Question Classification

Ling573
NLP Systems and Applications
April 22, 2014

Roadmap

- Question classification variations:
  - Classification with diverse features
  - SVM classifiers
  - Sequence classifiers

Question Classification: Li&Roth

Why Question Classification?

Why Question Classification?

- Question classification categorizes possible answers

Why Question Classification?

- Question classification categorizes possible answers
  - Constrains answers types to help find, verify answer

    Q: What Canadian city has the largest population?

    - Type?

Why Question Classification?

- Question classification categorizes possible answers
  - Constrains answers types to help find, verify answer

    Q: What Canadian city has the largest population?
    - Type? → City
    - Can ignore all non-city NPs

Why Question Classification?

- Question classification categorizes possible answers
  - Constrains answers types to help find, verify answer

    Q: What Canadian city has the largest population?
    - Type? → City
    - Can ignore all non-city NPs

  - Provides information for type-specific answer selection
    - *Q: What is a prism?*
    - Type? →

Why Question Classification?

- Question classification categorizes possible answers
  - Constrains answers types to help find, verify answer

    Q: What Canadian city has the largest population?
    - Type? → City
    - Can ignore all non-city NPs

  - Provides information for type-specific answer selection
    - *Q: What is a prism?*
    - Type? → Definition
      - Answer patterns include: 'A prism is…'

Challenges

Challenges

- Variability:
  - What tourist attractions are there in Reims?
  - What are the names of the tourist attractions in Reims?
  - What is worth seeing in Reims?
    - Type?

Challenges

- Variability:
  - What tourist attractions are there in Reims?
  - What are the names of the tourist attractions in Reims?
  - What is worth seeing in Reims?
    - Type? → Location

Challenges

- Variability:
  - What tourist attractions are there in Reims?
  - What are the names of the tourist attractions in Reims?
  - What is worth seeing in Reims?
    - Type? → Location
- Manual rules?

Challenges

- Variability:
  - What tourist attractions are there in Reims?
  - What are the names of the tourist attractions in Reims?
  - What is worth seeing in Reims?
    - Type? → Location
- Manual rules?
  - Nearly impossible to create sufficient patterns
- Solution?

Challenges

- Variability:
  - What tourist attractions are there in Reims?
  - What are the names of the tourist attractions in Reims?
  - What is worth seeing in Reims?
    - Type? → Location
- Manual rules?
  - Nearly impossible to create sufficient patterns
- Solution?
  - Machine learning – rich feature set

Approach

- Employ machine learning to categorize by answer type
  - Hierarchical classifier on semantic hierarchy of types
    - Coarse vs fine-grained
      - Up to 50 classes
  - Differs from text categorization?

Approach

- Employ machine learning to categorize by answer type
  - Hierarchical classifier on semantic hierarchy of types
    - Coarse vs fine-grained
      - Up to 50 classes
  - Differs from text categorization?
    - Shorter (much!)
    - Less information, but
    - Deep analysis more tractable

Approach

- Exploit syntactic and semantic information
  - Diverse semantic resources

Approach

- Exploit syntactic and semantic information
  - Diverse semantic resources
    - Named Entity categories
    - WordNet sense
    - Manually constructed word lists
    - Automatically extracted semantically similar word lists

Approach

- Exploit syntactic and semantic information
  - Diverse semantic resources
    - Named Entity categories
    - WordNet sense
    - Manually constructed word lists
    - Automatically extracted semantically similar word lists
- Results:
  - Coarse: 92.5%; Fine: 89.3%
  - Semantic features reduce error by 28%

Question Hierarchy

Class	#	Class	#
ABBREVIATION	18	term	19
abbreviation	2	vehicle	7
expression	16	word	0
DESCRIPTION	153	**HUMAN**	171
definition	126	group	24
description	13	individual	140
manner	7	title	4
reason	7	description	3
ENTITY	174	**LOCATION**	195
animal	27	city	44
body	5	country	21
color	12	mountain	5
creative	14	other	114
currency	8	state	11
disease/medicine	3	**NUMERIC**	289
event	6	code	1
food	7	count	22
instrument	1	date	146
lang	3	distance	38
letter	0	money	9
other	19	order	0
plant	7	other	24
product	9	period	18
religion	1	percent	7
sport	3	speed	9
substance	20	temp	7
symbol	2	vol.size	4
technique	1	weight	4

Learning a Hierarchical Question Classifier

- Many manual approaches use only :

