Harvesting Paragraph-Level Question-Answer Pairs from Wikipedia

Xinya Du and **Claire Cardie**

Department of Computer Science

Cornell University

ACL 2018

Nikola Tesla

For other uses, see Nikola Tesla (disambiguation).

Nikola Tesla (Serbian Cyrillic: Никола Тесла; 10 July 1856 – 7 January 1943) was a Serbian-American[3][4][5][6] inventor, electrical engineer, mechanical engineer, physicist, and futurist best known for his contributions to the design of the modern alternating current (AC) electricity supply system.[7]

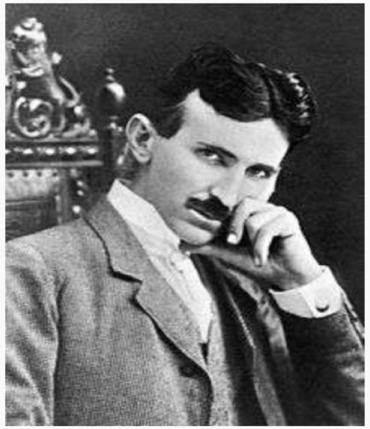
Tesla gained experience in telephony and electrical engineering before emigrating to the United States in 1884 to work for Thomas Edison in New York City. He soon struck out on his own with financial backers, setting up

……

Tesla was renowned for his achievements and showmanship, eventually earning him a reputation in popular culture as an archetypal "mad scientist".[10] His patents earned him a considerable amount of money, much of which was used to finance his own projects with varying degrees of success.[11] He lived most of his life in a series of New York hotels through his retirement. Tesla died on 7 January 1943 in New York City.[12] His work fell into relative obscurity after his

WIKIPEDIA
The Free Encyclopedia

Nikola Tesla



Tesla, circa 1896.

Paragraph:

(1) *Tesla* was renowned for *his* achievements and showmanship, eventually earning *him* a reputation in popular culture as an archetypal "mad scientist". (2) *His* patents earned *him* a considerable amount of money, much of which was used to finance *his* own projects with varying degrees of success. (3) *He* lived most of his life in a series of New York hotels, through *his* retirement. (4) *Tesla* died on 7 January 1943. ...

Questions:

– What was Tesla's reputation in popular culture?

mad scientist

– How did Tesla finance his work?

patents

– Where did Tesla live for much of his life?

New York hotels

Figure 1: Example input from the fourth paragraph of a Wikipedia article on *Nikola Tesla*,

* *QA pairs from* ***SQuAD*** *(Rajpurkar et al., 2016)*

Background: Question Generation

- Sentence-Level Question Generation (text based)
  - Rule-based methods: Rus et al. (2010), Heilman and Smith (2010)
  - NN-based (Seq2seq) methods: Du et al. (2017), Zhou et al. (2017)
  - …

Question: How to generate better questions at **paragraph-level**?

What we found: Leveraging the *coreference* knowledge aids question generation significantly.

Paragraph:

(1)*Tesla* was renowned for *his* achievements and showmanship, eventually earning *him* a reputation in popular culture as an archetypal "mad scientist". (2)*His* patents earned *him* a considerable amount of money, much of which was used to finance *his* own projects with varying degrees of success. (3)*He* lived most of his life in a series of New York hotels, through *his* retirement. (4)*Tesla* died on 7 January 1943. ...

Figure 1: The set of mentions in red all refer to Nikola Tesla — *Tesla*, *him*, *his*, *he*, etc.

Methodology (Answer Span Extraction)

- Formalize as a sequence-labeling task
  - "Extracting" the *question-worthy* concepts/spans.
  - BiLSTM-CRF w/ char-level and w/ NER features.

