Shallow & Deep QA Systems

Ling 573
NLP Systems and Applications
April 9, 2013

Announcement

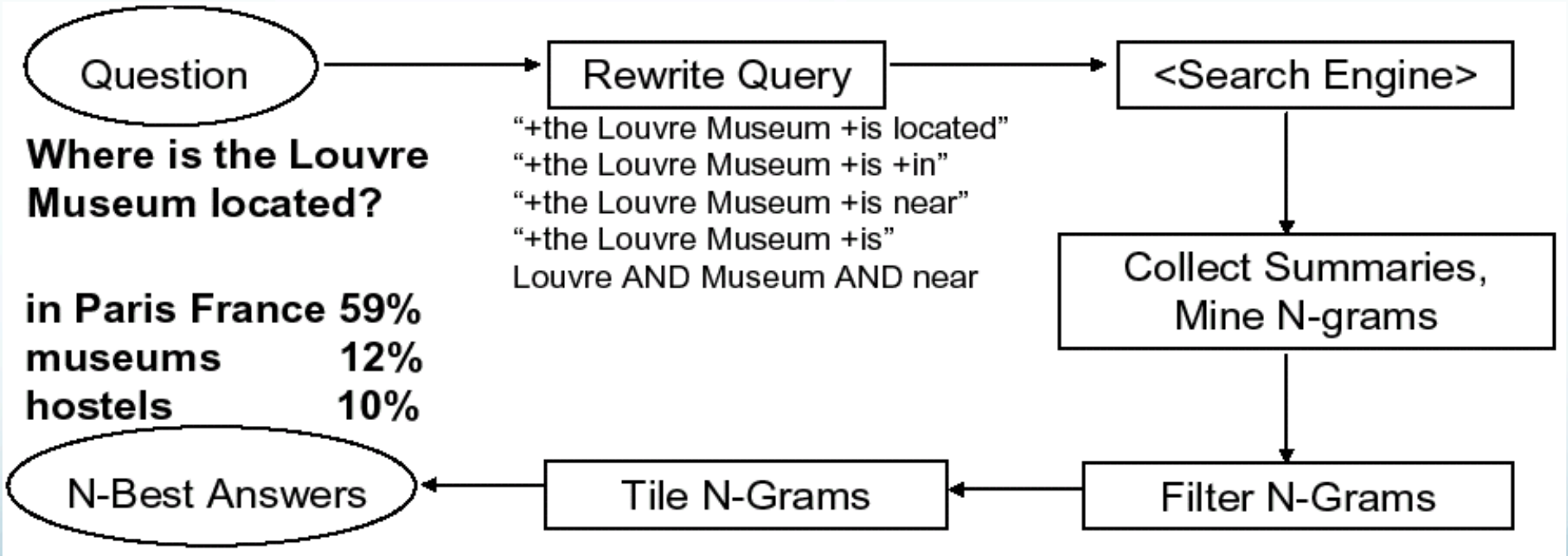
- Thursday's class will be pre-recorded.
- Will be accessed from the Adobe Connect recording.
- Will be linked before regular Thursday class time.
- Please post any questions to the GoPost.

Roadmap

- Two extremes in QA systems:
  - Redundancy-based QA: Aranea
  - LCC's PowerAnswer-2
- Deliverable #2

Redundancy-based QA

- AskMSR (2001,2002); Aranea (Lin, 2007)



Redundancy-based QA

- Systems exploit statistical regularity to find “easy” answers to factoid questions on the Web

Redundancy-based QA

- Systems exploit statistical regularity to find "easy" answers to factoid questions on the Web
  - —When did Alaska become a state?
  - **(1) Alaska became a state on January 3, 1959.**
  - **(2) Alaska was admitted to the Union on January 3, 1959.**

Redundancy-based QA

- Systems exploit statistical regularity to find "easy" answers to factoid questions on the Web
  - —When did Alaska become a state?
  - **(1) Alaska became a state on January 3, 1959.**
  - **(2) Alaska was admitted to the Union on January 3, 1959.**

  - —Who killed Abraham Lincoln?
  - **(1) John Wilkes Booth killed Abraham Lincoln.**
  - **(2) John Wilkes Booth altered history with a bullet. He will forever be known as the man who ended Abraham Lincoln's life.**