Learning a Hierarchical Question Classifier

- Many manual approaches use only :
  - Small set of entity types, set of handcrafted rules

Learning a Hierarchical Question Classifier

- Many manual approaches use only :
  - Small set of entity types, set of handcrafted rules
    - Note: Webclopedia's 96 node taxo w/276 manual rules

Learning a Hierarchical Question Classifier

- Many manual approaches use only :
  - Small set of entity types, set of handcrafted rules
    - Note: Webclopedia's 96 node taxo w/276 manual rules
- Learning approaches can learn to generalize
  - Train on new taxonomy, but

Learning a Hierarchical Question Classifier

- Many manual approaches use only :
  - Small set of entity types, set of handcrafted rules
    - Note: Webclopedia's 96 node taxo w/276 manual rules
- Learning approaches can learn to generalize
  - Train on new taxonomy, but
    - Someone still has to label the data…
- Two step learning: (Winnow)
  - Same features in both cases

Learning a Hierarchical Question Classifier

- Many manual approaches use only :
  - Small set of entity types, set of handcrafted rules
    - Note: Webclopedia's 96 node taxo w/276 manual rules
- Learning approaches can learn to generalize
  - Train on new taxonomy, but
    - Someone still has to label the data...
- Two step learning: (Winnow)
  - Same features in both cases
    - First classifier produces (a set of) coarse labels
    - Second classifier selects from fine-grained children of coarse tags generated by the previous stage
    - Select highest density classes above threshold

Features for Question Classification

- Primitive lexical, syntactic, lexical-semantic features
  - Automatically derived
  - Combined into conjunctive, relational features
  - Sparse, binary representation

Features for Question Classification

- Primitive lexical, syntactic, lexical-semantic features
  - Automatically derived
  - Combined into conjunctive, relational features
  - Sparse, binary representation
- Words
  - Combined into ngrams

Features for Question Classification

- Primitive lexical, syntactic, lexical-semantic features
  - Automatically derived
  - Combined into conjunctive, relational features
  - Sparse, binary representation
- Words
  - Combined into ngrams
- Syntactic features:
  - Part-of-speech tags
  - Chunks
  - Head chunks : 1st N, V chunks after Q-word

Syntactic Feature Example

- Q: Who was the first woman killed in the Vietnam War?

Syntactic Feature Example

- Q: Who was the first woman killed in the Vietnam War?
- POS: [Who WP] [was VBD] [the DT] [first JJ] [woman NN] [killed VBN] [in IN] [the DT] [Vietnam NNP] [War NNP] [? .]

Syntactic Feature Example

- Q: Who was the first woman killed in the Vietnam War?
- POS: [Who WP] [was VBD] [the DT] [first JJ] [woman NN] [killed VBN] {in IN] [the DT] [Vietnam NNP] [War NNP] [? .]
- Chunking: [NP Who] [VP was] [NP the first woman] [VP killed] [PP in] [NP the Vietnam War] ?

Syntactic Feature Example

- Q: Who was the first woman killed in the Vietnam War?
- POS: [Who WP] [was VBD] [the DT] [first JJ] [woman NN] [killed VBN] {in IN] [the DT] [Vietnam NNP] [War NNP] [? .]
- Chunking: [NP Who] [VP was] [NP the first woman] [VP killed] [PP in] [NP the Vietnam War] ?
- Head noun chunk: ‘the first woman’

Semantic Features

- Treat analogously to syntax?

Semantic Features

- Treat analogously to syntax?
  - Q1:What's the semantic equivalent of POS tagging?

Semantic Features

- Treat analogously to syntax?
  - Q1:What's the semantic equivalent of POS tagging?
  - Q2: POS tagging > 97% accurate;
    - Semantics? Semantic ambiguity?

Semantic Features

- Treat analogously to syntax?
  - Q1:What's the semantic equivalent of POS tagging?
  - Q2: POS tagging > 97% accurate;
    - Semantics? Semantic ambiguity?

- A1: Explore different lexical semantic info sources
    - Differ in granularity, difficulty, and accuracy

Semantic Features

- Treat analogously to syntax?
  - Q1:What's the semantic equivalent of POS tagging?
  - Q2: POS tagging > 97% accurate;
    - Semantics? Semantic ambiguity?