Intuition: SQuAD answer spans contain a large number of named entities, numeric phrases, etc

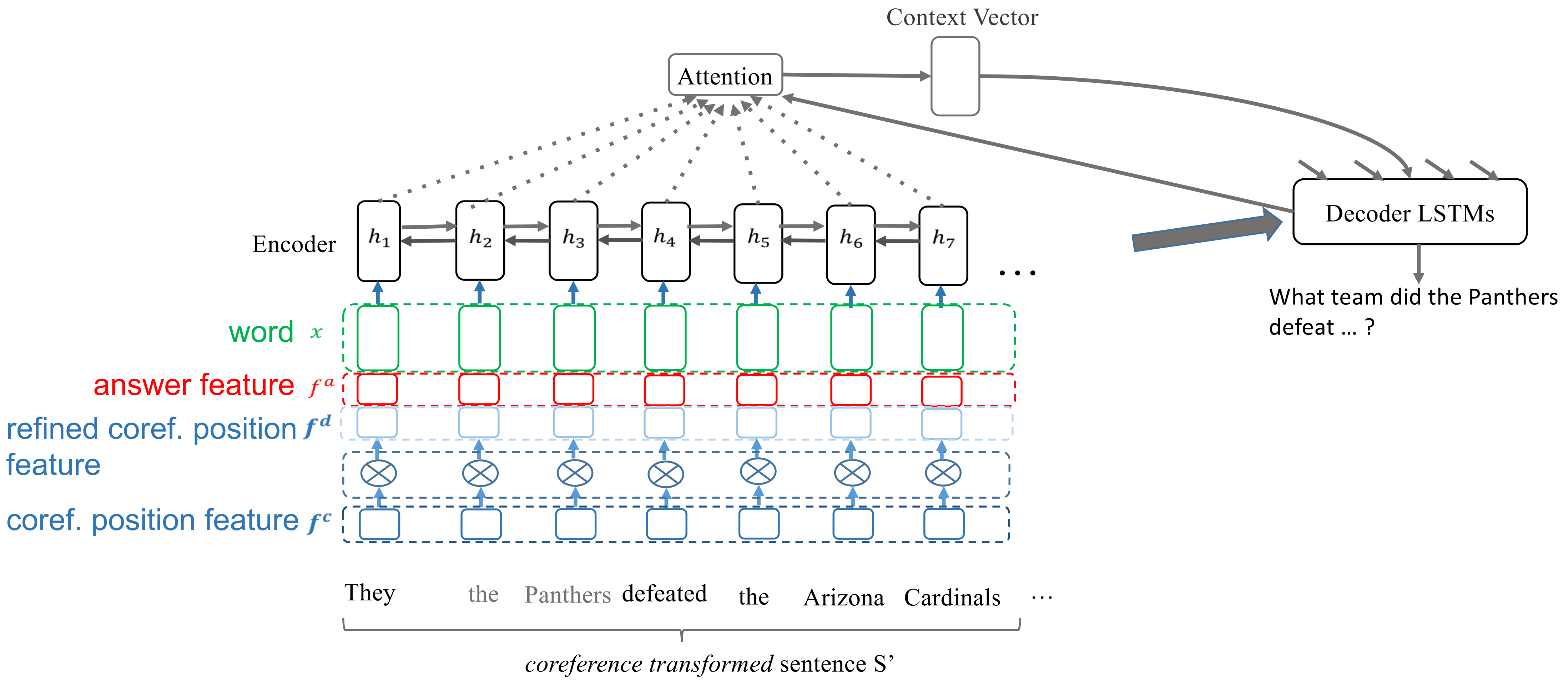
Methodology (Question Generation)

Original sentence: They defeated the Arizona Cardinals 49 – 15 in the NFC championship game.

word	they	the	panthers	defeated	the	arizona	cardinals	49	–	15	...
ans. feature	O	O	O	O	B_ANS	I_ANS	I_ANS	O	O	O	...
coref. feature	B_PRO	B_ANT	I_ANT	O	O	O	O	O	O	O	...

- ❖ For each pronoun (they) in sentence, we run the coref. model to identify the most "representative" antecedent (the panthers).
- ❖ Afterwards, we append the panthers after they.
- ❖ The row answer feature marks each token for belonging to an answer span.
- ❖ The row coreference feature marks each token for belonging to an coreferent entity.

