Question Classification II

Ling573
NLP Systems and Applications
April 30, 2013

Roadmap

- Question classification variations:
  - SVM classifiers
  - Sequence classifiers
  - Sense information improvements
- Question series

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 200
- Same taxonomy, training, test data as Li & Roth
- Approach:
  - Shallow processing
  - Simpler features
  - Strong discriminative classifiers

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 9 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%	<u>82.0%</u> (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
    - Better than Li & Roth w/POS+chunk, but no semantics

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
    - Better than Li & Roth w/POS+chunk, but no semantics
    - 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

		B	Only Informers			B+	B+	B+
Type	#Quest.	(Bigrams)	Perf.Inf	H.Inf	CRF.Inf	Perf.Inf	H.Inf	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

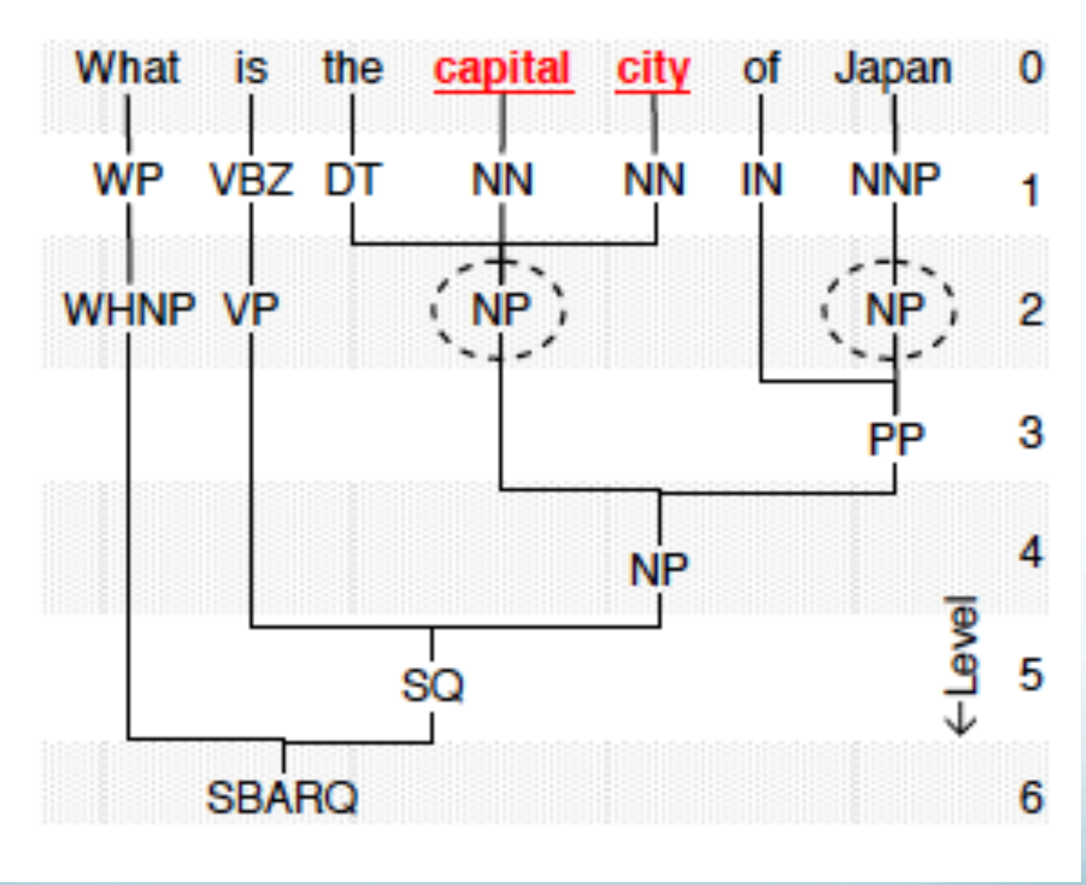
- Idea: contiguous spans, syntactically governed
  - Use sequential learner w/syntactic information
- Tag spans with B(egin),I(nside),O(outside)
  - Employ syntax to capture long range factors
- Matrix of features derived from parse tree

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%

Question Classification Using Headwords and Their Hypernyms

- Huang, Thint, and Qin 2008
- Questions:
  - Why didn't WordNet/Hypernym features help in L&R?

Question Classification Using Headwords and Their Hypernyms

- Huang, Thint, and Qin 2008
- Questions:
  - Why didn't WordNet/Hypernym features help in L&R?
  - Best results in L&R - ~200,000 feats; ~700 active
    - Can we do as well with fewer features?