- A1: Explore different lexical semantic info sources
    - Differ in granularity, difficulty, and accuracy
  - Named Entities
  - WordNet Senses
  - Manual word lists
  - Distributional sense clusters

Tagging & Ambiguity

- Augment each word with semantic category

- What about ambiguity?
  - E.g. ‘water’ as ‘liquid’ or ‘body of water’

Tagging & Ambiguity

- Augment each word with semantic category

- What about ambiguity?
  - E.g. ‘water’ as ‘liquid’ or ‘body of water’
  - Don’t disambiguate
    - Keep all alternatives
    - Let the learning algorithm sort it out
    - Why?

Semantic Categories

- Named Entities
  - Expanded class set: 34 categories
    - E.g. Profession, event, holiday, plant,...

Semantic Categories

- Named Entities
  - Expanded class set: 34 categories
    - E.g. Profession, event, holiday, plant,...
- WordNet: IS-A hierarchy of senses
  - All senses of word + direct hyper/hyponyms

Semantic Categories

- Named Entities
  - Expanded class set: 34 categories
    - E.g. Profession, event, holiday, plant,...
- WordNet: IS-A hierarchy of senses
  - All senses of word + direct hyper/hyponyms
- Class-specific words
  - Manually derived from 5500 questions
    - E.g. Class: Food
      - {alcoholic, apple, beer, berry, breakfast brew butter candy cereal champagne cook delicious eat fat ..}
      - Class is semantic tag for word in the list

Semantic Types

- Distributional clusters:
  - Based on Pantel and Lin
  - Cluster based on similarity in dependency relations
  - Word lists for 20K English words

Semantic Types

- Distributional clusters:
  - Based on Pantel and Lin
  - Cluster based on similarity in dependency relations
  - Word lists for 20K English words
    - Lists correspond to word senses
    - Water:
      - Sense 1: { oil gas fuel food milk liquid}
      - Sense 2: {air moisture soil heat area rain}
      - Sense 3: {waste sewage pollution runoff}

Semantic Types

- Distributional clusters:
  - Based on Pantel and Lin
  - Cluster based on similarity in dependency relations
  - Word lists for 20K English words
    - Lists correspond to word senses
    - Water:
      - Sense 1: { oil gas fuel food milk liquid}
      - Sense 2: {air moisture soil heat area rain}
      - Sense 3: {waste sewage pollution runoff}
  - Treat head word as semantic category of words on list

Evaluation

- Assess hierarchical coarse->fine classification
- Assess impact of different semantic features
- Assess training requirements for diff't feature set

Evaluation

- Assess hierarchical coarse->fine classification
- Assess impact of different semantic features
- Assess training requirements for diff't feature set
- Training:
  - 21.5K questions from TREC 8,9; manual; USC data
- Test:
  - 1K questions from TREC 10,11

Evaluation

- Assess hierarchical coarse->fine classification
- Assess impact of different semantic features
- Assess training requirements for diff't feature set
- Training:
  - 21.5K questions from TREC 8,9; manual; USC data
- Test:
  - 1K questions from TREC 10,11
- Measures: Accuracy and class-specific precision

Results

- Syntactic features only:

Classifier	Word	POS	Chunk	Head(SYN)
Coarse	85.10	91.80	91.80	**92.50**
Fine	82.60	84.90	84.00	**85.00**

  - POS useful; chunks useful to contribute head chunks
  - Fine categories more ambiguous

Results

- Syntactic features only:

Classifier	Word	POS	Chunk	Head(SYN)
Coarse	85.10	91.80	91.80	**92.50**
Fine	82.60	84.90	84.00	**85.00**