Redundancy-based QA

- Systems exploit statistical regularity to find "easy" answers to factoid questions on the Web
  - —When did Alaska become a state?
  - **(1) Alaska became a state on January 3, 1959.**
  - **(2) Alaska was admitted to the Union on January 3, 1959.**
  - —Who killed Abraham Lincoln?
  - **(1) John Wilkes Booth killed Abraham Lincoln.**
  - **(2) John Wilkes Booth altered history with a bullet. He will forever be known as the man who ended Abraham Lincoln's life.**
- Text collection

Redundancy-based QA

- Systems exploit statistical regularity to find "easy" answers to factoid questions on the Web
  - —When did Alaska become a state?
  - **(1) Alaska became a state on January 3, 1959.**
  - **(2) Alaska was admitted to the Union on January 3, 1959.**

  - —Who killed Abraham Lincoln?
  - **(1) John Wilkes Booth killed Abraham Lincoln.**
  - **(2) John Wilkes Booth altered history with a bullet. He will forever be known as the man who ended Abraham Lincoln's life.**
- Text collection may only have (2), but web?

Redundancy-based QA

- Systems exploit statistical regularity to find "easy" answers to factoid questions on the Web
  - —When did Alaska become a state?
  - **(1) Alaska became a state on January 3, 1959.**
  - **(2) Alaska was admitted to the Union on January 3, 1959.**
  - —Who killed Abraham Lincoln?
  - **(1) John Wilkes Booth killed Abraham Lincoln.**
  - **(2) John Wilkes Booth altered history with a bullet. He will forever be known as the man who ended Abraham Lincoln's life.**
- Text collection may only have (2), but web? anything

Redundancy & Answers

- How does redundancy help find answers?

Redundancy & Answers

- How does redundancy help find answers?
- Typical approach:
  - Answer type matching
    - E.g. NER, but
    - Relies on large knowledge-base
- Redundancy approach:

Redundancy & Answers

- How does redundancy help find answers?
- Typical approach:
  - Answer type matching
    - E.g. NER, but
    - Relies on large knowledge-based
- Redundancy approach:
  - Answer should have high correlation w/query terms
    - Present in many passages
      - Uses n-gram generation and processing

Redundancy & Answers

- How does redundancy help find answers?
- Typical approach:
  - Answer type matching
    - E.g. NER, but
    - Relies on large knowledge-based
- Redundancy approach:
  - Answer should have high correlation w/query terms
    - Present in many passages
      - Uses n-gram generation and processing
  - In 'easy' passages, simple string match effective

Redundancy Approaches

- AskMSR (2001):
  - Lenient: 0.43; Rank: 6/36; Strict: 0.35; Rank: 9/36

Redundancy Approaches

- AskMSR (2001):
  - Lenient: 0.43; Rank: 6/36; Strict: 0.35; Rank: 9/36
- Aranea (2002, 2003):
  - Lenient: 45%; Rank: 5; Strict: 30%; Rank:6-8

Redundancy Approaches

- AskMSR (2001):
  - Lenient: 0.43; Rank: 6/36; Strict: 0.35; Rank: 9/36
- Aranea (2002, 2003):
  - Lenient: 45%; Rank: 5; Strict: 30%; Rank:6-8
- Concordia (2007): Strict: 25%; Rank 5

Redundancy Approaches

- AskMSR (2001):
  - Lenient: 0.43; Rank: 6/36; Strict: 0.35; Rank: 9/36
- Aranea (2002, 2003):
  - Lenient: 45%; Rank: 5; Strict: 30%; Rank:6-8
- Concordia (2007): Strict: 25%; Rank 5
- Many systems incorporate some redundancy
  - Answer validation
  - Answer reranking
    - LCC: huge knowledge-based system, redundancy improved

Intuition

- Redundancy is useful!
  - If similar strings appear in many candidate answers, likely to be solution
    - Even if can't find obvious answer strings

Intuition

- Redundancy is useful!
  - If similar strings appear in many candidate answers, likely to be solution
    - Even if can't find obvious answer strings
- Q: How many times did Bjorn Borg win Wimbledon?
  - Bjorn Borg blah blah blah Wimbledon blah 5 blah
  - Wimbledon blah blah blah Bjorn Borg blah 37 blah.
  - blah Bjorn Borg blah blah 5 blah blah Wimbledon
  - 5 blah blah Wimbledon blah blah Bjorn Borg.

