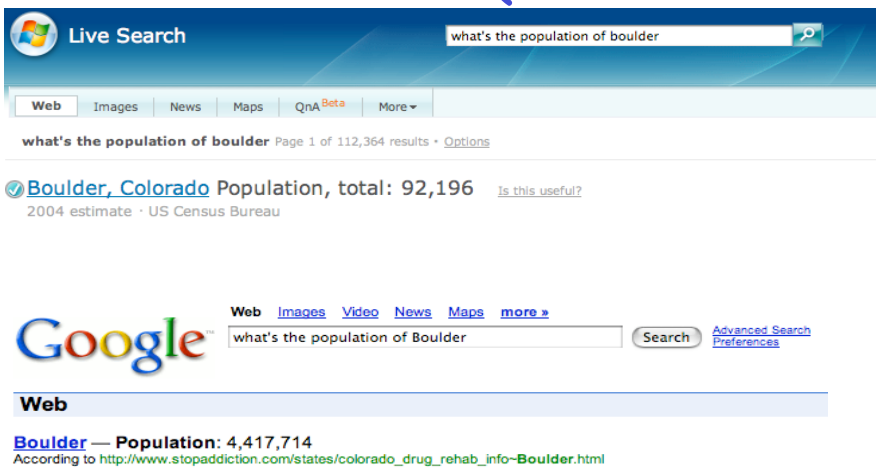
Finding Answers in Text

- Not a new idea… (Simmons et al 1963)
  - Take an encyclopedia and load it onto a computer.
  - Take a question and parse it into a logical form
  - Perform simple information retrieval to get relevant texts
  - Parse those into a logical form
  - Match and rank

Simmons, Klein, McConlogue 1963: Parse Q+A using dependency parser (Hays 1962)



Web QA



Finding Answers in Text

- Fundamentally, this is about modifying, processing, enriching or marking up both the question and potential answer texts to allow a simple match.
- All current systems do pretty much that.

29

People *do* ask questions…

Examples from search engine query logs

Which english translation of the bible is used in official Catholic liturgies?
How tall is the sears tower
How can i find someone in texas
Where can i find information on puritan religion?
What are the 7 wonders of the world
How can i eliminate stress
What vacuum cleaner does Consumers Guide recommend

Full-Blown Heavy-Weight System

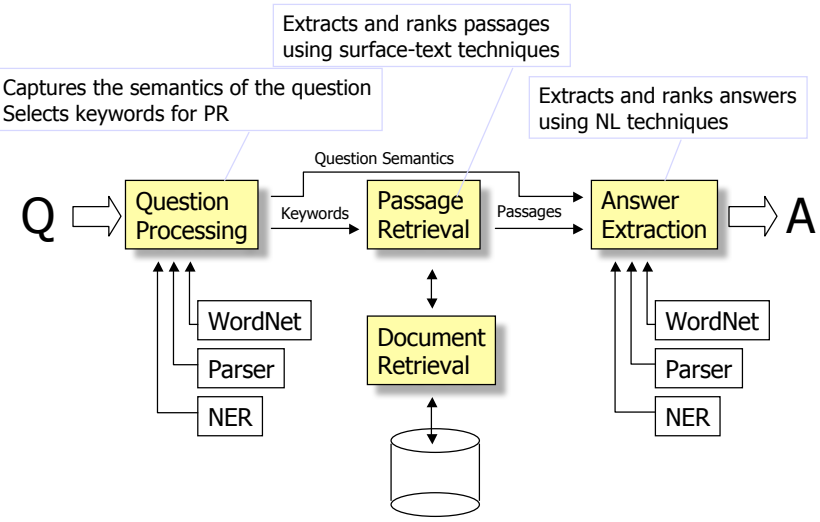
- Parse and analyze the question
- Formulate queries suitable for use with an IR system (search engine)
- Retrieve ranked results
- Break into suitable units
- Perform NLP on those rank units
- Re-Rank snippets based on NLP processing
- Done

31

UT Dallas Q/A Systems

- This system contains many components used by other systems, but more complex in some ways
- Next slides based mainly on:
  - Paşca and Harabagiu, *High-Performance Question Answering from Large Text Collections*, SIGIR'01.
  - Paşca and Harabagiu, *Answer Mining from Online Documents*, ACL'01.
  - Harabagiu, Paşca, Maiorano: *Experiments with Open-Domain Textual Question Answering*. COLING'00

QA Block Architecture



Question Processing

- Two main tasks
  - Determining the type of the answer
    - →If you know the type of the answer you can focus your processing only on docs that have things of the right type
  - Extract keywords from the question and formulate a query
    - →Assume that a generic IR search engine can find docs with an answer (and lots that don't). Ie. The NLP/QA system is dealing with precision not recall