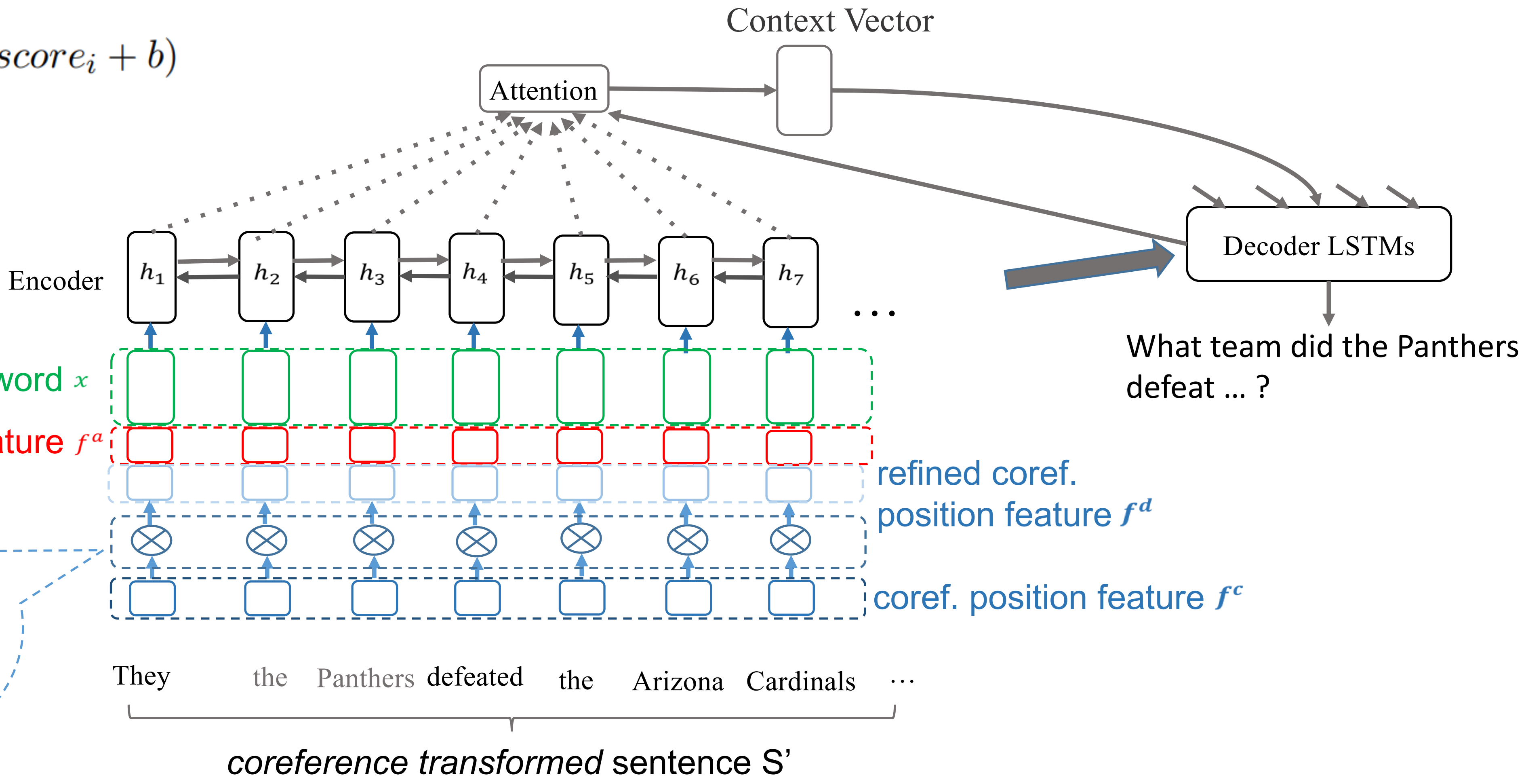
Methodology (Question Generation)