Intuition

- Redundancy is useful!
  - If similar strings appear in many candidate answers, likely to be solution
    - Even if can't find obvious answer strings
- Q: How many times did Bjorn Borg win Wimbledon?
  - Bjorn Borg blah blah blah Wimbledon blah 5 blah
  - Wimbledon blah blah blah Bjorn Borg blah 37 blah.
  - blah Bjorn Borg blah blah 5 blah blah Wimbledon
  - 5 blah blah Wimbledon blah blah Bjorn Borg.
  - Probably 5

Query Reformulation

- Identify question type:
  - E.g. Who, When, Where,...
- Create question-type specific rewrite rules:

Query Reformulation

- Identify question type:
  - E.g. Who, When, Where,…
- Create question-type specific rewrite rules:
  - Hypothesis: Wording of question similar to answer
    - For ‘where’ queries, move ‘is’ to all possible positions
      - Where is the Louvre Museum located? =>
        - Is the Louvre Museum located
        - The is Louvre Museum located
        - The Louvre Museum is located, .etc.

Query Reformulation

- Identify question type:
  - E.g. Who, When, Where,…
- Create question-type specific rewrite rules:
  - Hypothesis: Wording of question similar to answer
    - For ‘where’ queries, move ‘is’ to all possible positions
      - Where is the Louvre Museum located? =>
        - Is the Louvre Museum located
        - The is Louvre Museum located
        - The Louvre Museum is located, .etc.
- Create type-specific answer type (Person, Date, Loc)

Query Form Generation

- 3 query forms:
  - Initial baseline query

Query Form Generation

- 3 query forms:
  - Initial baseline query
  - Exact reformulation: weighted 5 times higher
    - Attempts to anticipate location of answer

Query Form Generation

- 3 query forms:
  - Initial baseline query
  - Exact reformulation: weighted 5 times higher
    - Attempts to anticipate location of answer
    - Extract using surface patterns
      - **‘‘When was the telephone invented?’’**

Query Form Generation

- 3 query forms:
  - Initial baseline query
  - Exact reformulation: weighted 5 times higher
    - Attempts to anticipate location of answer
    - Extract using surface patterns
      - **‘‘When was the telephone invented?’’**
      - **‘‘the telephone was invented ?x’’**

Query Form Generation

- 3 query forms:
  - Initial baseline query
  - Exact reformulation: weighted 5 times higher
    - Attempts to anticipate location of answer
    - Extract using surface patterns
      - **‘‘When was the telephone invented?’’**
      - **‘‘the telephone was invented ?x’’**
    - Generated by ~12 pattern matching rules on terms, POS
      - E.g. wh-word did A verb B -

Query Form Generation

- 3 query forms:
  - Initial baseline query
  - Exact reformulation: weighted 5 times higher
    - Attempts to anticipate location of answer
    - Extract using surface patterns
      - **“When was the telephone invented?”**
      - **“the telephone was invented ?x”**
    - Generated by ~12 pattern matching rules on terms, POS
      - E.g. wh-word did A verb B -> A verb+ed B ?x (general)
      - Where is A? ->

Query Form Generation

- 3 query forms:
  - Initial baseline query
  - Exact reformulation: weighted 5 times higher
    - Attempts to anticipate location of answer
    - Extract using surface patterns
      - **"When was the telephone invented?"**
      - **"the telephone was invented ?x"**
    - Generated by ~12 pattern matching rules on terms, POS
      - E.g. wh-word did A verb B -> A verb+ed B ?x (general)
      - Where is A? -> A is located in ?x (specific)
  - Inexact reformulation: bag-of-words

Query Reformulation

- Examples

What year did Alaska become a state?

[baseline]	What year did Alaska become a state
[inexact]	Alaska became a state
[exact]	Alaska became a state ?x

Who was the first person to run the mile in less than four minutes?

[baseline]	Who was the first person to run the mile in less than four minutes?
[inexact]	the first person to run the mile in less than four minutes
[exact]	the first person to run the mile in less than four minutes was ?x
[exact]	?x was the first person to run the mile in less than four minutes

Redundancy-based Answer Extraction

- Prior processing:
  - Question formulation
  - Web search
  - Retrieve snippets – top 100

Redundancy-based Answer Extraction

- Prior processing:
  - Question formulation
  - Web search
  - Retrieve snippets – top 100
- N-grams:
  - Generation
  - Voting
  - Filtering
  - Combining
  - Scoring
  - Reranking

N-gram Generation & Voting

- N-gram generation from unique snippets:
  - Approximate chunking – without syntax
  - All uni-, bi-, tri-, tetra- grams
    - Concordia added 5-grams (prior errors)

N-gram Generation & Voting

- N-gram generation from unique snippets:
  - Approximate chunking – without syntax
  - All uni-, bi-, tri-, tetra- grams
    - Concordia added 5-grams (prior errors)
  - Score: based on source query: exact 5x, others 1x
- N-gram voting:
  - Collates n-grams
  - N-gram gets sum of scores of occurrences
  - What would be highest ranked ?