Answer Types

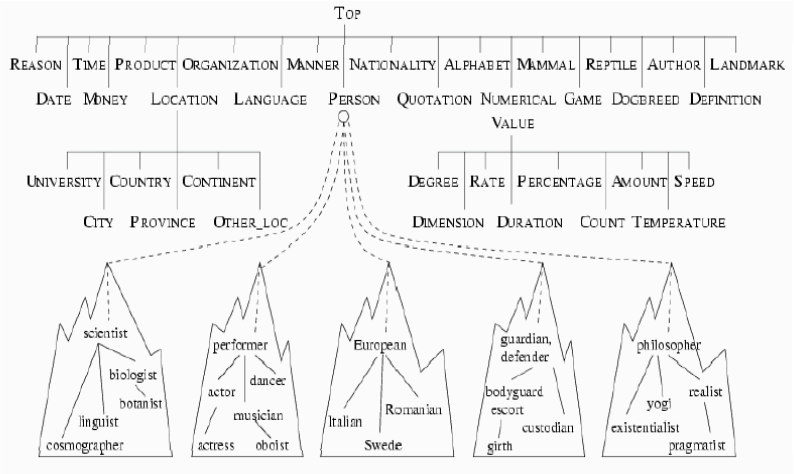
- Factoid questions…
  - Who, where, when, how many…
  - The answers fall into a limited and somewhat predictable set of categories
    - Who questions are going to be answered by…
    - Where questions…
  - Generally, systems select answer types from a set of Named Entities, augmented with other types that are relatively easy to extract

Answer Types

- Of course, it isn't that easy…
  - Who questions can have organizations as answers
    - Who sells the most hybrid cars?
  - Which questions can have people as answers
    - Which president went to war with Mexico?

Answer Type Taxonomy

- Contains ~9000 concepts reflecting expected answer types



Answer Type Detection

- Most systems use a combination of hand-crafted rules and supervised machine learning to determine the right answer type for a question.
- But remember our notion of matching. It doesn't do any good to do something complex here if it can't also be done in potential answer texts.

Keyword Selection

- Answer Type indicates *what* the question is looking for, but that doesn't really help in finding relevant texts (i.e. Ok, let's look for texts with people in them)
- Lexical terms (keywords) from the question, possibly expanded with lexical/semantic variations provide the required context.

Lexical Terms Extraction

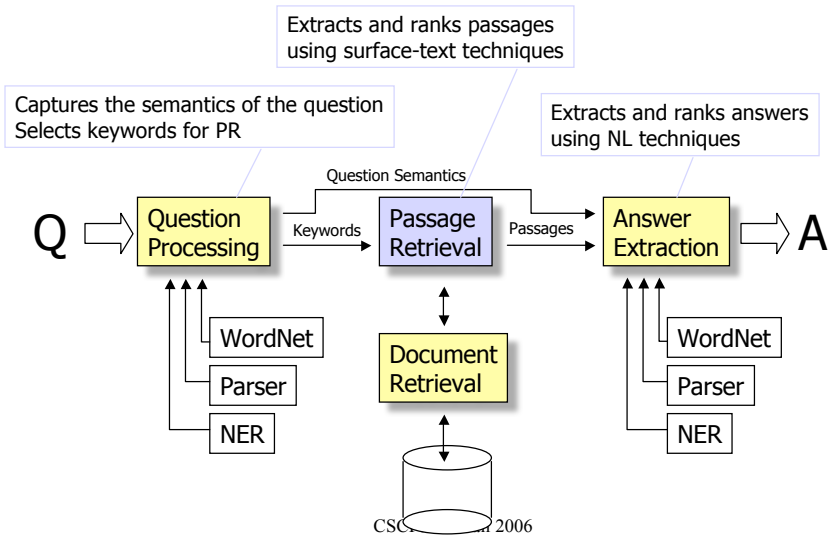
- Questions approximated by sets of unrelated words (lexical terms)
- Similar to bag-of-word IR models

Question (from TREC QA track)	Lexical terms
Q002: What was the monetary value of the Nobel Peace Prize in 1989?	monetary, value, Nobel, Peace, Prize
Q003: What does the Peugeot company manufacture?	Peugeot, company, manufacture
Q004: How much did Mercury spend on advertising in 1993?	Mercury, spend, advertising, 1993
Q005: What is the name of the managing director of Apricot Computer?	name, managing, director, Apricot, Computer

Keyword Selection Algorithm

- Select all non-stopwords in quotations
- Select all NNP words in recognized named entities
- Select all complex nominals with their adjectival modifiers
- Select all other complex nominals
- Select all nouns with adjectival modifiers
- Select all other nouns
- Select all verbs
- Select the answer type word