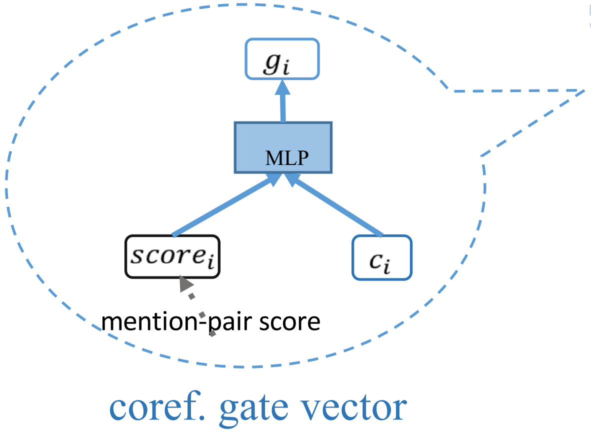


Methodology (Question Generation)

$$g_i = \mathrm{ReLU}(\mathbf{W}_a c_i + \mathbf{W}_b score_i + b)$$

$$d_i = g_i \odot c_i$$





Experiments (Data for Train/Test)

- ❖We use the SQuAD dataset (Rajpurkar et al., 2016) to train our models.
  - ➢one of the largest general purpose QA datasets derived from Wikipedia.
  - ➢100k questions posed by crowdworkers.
- ❖To quantify the effect of using predicted answer spans on question generation,
  - ➢We also train the QG models on dataset augmented w/ examples with predicted answer spans, that overlap with gold answer spans.
  - ➢Denoted as “Training set w/ noisy examples”.

Experiments (Results)

❖Automatic Evaluation for Answer Span Extraction

Models	Precision			Recall			F-measure		
	Prop.	Bin.	Exact	Prop.	Bin.	Exact	Prop.	Bin.	Exact
NER	24.54	25.94	12.77	**58.20**	**67.66**	**38.52**	34.52	37.50	19.19
BiLSTM	43.54	45.08	22.97	28.43	35.99	18.87	34.40	40.03	20.71
BiLSTM w/ NER	44.35	46.02	25.33	33.30	40.81	23.32	38.04	43.26	24.29
BiLSTM-CRF w/ char	**49.35**	**51.92**	**38.58**	30.53	32.75	24.04	37.72	40.16	29.62
BiLSTM-CRF w/ char w/ NER	45.96	51.61	33.90	41.05	43.98	28.37	**43.37**	**47.49**	**30.89**

Table 3: Evaluation results of answer extraction systems.

Experiments (Results)

❖Automatic Evaluation for Question Generation

Models	Training set			Training set w/ noisy examples		
	BLEU-3	BLEU-4	METEOR	BLEU-3	BLEU-4	METEOR
Baseline (Du et al., 2017) (w/o answer)	17.50	12.28	16.62	15.81	10.78	15.31
Seq2seq + copy (w/ answer)	20.01	14.31	18.50	19.61	13.96	18.19
ContextNQG: Seq2seq + copy (w/ full context + answer)	20.31	14.58	18.84	19.57	14.05	18.19
CorefNQG	**20.90**	**15.16**	**19.12**	**20.19**	**14.52**	18.59
- gating	20.68	14.84	18.98	20.08	14.40	**18.64**
- mention-pair score	20.56	14.75	18.85	19.73	14.13	18.38

Table 2: Evaluation results for question generation.

Experiments (Results)

❖Human Evaluation for Question Generation

	Grammaticality	Making Sense	Answerability	Avg. rank
ContextNQG	3.793	3.836	3.892	1.768
CorefNQG	3.804*	3.847**	3.895*	1.762
Human	**3.807**	**3.850**	**3.902**	**1.758**

- ➢ Human questions are preferred over the two NQG systems' outputs.
- ➢ In terms of only grammaticality, the neural models do quite well, close to human-level questions.
- ➢ CorefNQG performs statistically significantly better across all metrics than ContextNQG.