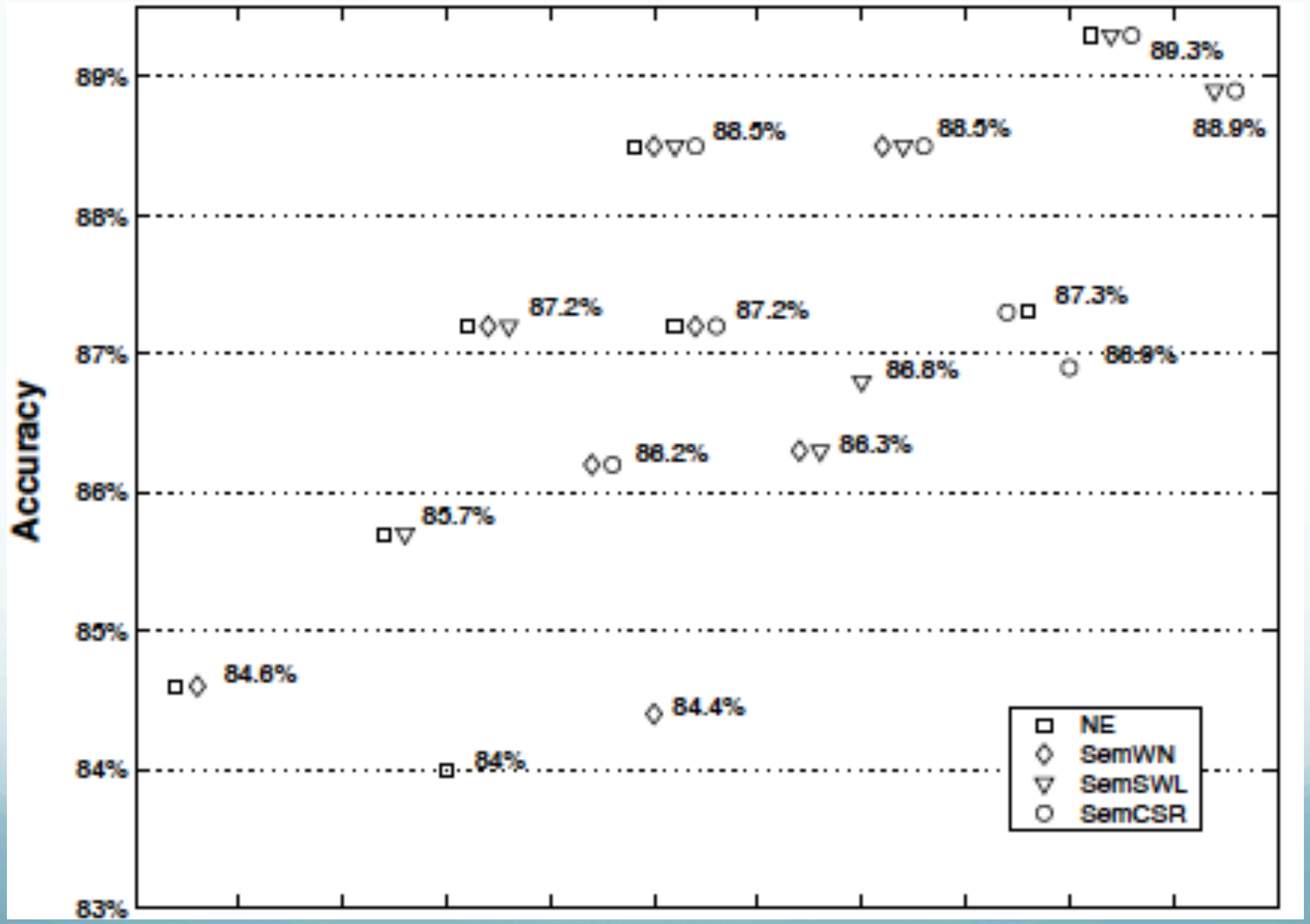
  - POS useful; chunks useful to contribute head chunks
  - Fine categories more ambiguous
- Semantic features:
  - Best combination: SYN, NE, Manual & Auto word lists
    - Coarse: same; Fine: 89.3% (28.7% error reduction)

Results

- Syntactic features only:

Classifier	Word	POS	Chunk	Head(SYN)
Coarse	85.10	91.80	91.80	**92.50**
Fine	82.60	84.90	84.00	**85.00**

  - POS useful; chunks useful to contribute head chunks
  - Fine categories more ambiguous
- Semantic features:
  - Best combination: SYN, NE, Manual & Auto word lists
    - Coarse: same; Fine: 89.3% (28.7% error reduction)
- Wh-word most common class: 41%


Accuracy
89%
88%
87%
86%
85%
84%
83%
89.3%
88.9%
88.5%
88.5%
87.2%
87.2%
87.3%
86.9%
86.8%
86.2%
86.3%
85.7%
84.6%
84.4%
84%
NE
SemWN
SemSWL
SemCSR