Passage Retrieval

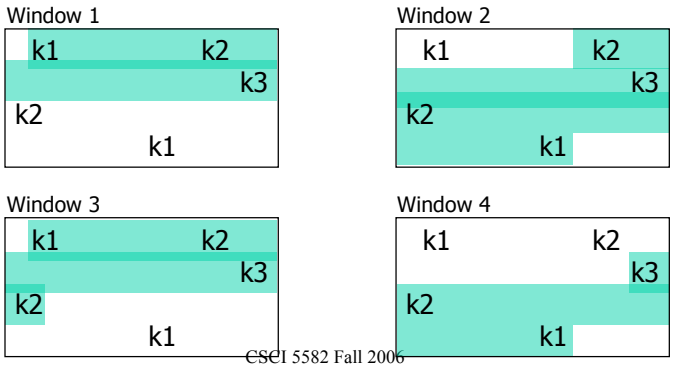


Passage Extraction Loop

- Passage Extraction Component
  - Extracts passages that contain all selected keywords
  - Passage size dynamic
  - Start position dynamic
- Passage quality and keyword adjustment
  - In the first iteration use the first 6 keyword selection heuristics
  - If the number of passages is lower than a threshold ⇒ query is too strict ⇒ drop a keyword
  - If the number of passages is higher than a threshold ⇒ query is too relaxed ⇒ add a keyword

Passage Scoring

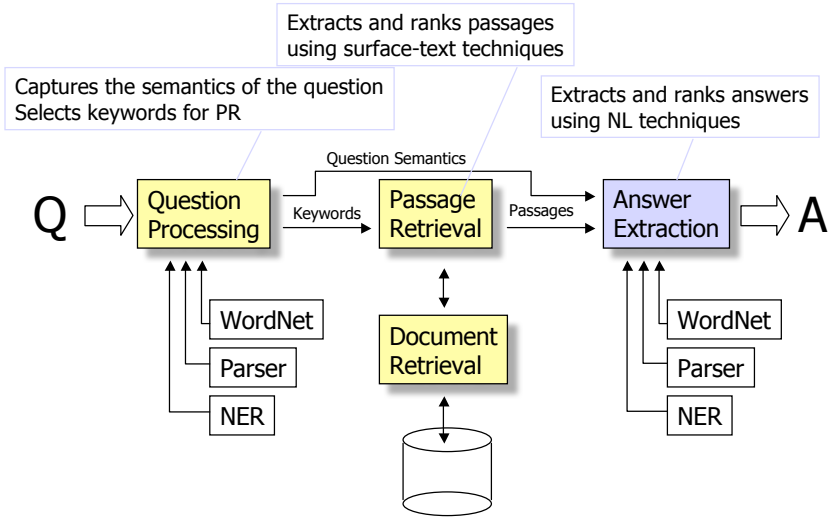
- Passages are scored based on keyword windows
  - For example, if a question has a set of keywords: {k1, k2, k3, k4}, and in a passage k1 and k2 are matched twice, k3 is matched once, and k4 is not matched, the following windows are built:



Passage Scoring

- Passage ordering is performed using a trained re-ranking algorithm that involves three scores:
  - The number of words from the question that are recognized in the same sequence in the window
  - The number of words that separate the most distant keywords in the window
  - The number of unmatched keywords in the window

Answer Extraction



Ranking Candidate Answers

Q066: Name the first private citizen to fly in space.

- Answer type: Person
- Text passage:

  "Among them was Christa McAuliffe, the first private citizen to fly in space. Karen Allen, best known for her starring role in "Raiders of the Lost Ark", plays McAuliffe. Brian Kerwin is featured as shuttle pilot Mike_Smith..."

Ranking Candidate Answers

Q066: Name the first private citizen to fly in space.

- Answer type: Person
- Text passage:

  "Among them was Christa McAuliffe, the first private citizen to fly in space. Karen Allen, best known for her starring role in "Raiders of the Lost Ark", plays McAuliffe. Brian Kerwin is featured as shuttle pilot Mike_Smith..."

- Best candidate answer: Christa McAuliffe

Features for Answer Ranking

- Number of question terms matched in the answer passage
- Number of question terms matched in the same phrase as the candidate answer
- Number of question terms matched in the same sentence as the candidate answer
- Flag set to 1 if the candidate answer is followed by a punctuation sign
- Number of question terms matched, separated from the candidate answer by at most three words and one comma
- Number of terms occurring in the same order in the answer passage as in the question
- Average distance from candidate answer to question term matches

Evaluation

- Evaluation of this kind of system is usually based on some kind of TREC-like metric.
- In Q/A the most frequent metric is
  - Mean Reciprocal Rank

    You're allowed to return N answers. Your score is based on 1/Rank of the first right answer.

    Averaged over all the questions you answer.

Is the Web Different?