Experiments (Analysis)

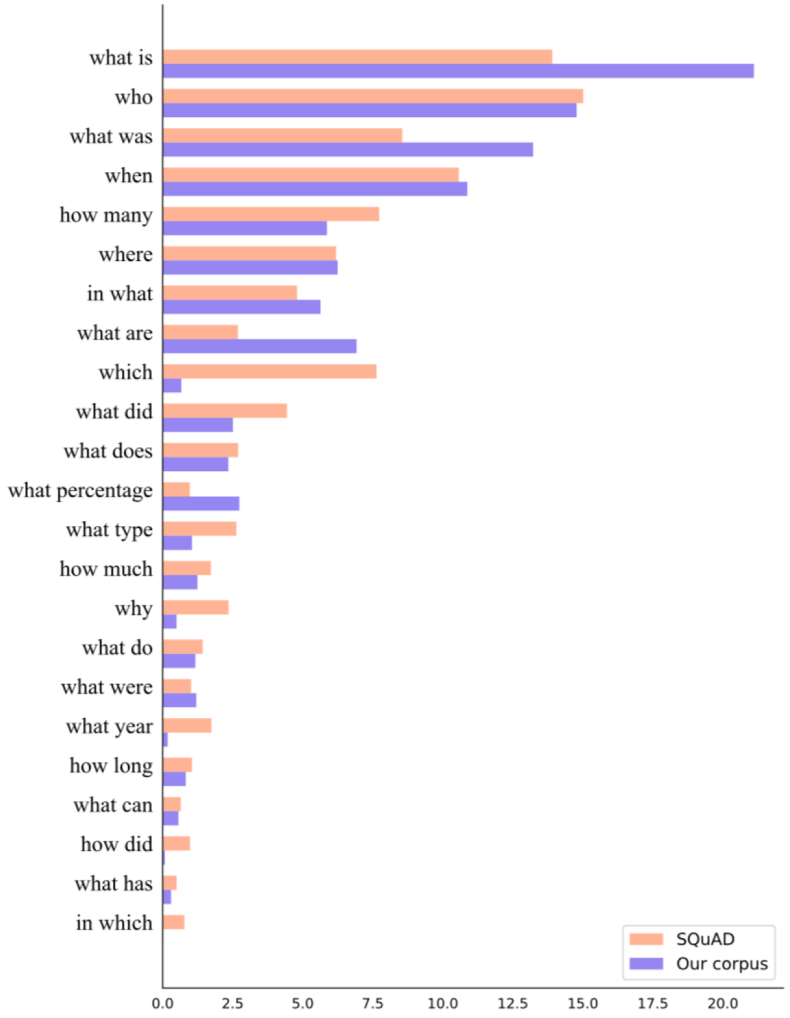
❖On the portion (36.42%) of data that requires coreference knowledge.

	BLEU-3	BLEU-4	METEOR
Seq2seq + copy (w/ ans.)	17.81	12.30	17.11
ContextNQG	18.05	12.53	17.33
CorefNQG	**18.46**	**12.96**	**17.58**

Table 4: Evaluation results for question generation on the portion that requires coreference knowledge (36.42% examples of the original test set).

Analysis for the Generated Dataset

❖ Distribution of question types of our corpus and SQuAD training set.



Analysis for the Generated Dataset

Input 1: The elizabethan navigator, sir francis drake was born in the nearby town of tavistock and was the mayor of plymouth. ... . he died of dysentery in 1596 off the coast of puerto rico.
Human: In what year did Sir Francis Drake die ?
ContextNQG: When did he die ?
CorefNQG: When did sir francis drake die ?

Input 2: american idol is an american singing competition ... . it began airing on fox on june 11 , 2002, as an addition to the idols format based on the british series pop idol and has since become one of the most successful shows in the history of american television.
Human: When did american idol first air on tv ?
ContextNQG: When did fox begin airing ?
CorefNQG: When did american idol begin airing ?

Analysis for the Generated Dataset

❖ Neural MR Model's Performance

	Exact Match		F-1	
	Dev	Test	Dev	Test
DocReader (Chen et al., 2017)	82.33	81.65	88.20	87.79

Table 6: Performance of the neural machine reading comprehension model (no initialization with pretrained embeddings) on our generated corpus.

Remarks

- Coreference chains are useful for generation tasks
  - paragraph level input
  - multi-turn conservation
- Building end-to-end models that take account coreference knowledge in a latent way is an interesting direction to explore.