Class	#	Precision[c]	Class	#	Precision[c]
abb	2	100%	desc	25	36%
exp	17	94.11%	manner	8	87.5%
animal	27	85.18%	reason	7	85.71%
body	4	100%	gr	19	89.47%
color	12	100%	ind	154	90.25%
cremat	13	76.92%	title	4	100%
currency	6	100%	desc	3	100%
dismed	4	50%	city	41	97.56%
event	4	75%	country	21	95.23%
food	6	100%	mount	2	100%
instru	1	100%	LOC:other	116	89.65%
lang	3	100%	state	14	78.57%
ENTY:other	24	37.5%	count	24	91.66%
plant	3	100%	date	145	100%
product	6	66.66%	dist	37	97.29%
religion	1	100%	money	6	100%
sport	4	75%	NUM:other	15	93.33%
substance	21	80.95%	period	20	85%
symbol	2	100%	perc	9	77.77%
termeq	22	63.63%	speed	8	100%
veh	7	71.42%	temp	4	100%
def	125	97.6%	weight	4	100%
TOTAL	**1000**	**89.3%**			

Observations

- Effective coarse and fine-grained categorization
  - Mix of information sources and learning
  - Shallow syntactic features effective for coarse
  - Semantic features improve fine-grained
    - Most feature types help
      - WordNet features appear noisy
      - Use of distributional sense clusters dramatically increases feature dimensionality

NE	0.23
SemWN	16
SemCSR	23
SemSWL	557

Question Classification with Support Vector Machines

- Hacioglu & Ward 2003
- Same taxonomy, training, test data as Li & Roth

Question Classification with Support Vector Machines

- Hacioglu & Ward 2003
- Same taxonomy, training, test data as Li & Roth
- Approach:
  - Shallow processing
  - Simpler features
  - Strong discriminative classifiers

Features & Processing

- Contrast: (Li & Roth)
  - POS, chunk info; NE tagging; other sense info

Features & Processing

- Contrast: (Li & Roth)
  - POS, chunk info; NE tagging; other sense info
- Preprocessing:
  - Only letters, convert to lower case, stopped, stemmed

Features & Processing

- Contrast: (Li & Roth)
  - POS, chunk info; NE tagging; other sense info
- Preprocessing:
  - Only letters, convert to lower case, stopped, stemmed
- Terms:
  - Most informative 2000 word N-grams
  - Identifinder NE tags (7 or 29 tags)

Classification & Results

- Employs support vector machines for classification
  - Best results: Bi-gram, 7 NE classes

Method	1-gram	2-gram	3-gram
No NE	79.4%	80.2% (77.8%)	78.4%
NE-7	81.4%	82.0% (81.2%)	80.2%
NE-29	75.4	78.6% (79.2%)	78.8%

Classification & Results

- Employs support vector machines for classification
  - Best results: Bi-gram, 7 NE classes
    - Fewer NE categories better
      - More categories, more errors

Method	1-gram	2-gram	3-gram
No NE	79.4%	80.2% (77.8%)	78.4%
NE-7	81.4%	82.0% (81.2%)	80.2%
NE-29	75.4	78.6% (79.2%)	78.8%

Enhanced Answer Type Inference … Using Sequential Models

- Krishnan, Das, and Chakrabarti 2005
- Improves QC with CRF extraction of ‘informer spans’

Enhanced Answer Type Inference … Using Sequential Models

- Krishnan, Das, and Chakrabarti 2005
- Improves QC with CRF extraction of ‘informer spans’
- Intuition:
  - Humans identify Atype from few tokens w/little syntax

Enhanced Answer Type Inference … Using Sequential Models

- Krishnan, Das, and Chakrabarti 2005
- Improves QC with CRF extraction of ‘informer spans’
- Intuition:
  - Humans identify Atype from few tokens w/little syntax
    - Who wrote Hamlet?

Enhanced Answer Type Inference … Using Sequential Models

- Krishnan, Das, and Chakrabarti 2005
- Improves QC with CRF extraction of 'informer spans'
- Intuition:
  - Humans identify Atype from few tokens w/little syntax
    - **Who** wrote Hamlet?

Enhanced Answer Type Inference … Using Sequential Models

- Krishnan, Das, and Chakrabarti 2005
- Improves QC with CRF extraction of ‘informer spans’
- Intuition:
  - Humans identify Atype from few tokens w/little syntax
    - Who wrote Hamlet?
    - How many dogs pull a sled at Iditarod?

Enhanced Answer Type Inference … Using Sequential Models

- Krishnan, Das, and Chakrabarti 2005
- Improves QC with CRF extraction of 'informer spans'
- Intuition:
  - Humans identify Atype from few tokens w/little syntax
    - Who wrote Hamlet?
    - **How many** dogs pull a sled at Iditarod?