Question Classification Using Headwords and Their Hypernyms

- Huang, Thint, and Qin 2008
- Questions:
  - Why didn't WordNet/Hypernym features help in L&R?
  - Best results in L&R - ~200,000 feats; ~700 active
    - Can we do as well with fewer features?
- Approach:
  - Refine features:

Question Classification Using Headwords and Their Hypernyms

- Huang, Thint, and Qin 2008
- Questions:
  - Why didn't WordNet/Hypernym features help in L&R?
  - Best results in L&R - ~200,000 feats; ~700 active
    - Can we do as well with fewer features?
- Approach:
  - Refine features:
    - Restrict use of WordNet to headwords

Question Classification Using Headwords and Their Hypernyms

- Huang, Thint, and Qin 2008
- Questions:
  - Why didn't WordNet/Hypernym features help in L&R?
  - Best results in L&R - ~200,000 feats; ~700 active
    - Can we do as well with fewer features?
- Approach:
  - Refine features:
    - Restrict use of WordNet to headwords
    - Employ WSD techniques
  - SVM, MaxEnt classifiers

Head Word Features

- Head words:
  - Chunks and spans can be noisy

Head Word Features

- Head words:
  - Chunks and spans can be noisy
    - E.g. Bought a share in *which baseball team*?

Head Word Features

- Head words:
  - Chunks and spans can be noisy
    - E.g. Bought a share in *which baseball team*?
      - Type: HUM: group (not ENTY:sport)
      - Head word is more specific

Head Word Features

- Head words:
  - Chunks and spans can be noisy
    - E.g. Bought a share in *which baseball team*?
      - Type: HUM: group (not ENTY:sport)
      - Head word is more specific
  - Employ rules over parse trees to extract head words

Head Word Features

- Head words:
  - Chunks and spans can be noisy
    - E.g. Bought a share in *which baseball team*?
      - Type: HUM: group (not ENTY:sport)
      - Head word is more specific
  - Employ rules over parse trees to extract head words
  - Issue: vague heads
    - E.g. What is the proper name for a female walrus?
      - Head = ‘name’?

Head Word Features

- Head words:
  - Chunks and spans can be noisy
    - E.g. Bought a share in *which baseball team*?
      - Type: HUM: group (not ENTY:sport)
      - Head word is more specific
  - Employ rules over parse trees to extract head words
  - Issue: vague heads
    - E.g. What is the proper name for a female walrus?
      - Head = 'name'?
  - Apply fix patterns to extract sub-head (e.g. walrus)

Head Word Features

- Head words:
  - Chunks and spans can be noisy
    - E.g. Bought a share in *which baseball team*?
      - Type: HUM: group (not ENTY:sport)
      - Head word is more specific
  - Employ rules over parse trees to extract head words
  - Issue: vague heads
    - E.g. What is the proper name for a female walrus?
      - Head = 'name'?
  - Apply fix patterns to extract sub-head (e.g. walrus)
  - Also, simple regexp for other feature type
    - E.g. 'what is' cue to definition type

WordNet Features

- Hypernyms:
  - Enable generalization: dog->..->animal
  - Can generate noise: also

WordNet Features

- Hypernyms:
  - Enable generalization: dog->..->animal
  - Can generate noise: also dog ->...-> person

WordNet Features

- Hypernyms:
  - Enable generalization: dog->..->animal
  - Can generate noise: also dog ->...-> person
- Adding low noise hypernyms
  - Which senses?

WordNet Features

- Hypernyms:
  - Enable generalization: dog->..->animal
  - Can generate noise: also dog ->…-> person
- Adding low noise hypernyms
  - Which senses?
    - Restrict to matching WordNet POS

WordNet Features

- Hypernyms:
  - Enable generalization: dog->..->animal
  - Can generate noise: also dog ->...-> person
- Adding low noise hypernyms
  - Which senses?
    - Restrict to matching WordNet POS
  - Which word senses?

WordNet Features

- Hypernyms:
  - Enable generalization: dog->..->animal
  - Can generate noise: also dog ->...-> person
- Adding low noise hypernyms
  - Which senses?
    - Restrict to matching WordNet POS
  - Which word senses?
    - Use Lesk algorithm: overlap b/t question & WN gloss

WordNet Features

- Hypernyms:
  - Enable generalization: dog->..->animal
  - Can generate noise: also dog ->...-> person
- Adding low noise hypernyms
  - Which senses?
    - Restrict to matching WordNet POS
  - Which word senses?
    - Use Lesk algorithm: overlap b/t question & WN gloss
  - How deep?

WordNet Features

- Hypernyms:
  - Enable generalization: dog->..->animal
  - Can generate noise: also dog ->…-> person
- Adding low noise hypernyms
  - Which senses?
    - Restrict to matching WordNet POS
  - Which word senses?
    - Use Lesk algorithm: overlap b/t question & WN gloss
  - How deep?
    - Based on validation set: 6

WordNet Features

- Hypernyms:
  - Enable generalization: dog->..->animal
  - Can generate noise: also dog ->...-> person
- Adding low noise hypernyms
  - Which senses?
    - Restrict to matching WordNet POS
  - Which word senses?
    - Use Lesk algorithm: overlap b/t question & WN gloss
  - How deep?
    - Based on validation set: 6
- Q Type similarity: compute similarity b/t headword & type
  - Use type as feature

Other Features

- Question wh-word:
  - What,which,who,where,when,how,why, and rest

Other Features

- Question wh-word:
  - What,which,who,where,when,how,why, and rest
- N-grams: uni-,bi-,tri-grams