N-gram Generation & Voting

- N-gram generation from unique snippets:
  - Approximate chunking – without syntax
  - All uni-, bi-, tri-, tetra- grams
    - Concordia added 5-grams (prior errors)
  - Score: based on source query: exact 5x, others 1x
- N-gram voting:
  - Collates n-grams
  - N-gram gets sum of scores of occurrences
  - What would be highest ranked ?
    - Specific, frequent: Question terms, stopwords

N-gram Filtering

- Throws out ‘blatant’ errors
  - Conservative or aggressive?

N-gram Filtering

- Throws out ‘blatant’ errors
  - Conservative or aggressive?
    - Conservative: can’t recover error
- Question-type-neutral filters:

N-gram Filtering

- Throws out ‘blatant’ errors
  - Conservative or aggressive?
    - Conservative: can’t recover error
- Question-type-neutral filters:
  - Exclude if begin/end with stopword
  - Exclude if contain words from question, except
    - ‘Focus words’ : e.g. units
- Question-type-specific filters:

N-gram Filtering

- Throws out ‘blatant’ errors
  - Conservative or aggressive?
    - Conservative: can’t recover error
- Question-type-neutral filters:
  - Exclude if begin/end with stopword
  - Exclude if contain words from question, except
    - ‘Focus words’ : e.g. units
- Question-type-specific filters:
  - ‘how far’, ‘how fast’:

N-gram Filtering

- Throws out ‘blatant’ errors
  - Conservative or aggressive?
    - Conservative: can’t recover error
- Question-type-neutral filters:
  - Exclude if begin/end with stopword
  - Exclude if contain words from question, except
    - ‘Focus words’ : e.g. units
- Question-type-specific filters:
  - ‘how far’, ‘how fast’: exclude if no numeric
  - ‘who’,’where’:

N-gram Filtering

- Throws out ‘blatant’ errors
  - Conservative or aggressive?
    - Conservative: can’t recover error
- Question-type-neutral filters:
  - Exclude if begin/end with stopword
  - Exclude if contain words from question, except
    - ‘Focus words’ : e.g. units
- Question-type-specific filters:
  - ‘how far’, ‘how fast’: exclude if no numeric
  - ‘who’,’where’: exclude if not NE (first & last caps)

N-gram Filtering

- Closed-class filters:
  - Exclude if not members of an enumerable list

N-gram Filtering

- Closed-class filters:
  - Exclude if not members of an enumerable list
  - E.g. ‘what year ‘ -> must be acceptable date year

N-gram Filtering

- Closed-class filters:
  - Exclude if not members of an enumerable list
  - E.g. ‘what year ‘ -> must be acceptable date year

- Example after filtering:
  - Who was the first person to run a sub-four-minute mile?

Candidate	Score
Bannister	137
Roger	114
Roger Bannister	103
English	26
...	...

N-gram Filtering

- Impact of different filters:
  - Highly significant differences when run w/subsets

N-gram Filtering

- Impact of different filters:
  - Highly significant differences when run w/subsets
  - No filters: drops 70%

N-gram Filtering

- Impact of different filters:
  - Highly significant differences when run w/subsets
  - No filters: drops 70%
  - Type-neutral only: drops 15%

N-gram Filtering

- Impact of different filters:
  - Highly significant differences when run w/subsets
  - No filters: drops 70%
  - Type-neutral only: drops 15%
  - Type-neutral & Type-specific: drops 5%

N-gram Combining

- Current scoring favors longer or shorter spans?

N-gram Combining

- Current scoring favors longer or shorter spans?
  - E.g. Roger or Bannister or Roger Bannister or Mr.....