Enhanced Answer Type Inference … Using Sequential Models

- Krishnan, Das, and Chakrabarti 2005
- Improves QC with CRF extraction of 'informer spans'
- Intuition:
  - Humans identify Atype from few tokens w/little syntax
    - Who wrote Hamlet?
    - How many dogs pull a sled at Iditarod?
    - How much does a rhino weigh?

Enhanced Answer Type Inference … Using Sequential Models

- Krishnan, Das, and Chakrabarti 2005
- Improves QC with CRF extraction of 'informer spans'
- Intuition:
  - Humans identify Atype from few tokens w/little syntax
    - Who wrote Hamlet?
    - How many dogs pull a sled at Iditarod?
    - How much does a rhino **weigh**?

Enhanced Answer Type Inference … Using Sequential Models

- Krishnan, Das, and Chakrabarti 2005
- Improves QC with CRF extraction of 'informer spans'
- Intuition:
  - Humans identify Atype from few tokens w/little syntax
    - Who wrote Hamlet?
    - How many dogs pull a sled at Iditarod?
    - How much does a rhino weigh?
  - Single contiguous span of tokens

Enhanced Answer Type Inference … Using Sequential Models

- Krishnan, Das, and Chakrabarti 2005
- Improves QC with CRF extraction of 'informer spans'
- Intuition:
  - Humans identify Atype from few tokens w/little syntax
    - Who wrote Hamlet?
    - How many dogs pull a sled at Iditarod?
    - How much does a rhino weigh?
  - Single contiguous span of tokens
    - *How much* does a rhino **weigh?**

Enhanced Answer Type Inference … Using Sequential Models

- Krishnan, Das, and Chakrabarti 2005
- Improves QC with CRF extraction of 'informer spans'
- Intuition:
  - Humans identify Atype from few tokens w/little syntax
    - Who wrote Hamlet?
    - How many dogs pull a sled at Iditarod?
    - How much does a rhino weigh?
  - Single contiguous span of tokens
    - How much does a rhino weigh?
    - *Who* is the **CEO** of IBM?

Informer Spans as Features

- Sensitive to question structure
  - What is Bill Clinton's wife's profession?

Informer Spans as Features

- Sensitive to question structure
  - What is Bill Clinton's wife's **profession**?

Informer Spans as Features

- Sensitive to question structure
  - What is Bill Clinton's wife's profession?
- Idea: Augment Q classifier word ngrams w/IS info

Informer Spans as Features

- Sensitive to question structure
  - What is Bill Clinton's wife's profession?
- Idea: Augment Q classifier word ngrams w/IS info
- Informer span features:
  - IS ngrams

Informer Spans as Features

- Sensitive to question structure
  - What is Bill Clinton's wife's profession?
- Idea: Augment Q classifier word ngrams w/IS info
- Informer span features:
  - IS ngrams
  - Informer ngrams hypernyms:
    - Generalize over words or compounds

Informer Spans as Features

- Sensitive to question structure
  - What is Bill Clinton's wife's profession?
- Idea: Augment Q classifier word ngrams w/IS info
- Informer span features:
  - IS ngrams
  - Informer ngrams hypernyms:
    - Generalize over words or compounds
    - WSD?

Informer Spans as Features

- Sensitive to question structure
  - What is Bill Clinton's wife's profession?
- Idea: Augment Q classifier word ngrams w/IS info
- Informer span features:
  - IS ngrams
  - Informer ngrams hypernyms:
    - Generalize over words or compounds
    - WSD? No

Effect of Informer Spans

- Classifier: Linear SVM + multiclass

Features	Coarse	Fine
Question trigrams	91.2	77.6
All question *q*grams	87.2	71.8
All question unigrams	88.4	78.2
Question bigrams	91.6	79.4
+informer q-grams	94.0	82.4
+informer hypernyms	**94.2**	**88.0**
Question unigrams + all informer	93.4	88.0
Only informer	92.2	85.0
Question bigrams + hypernyms	91.6	79.4

Effect of Informer Spans

- Classifier: Linear SVM + multiclass
  - Notable improvement for IS hypernyms

Features	Coarse	Fine
Question trigrams	91.2	77.6
All question *q*grams	87.2	71.8
All question unigrams	88.4	78.2
Question bigrams	91.6	79.4
+informer q-grams	94.0	82.4
+informer hypernyms	**94.2**	**88.0**
Question unigrams + all informer	93.4	88.0
Only informer	92.2	85.0
Question bigrams + hypernyms	91.6	79.4

