Identifying Where to Focus in Reading Comprehension for Neural Question Generation

Xinya Du and **Claire Cardie**
Department of Computer Science
Cornell University
Ithaca, NY, 14853, USA
{xdu, cardie}@cs.cornell.edu

Abstract

A first step in the task of automatically generating questions for testing reading comprehension is to identify *question-worthy* sentences, i.e. sentences in a text passage that humans find it worthwhile to ask questions about. We propose a hierarchical neural sentence-level sequence tagging model for this task, which existing approaches to question generation have ignored. The approach is fully data-driven — with no sophisticated NLP pipelines or any hand-crafted rules/features — and compares favorably to a number of baselines when evaluated on the SQuAD data set. When incorporated into an existing neural question generation system, the resulting end-to-end system achieves state-of-the-art performance for paragraph-level question generation for reading comprehension.

1 Introduction and Related Work

Automatically generating questions for testing reading comprehension is a challenging task (Mannem et al., 2010; Rus et al., 2010). First and foremost, the question generation system must determine which concepts in the associated text passage are important, i.e. are worth asking a question about.

The little previous work that exists in this area currently circumvents this critical step in passage-level question generation by assuming that such sentences have already been identified. In particular, prior work focuses almost exclusively on *sentence-level* question generation: given a text passage, assume that all sentences contain a question-worthy concept and generate one or more questions for each (Heilman and Smith, 2010; Du et al., 2017; Zhou et al., 2017).

In contrast, we study the task of *passage-level question generation (QG)*. Inspired by the large body of research in text summarization on identifying sentences that contain "summary-worthy" content (e.g. Mihalcea (2005), Berg-Kirkpatrick et al. (2011), Yang et al. (2017)), we develop a method to identify the *question-worthy sentences* in each paragraph of a reading comprehension passage. Inspired further by the success of neural sequence models for many natural language processing tasks (e.g. named entity recognition (Collobert et al., 2011), sentiment classification (Socher et al., 2013), machine translation (Sutskever et al., 2014), dependency parsing (Chen and Manning, 2014)), including very recently document-level text summarization (Cheng and Lapata, 2016), we propose a hierarchical neural sentence-level sequence tagging model for question-worthy sentence identification.

We employ the SQuAD reading comprehension data set (Rajpurkar et al., 2016) for evaluation and show that our sentence selection approach compares favorably to a number of baselines including the feature-rich sentence selection model of Cheng and Lapata (2016) proposed in the context of extract-based summarization, and the convolutional neural network model of Kim (2014) that achieves state-of-the-art results on a variety of sentence classification tasks.

We also incorporate our sentence selection component into the neural question generation system of Du et al. (2017) and show, again using SQuAD, that our resulting end-to-end system achieves state-of-the-art performance for the challenging task of paragraph-level question generation for reading comprehension.

2 Problem Formulation

In this section, we define the tasks of *important* (i.e. question-worthy) *sentence selection* and *sentence-level question generation (QG)*. Our full paragraph-level QG system includes both of these components. For the sentence selection task, given a paragraph D consisting of a sequence of sentences $\{s_1, ..., s_m\}$, we aim to select a subset of k question-worthy sentences $(k < m)$. The goal is defined as finding $\overline{\mathbf{y}} = \{y_1, ..., y_m\}$, such that,

$$\begin{aligned}\overline{\mathbf{y}} &= \arg\max_{\mathbf{y}} \log P_1(\mathbf{y}|D) \\ &= \arg\max_{\mathbf{y}} \sum_{t=1}^{|\mathbf{y}|} \log P_1(y_t|D)\end{aligned} \tag{1}$$

where $\log\ P(\mathbf{y}|D)$ is the conditional log-likelihood of the label sequence $\mathbf{y}$; and $y_i = 1$ means sentence i is question-worthy (contains at least one answer), otherwise $y_i = 0$.

For sentence-level QG, the goal is to find the best word sequence $\overline{\mathbf{z}}$ (a question of arbitrary length) that maximizes the conditional likelihood given the input sentence $\mathbf{x}$ and satisfies:

$$\begin{aligned}\overline{\mathbf{z}} &= \arg\max_{\mathbf{z}} \log P_2(\mathbf{z}|\mathbf{x}) \\ &= \arg\max_{\mathbf{z}} \sum_{t=1}^{|\mathbf{z}|} \log P_2(z_t|\mathbf{x}, z_{<t})\end{aligned} \tag{2}$$

where $P_2(\mathbf{z}|\mathbf{x})$ is modeled with a global attention mechanism (Section 3).