N-gram Combining

- Current scoring favors longer or shorter spans?
  - E.g. Roger or Bannister or Roger Bannister or Mr.....
    - Bannister pry highest – occurs everywhere R.B. +
- Generally, good answers longer (up to a point)

N-gram Combining

- Current scoring favors longer or shorter spans?
  - E.g. Roger or Bannister or Roger Bannister or Mr.....
    - Bannister pry highest – occurs everywhere R.B. +
- Generally, good answers longer (up to a point)
- Update score: S_c += ΣS_t, where t is unigram in c
- Possible issues:

N-gram Combining

- Current scoring favors longer or shorter spans?
  - E.g. Roger or Bannister or Roger Bannister or Mr.....
    - Bannister pry highest – occurs everywhere R.B. +
- Generally, good answers longer (up to a point)
- Update score: S_c += ΣS_t, where t is unigram in c
- Possible issues:
  - Bad units: Roger Bannister was

N-gram Combining

- Current scoring favors longer or shorter spans?
  - E.g. Roger or Bannister or Roger Bannister or Mr.....
    - Bannister pry highest – occurs everywhere R.B. +
- Generally, good answers longer (up to a point)
- Update score: S_c += ΣS_t, where t is unigram in c
- Possible issues:
  - Bad units: Roger Bannister was – blocked by filters
    - Also, increments score so long bad spans lower
- Improves significantly

N-gram Scoring

- Not all terms created equal

N-gram Scoring

- Not all terms created equal
  - Usually answers highly specific
  - Also disprefer non-units
- Solution

N-gram Scoring

- Not all terms created equal
  - Usually answers highly specific
  - Also disprefer non-units
- Solution: IDF-based scoring

  $S_c = S_c$ * average_unigram_idf

N-gram Scoring

- Not all terms created equal
  - Usually answers highly specific
  - Also disprefer non-units
- Solution: IDF-based scoring

  $S_c = S_c$ * average_unigram_idf

After combining

Candidate	Score
Roger Bannister	354
Sir Roger Gilbert Bannister	286
Sir Roger Bannister	280
Bannister Sir Roger	278
...	...

N-gram Scoring

- Not all terms created equal
  - Usually answers highly specific
  - Also disprefer non-units
- Solution: IDF-based scoring

  $S_c = S_c$ * average_unigram_idf

After combining

Candidate	Score
Roger Bannister	354
Sir Roger Gilbert Bannister	286
Sir Roger Bannister	280
Bannister Sir Roger	278
...	...

After scoring

Candidate	Score
Roger Bannister	2377
Englishman Roger Bannister	1853
Sir Roger Gilbert Bannister	1775
Sir Roger Bannister	1768
...	...

N-gram Reranking

- Promote best answer candidates:

N-gram Reranking

- Promote best answer candidates:
  - Filter any answers not in at least two snippets

N-gram Reranking

- Promote best answer candidates:
  - Filter any answers not in at least two snippets
  - Use answer type specific forms to raise matches
    - E.g. ‘where’ -> boosts ‘city, state’
- Small improvement depending on answer type

Summary

- Redundancy-based approaches
  - Leverage scale of web search
  - Take advantage of presence of ‘easy’ answers on web
  - Exploit statistical association of question/answer text

Summary

- Redundancy-based approaches
  - Leverage scale of web search
  - Take advantage of presence of 'easy' answers on web
  - Exploit statistical association of question/answer text
- Increasingly adopted:
  - Good performers independently for QA
  - Provide significant improvements in other systems
    - Esp. for answer filtering

Summary

- Redundancy-based approaches
  - Leverage scale of web search
  - Take advantage of presence of 'easy' answers on web
  - Exploit statistical association of question/answer text
- Increasingly adopted:
  - Good performers independently for QA
  - Provide significant improvements in other systems
    - Esp. for answer filtering
- Does require some form of 'answer projection'
  - Map web information to TREC document

Summary

- Redundancy-based approaches
  - Leverage scale of web search
  - Take advantage of presence of 'easy' answers on web
  - Exploit statistical association of question/answer text
- Increasingly adopted:
  - Good performers independently for QA
  - Provide significant improvements in other systems
    - Esp. for answer filtering
- Does require some form of 'answer projection'
  - Map web information to TREC document
- Aranea download:
  - http://www.umiacs.umd.edu/~jimmylin/resources.html

Deliverable #2: Due 4/19

- Baseline end-to-end Q/A system:
  - Redundancy-based with answer projection

  also viewed as
  - Retrieval with web-based boosting

- Implementation: Main components
  - Basic redundancy approach
  - Basic retrieval approach (IR next lecture)

Data

- Questions:
  - XML formatted questions and question series
- Answers:
  - Answer ‘patterns’ with evidence documents
- Training/Devtext/Evaltest:
  - Training: Thru 2005
  - Devtest: 2006
  - Held-out: ...
- Will be in /dropbox directory on patas
- Documents:
  - AQUAINT news corpus data with minimal markup