Other Features

- Question wh-word:
  - What,which,who,where,when,how,why, and rest
- N-grams: uni-,bi-,tri-grams
- Word shape:
  - Case features: all upper, all lower, mixed, all digit, other

Results

		6 class		50 class	
		SVM	ME	SVM	ME
wh-word + head word		92.0	92.2	81.4	82.0
wh-word + head word + direct hypernym	depth=1	92.0	91.8	84.6	84.8
	depth = 3	92.0	92.2	85.4	85.4
	depth = 6	92.6	91.8	85.4	85.6
wh-word + head + indirect hypernym		91.8	92.0	83.2	83.6
unigram		88.0	86.6	80.4	78.8
bigram		85.6	86.4	73.8	75.2
trigram		68.0	57.4	39.0	44.2
word shape		18.8	18.8	10.4	10.4

Per feature-type results:

Results: Incremental

- Additive improvement:

6 coarse classes										
Type	#Quest	wh+headword		+headword hypernym		+unigram		+word shape		
		SVM	ME	SVM	ME	SVM	ME	SVM	ME	
what	349	88.8	89.1	89.7	88.5	89.7	90.3	90.5	91.1	
which	11	90.9	90.9	100	100	100	100	100	100	
when	26	100	100	100	100	100	100	100	100	
where	27	100	100	100	100	100	100	100	100	
who	47	100	100	100	100	100	100	100	100	
how	34	100	100	100	100	100	100	100	100	
why	4	100	100	100	100	100	100	100	100	
rest	2	100	100	50.0	50.0	100	50.0	100	50.0	
total	500	92.0	92.2	92.6	91.8	92.8	93.0	93.4	93.6	

50 fine classes									
Type	#Quest	wh+headword		+headword hypernym		+unigram		+word shape	
		SVM	ME	SVM	ME	SVM	ME	SVM	ME
what	349	77.4	77.9	82.8	82.5	85.4	85.1	86.2	86.0
which	11	81.8	90.9	81.8	90.9	90.9	100	90.9	100
when	26	100	100	100	100	100	100	100	100
where	27	92.6	92.6	92.6	92.6	92.6	92.6	92.6	92.6
who	47	100	100	100	100	100	100	100	100
how	34	76.5	76.5	76.5	79.4	97.1	91.2	97.1	91.2
why	4	100	100	100	100	100	100	100	100
rest	2	0.0	0.0	50.0	50.0	0.0	50.0	0.0	50.0
total	500	81.4	82.0	85.4	85.6	88.6	88.4	89.2	89.0

Error Analysis

- Inherent ambiguity:
  - What is mad cow disease?
    - ENT: disease or DESC:def

Error Analysis

- Inherent ambiguity:
  - What is mad cow disease?
    - ENT: disease or DESC:def
- Inconsistent labeling:
  - What is the population of Kansas? NUM: other
  - What is the population of Arcadia, FL ?

Error Analysis

- Inherent ambiguity:
  - What is mad cow disease?
    - ENT: disease or DESC:def
- Inconsistent labeling:
  - What is the population of Kansas? NUM: other
  - What is the population of Arcadia, FL ? NUM:count
- Parser error

Question Classification: Summary

- Issue:
  - Integrating rich features/deeper processing

Question Classification: Summary

- Issue:
  - Integrating rich features/deeper processing
    - Errors in processing introduce noise

Question Classification: Summary

- Issue:
  - Integrating rich features/deeper processing
    - Errors in processing introduce noise
    - Noise in added features increases error

Question Classification: Summary

- Issue:
  - Integrating rich features/deeper processing
    - Errors in processing introduce noise
    - Noise in added features increases error
    - Large numbers of features can be problematic for training

Question Classification: Summary

- Issue:
  - Integrating rich features/deeper processing
    - Errors in processing introduce noise
    - Noise in added features increases error
    - Large numbers of features can be problematic for training
- Alternative solutions:

Question Classification: Summary

- Issue:
  - Integrating rich features/deeper processing
    - Errors in processing introduce noise
    - Noise in added features increases error
    - Large numbers of features can be problematic for training
- Alternative solutions:
  - Use more accurate shallow processing, better classifier

Question Classification: Summary

- Issue:
  - Integrating rich features/deeper processing
    - Errors in processing introduce noise
    - Noise in added features increases error
    - Large numbers of features can be problematic for training
- Alternative solutions:
  - Use more accurate shallow processing, better classifier
  - Restrict addition of features to

Question Classification: Summary

- Issue:
  - Integrating rich features/deeper processing
    - Errors in processing introduce noise
    - Noise in added features increases error
    - Large numbers of features can be problematic for training
- Alternative solutions:
  - Use more accurate shallow processing, better classifier
  - Restrict addition of features to
    - Informer spans
    - Headwords

Question Classification: Summary

- Issue:
  - Integrating rich features/deeper processing
    - Errors in processing introduce noise
    - Noise in added features increases error
    - Large numbers of features can be problematic for training
- Alternative solutions:
  - Use more accurate shallow processing, better classifier
  - Restrict addition of features to
    - Informer spans
    - Headwords
      - Filter features to be added