Effect of Informer Spans

- Classifier: Linear SVM + multiclass
  - Notable improvement for IS hypernyms
    - Better than all hypernyms – filter sources of noise
  - Biggest improvements for 'what', 'which' questions

Features	Coarse	Fine
Question trigrams	91.2	77.6
All question *q*grams	87.2	71.8
All question unigrams	88.4	78.2
Question bigrams	91.6	79.4
+informer q-grams	94.0	82.4
+informer hypernyms	**94.2**	**88.0**
Question unigrams + all informer	93.4	88.0
Only informer	92.2	85.0
Question bigrams + hypernyms	91.6	79.4

Perfect vs CRF Informer Spans

Type	#Quest.	B (Bigrams)	Only Informers Perf.Inf	H.Inf	CRF.Inf	B+ Perf.Inf	B+ H.Inf	B+ CRF.Inf
what	349	**88.8**	89.4	69.6	79.3	**91.7**	87.4	91.4
which	11	**72.7**	100.0	45.4	81.8	**100.0**	63.6	81.8
when	28	100.0	100.0	100.0	100.0	100.0	100.0	100.0
where	27	100.0	96.3	100.0	96.3	100.0	100.0	100.0
who	47	100.0	100.0	100.0	100.0	100.0	100.0	100.0
how_*	32	100.0	96.9	100.0	100.0	100.0	100.0	100.0
rest	6	100.0	100.0	100.0	66.7	100.0	66.7	66.7
Total	500	91.6	92.2	*77.2*	84.6	94.2	*90.0*	93.4
50 fine classes								
what	349	**73.6**	82.2	61.9	78.0	**85.1**	79.1	83.1
which	11	**81.8**	90.9	45.4	73.1	**90.9**	54.5	81.8
when	28	100.0	100.0	100.0	100.0	100.0	100.0	100.0
where	27	92.6	85.2	92.6	88.9	88.9	92.5	88.9
who	47	97.9	93.6	93.6	93.6	100.0	100.0	97.9
how_*	32	87.5	84.3	81.2	78.1	87.5	90.6	90.6
rest	6	66.7	66.7	66.7	66.7	100.0	66.7	66.7
Total	500	79.4	85.0	*69.6*	78.0	88.0	*82.6*	86.2

Recognizing Informer Spans

- Idea: contiguous spans, syntactically governed

Recognizing Informer Spans

- Idea: contiguous spans, syntactically governed
  - Use sequential learner w/syntactic information

Recognizing Informer Spans

- Idea: contiguous spans, syntactically governed
  - Use sequential learner w/syntactic information
- Tag spans with B(egin),I(nside),O(outside)
  - Employ syntax to capture long range factors

Recognizing Informer Spans

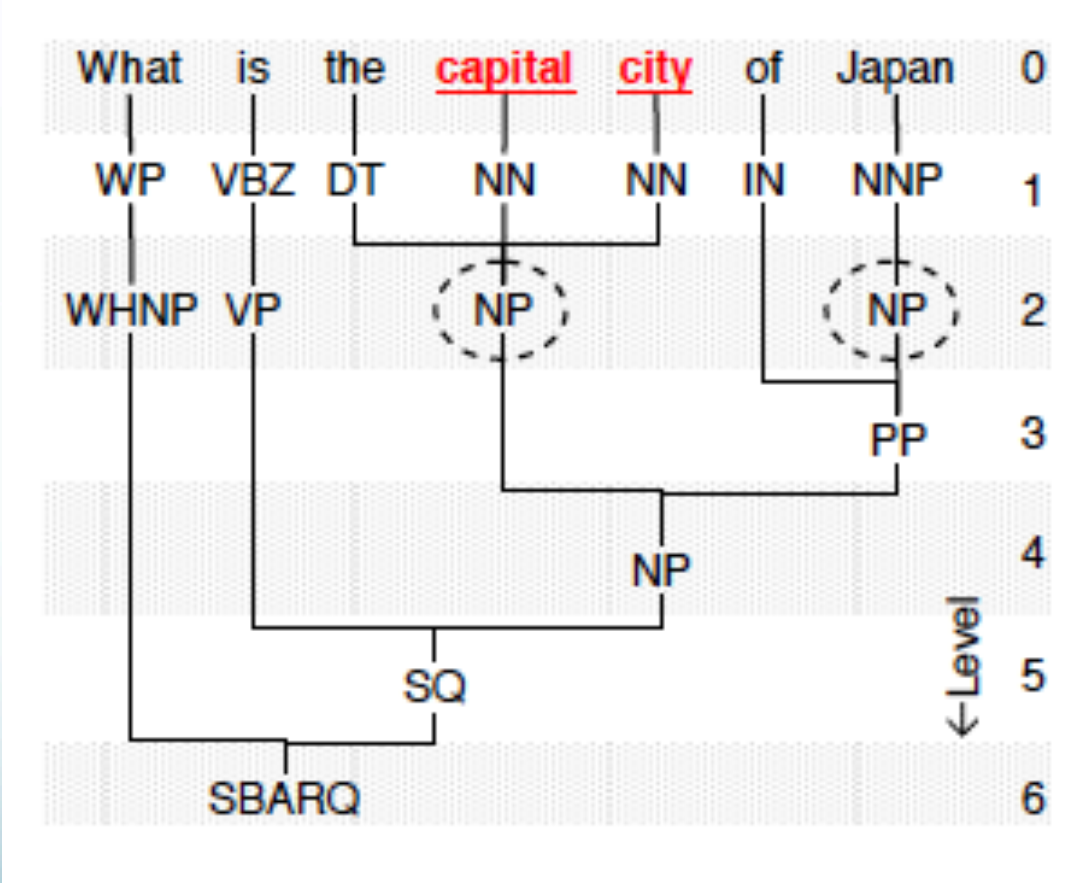
- Idea: contiguous spans, syntactically governed
  - Use sequential learner w/syntactic information
- Tag spans with B(egin),I(nside),O(outside)
  - Employ syntax to capture long range factors
- Matrix of features derived from parse tree