- In TREC (and most commercial applications), retrieval is performed against a closed relatively homogeneous collection of texts.
- The diversity/creativity in how people express themselves necessitates all that work to bring the question and the answer texts together.
- But…

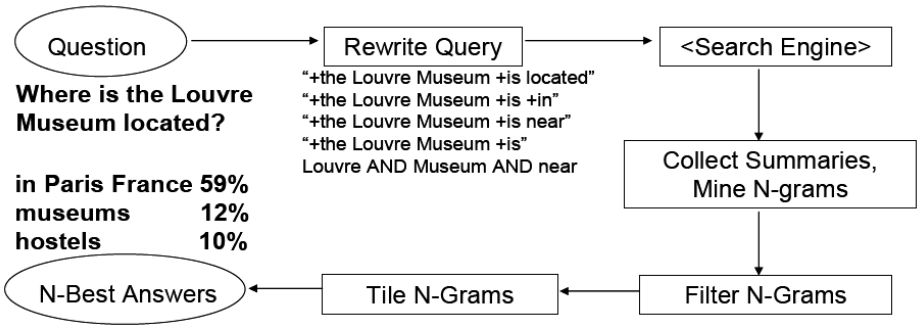
The Web is Different

- On the Web popular factoids are likely to be expressed in a gazzilion different ways.
- At least a few of which will likely match the way the question was asked.
- So why not just grep (or agrep) the Web using all or pieces of the original question.

AskMSR

- Process the question by…
  - Simple rewrite rules to rewriting the original question into a statement
    - Involves detecting the answer type
- Get some results
- Extract answers of the right type based on
  - How often they occur

AskMSR



Step 1: Rewrite the questions

- Intuition: Users' questions are often syntactically quite close to sentences that contain the answer
  - Where is the Louvre Museum located?
    - The Louvre Museum is located in ***Paris***
  - Who created the character of Scrooge?
    - ***Charles Dickens*** created the character of Scrooge.

Query rewriting

Classify question into seven categories

- **Who** is/was/are/were…?
- **When** is/did/will/are/were …?
- **Where** is/are/were …?

a. Hand-crafted category-specific transformation rules
e.g.: For *where* questions, move 'is' to all possible locations
Look to the right of the query terms for the answer.

"Where is the Louvre Museum located?"
→ "is the Louvre Museum located"
→ "the is Louvre Museum located"
→ "the Louvre is Museum located"
→ "the Louvre Museum is located"
→ "the Louvre Museum located is"

Step 2: Query search engine

- Send all rewrites to a Web search engine
- Retrieve top N answers (100-200)
- For speed, rely just on search engine's "snippets", not the full text of the actual document

Step 3: Gathering N-Grams

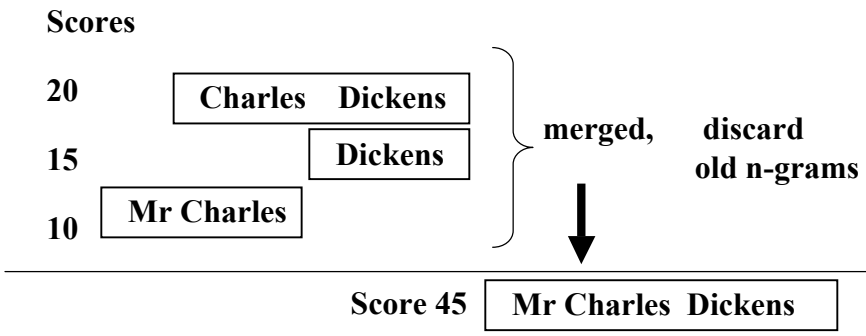
- Enumerate all N-grams (N=1,2,3) in all retrieved snippets
- Weight of an n-gram: occurrence count, each weighted by "reliability" (weight) of rewrite rule that fetched the document (can be trained).
  - Example: "Who created the character of Scrooge?"

Dickens		117
Christmas Carol	78	
Charles Dickens	75	
Disney		72
Carl Banks		54
A Christmas		41
Christmas Carol	45	
Uncle		31

Step 4: Filtering N-Grams

- Each question type is associated with one or more "data-type filters" = regular expressions for answer types
- Boost score of n-grams that match the expected answer type.
- Lower score of n-grams that don't match.
- For example
  - The filter for
    - How many dogs pull a sled in the Iditarod?
  - prefers a number
  - So disprefer candidate n-grams like
    - Dog race, run, Alaskan, dog racing
  - Prefer canddiate n-grams like
    - Pool of 16 dogs

Step 5: Tiling the Answers



Results

- Standard TREC contest test-bed (TREC 2001): 1M documents; 900 questions
  - Technique does ok, not great (would have placed in top 9 of ~30 participants)
    - MRR = 0.507
  - But with access to the Web… They do much better, would have come in second on TREC 2001
    - Be suspicious of any *after the bake-off is over* metrics