Recognizing Informer Spans

- Idea: contiguous spans, syntactically governed
  - Use sequential learner w/syntactic information
- Tag spans with B(egin),I(nside),O(outside)
  - Employ syntax to capture long range factors
- Matrix of features derived from parse tree
  - Cell:x[i,l], i is position, l is depth in parse tree, only 2
  - Values:
    - Tag: POS, constituent label in the position
    - Num: number of preceding chunks with same tag

Parser Output

- Parse



Parse Tabulation

- Encoding and table:

i	1	2	3	4	5	6	7
y_i	0	0	0	1	1	2	2
x_i	What	is	the	capital	city	of	Japan
$\ell \downarrow$	Features for x_is						
1	WP,1	VBZ,1	DT,1	NN,1	NN,1	IN,1	NNP,1
2	WHNP,1	VP,1	NP,1	**NP**,1	NP,1	Null,1	**NP,2**
3	Null,1	Null,1	Null,1	Null,1	Null,1	PP,1	PP,1
4	Null,1	Null,1	NP,1	NP,1	NP,1	NP,1	NP,1
5	Null,1	SQ,1	SQ,1	SQ,1	SQ,1	SQ,1	SQ,1
6	SBARQ	SBARQ	SBARQ	SBARQ	SBARQ	SBARQ	SBARQ

CRF Indicator Features

- Cell:
  - IsTag, IsNum: e.g. $y_4 = 1$ and x[4,2].tag=NP
  - Also, IsPrevTag, IsNextTag

CRF Indicator Features

- Cell:
  - IsTag, IsNum: e.g. $y_4 = 1$ and x[4,2].tag=NP
  - Also, IsPrevTag, IsNextTag
- Edge:
  - IsEdge: (u,v) , y_{i-1}=u and y_i=v
  - IsBegin, IsEnd

CRF Indicator Features

- Cell:
  - IsTag, IsNum: e.g. $y_4 = 1$ and x[4,2].tag=NP
  - Also, IsPrevTag, IsNextTag
- Edge:
  - IsEdge: (u,v) , y_{i-1}=u and y_i=v
  - IsBegin, IsEnd
- All features improve

IsTag	0.368
+IsNum	0.474
+IsPrevTag+IsNextTag	0.692
+IsEdge+IsBegin+IsEnd	**0.848**

CRF Indicator Features

- Cell:
  - IsTag, IsNum: e.g. $y_4 = 1$ and x[4,2].tag=NP
  - Also, IsPrevTag, IsNextTag
- Edge:
  - IsEdge: (u,v) , y_{i-1}=u and y_i=v
  - IsBegin, IsEnd
- All features improve

IsTag	0.368
+IsNum	0.474
+IsPrevTag+IsNextTag	0.692
+IsEdge+IsBegin+IsEnd	**0.848**

- Question accuracy: Oracle: 88%; CRF: 86.2%