3 Model

Important Sentence Selection Our general idea for the hierarchical neural network architecture is illustrated in Figure 1. First, we perform the encoding using sum operation or convolution+maximum pooling operation (Kim, 2014; dos Santos and Zadrozny, 2014) over the word vectors comprising each sentence in the input paragraph. For simplicity and consistency, we denote the sentence encoding process as ENC. Given the t^{th} sentence $\mathbf{x} = \{x_1, ..., x_n\}$ in the paragraph, we have its encoding:

$$\mathbf{s}_t = \text{ENC}([x_1, ..., x_n]) \tag{3}$$

Then we use a bidirectional LSTM (Hochreiter and Schmidhuber, 1997) to encode the paragraph,

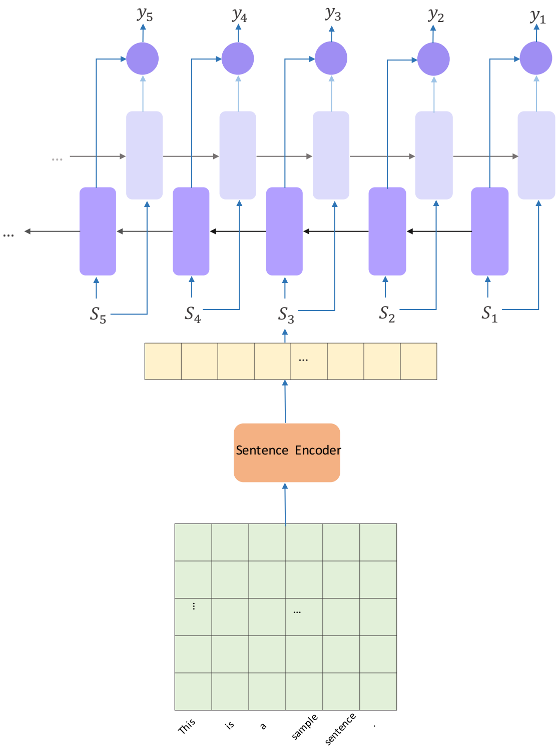


Figure 1: Hierarchical neural network architecture for sentence-level sequence labeling. The input is a paragraph consisting of sentences, whose encoded representation is fed into each hidden unit.

$$\begin{aligned}\overrightarrow{\mathbf{h}_t} &= \overrightarrow{\text{LSTM}}\left(\mathbf{s}_t, \overrightarrow{\mathbf{h}_{t-1}}\right) \\ \overleftarrow{\mathbf{h}_t} &= \overleftarrow{\text{LSTM}}\left(\mathbf{s}_t, \overleftarrow{\mathbf{h}_{t+1}}\right)\end{aligned}$$

We use the concatenation of the two, namely, $[\overrightarrow{\mathbf{h}_t}; \overleftarrow{\mathbf{h}_t}]$, as the hidden state $\mathbf{h}_t$ at time stamp t, and feed it to the upper layers to get the probability distribution of y_t $(\in \{0, 1\})$,

$$P_1(y_t|D;\theta) = \text{softmax}\left(\text{MLP}\left(\tanh\left([\overrightarrow{\mathbf{h}_t}; \overleftarrow{\mathbf{h}_t}]\right)\right)\right)$$

where MLP is multi-layer neural network and tanh is the activation function.

Question Generation Similar to Du et al. (2017), we implement the sentence-level question generator with an attention-based sequence-to-sequence learning framework (Sutskever et al., 2014; Bahdanau et al., 2015), to map a sentence in the reading comprehension article to natural questions. It consists of an LSTM encoder and decoder. The encoder is a bi-directional LSTM network; it encodes the input sentence $\mathbf{x}$ into a sequence of hidden states $\mathbf{q}_1, \mathbf{q}_2, ..., \mathbf{q}_{|\mathbf{x}|}$.

Model	Precision	Recall	F-measure	Acc.	Paragraph-level Acc.
RANDOM	63.45	50.29	56.11	50.27	11.69
Majority Baseline	63.21	100.00	77.46	63.21	32.30
CNN (Kim, 2014)	68.35	**90.13**	77.74	67.38	24.73
LREG$_{\text{(w/ BOW)}}$	68.52	86.55	76.49	66.37	31.36
LREG$_{\text{(w/ para.-level)}}$ (Cheng and Lapata, 2016)	70.49	89.08	78.70	69.52	33.95
Ours$_{\text{SUM}}$ (no pre-trained)	73.02	89.23	80.32	72.36	36.46
Ours$_{\text{SUM}}$ (w/ pre-trained)	73.85	87.65	80.16	**72.58**	36.30
Ours$_{\text{CNN}}$ (no pre-trained)	73.15	89.29	**80.42***	72.52	35.93
Ours$_{\text{CNN}}$ (w/ pre-trained)	**74.35**	86.11	79.80	72.44	**36.87**

Table 1: Automatic evaluation results for important sentence selection. The best performing system in each column is highlighted in boldface. Paragraph-level accuracies are calculated as the proportion of paragraphs in which *all* of the sentences are predicted correctly. We show two-tailed t-test results on F-measure for our best performing method compared to the other baselines. (Statistical significance is indicated with $^{*}(p < 0.005)$.)

The decoder is another LSTM that uses global attention over the encoder hidden states. The entire encoder-decoder structure learns the probability of generating a question given a sentence, as indicated by equation 2. To be more specific,

$$P_2\left(z_t|\mathbf{x}, z_{<t}\right) = \text{softmax}\left(\mathbf{W}_s \tanh\left(\mathbf{W}_t[\mathbf{h}_t; \mathbf{c}_t]\right)\right)$$

where $\mathbf{W}_s$, $\mathbf{W}_t$ are parameter matrices; $\mathbf{h}_t$ is the hidden state of the decoder LSTM; and $\mathbf{c}_t$ is the context vector created dynamically by the encoder LSTM — the weighted sum of the hidden states computed for the source sentence:

$$\mathbf{c}_t = \sum_{i=1,..,|\mathbf{x}|} a_{i,t}\mathbf{q}_i$$

The attention weights $a_{i,t}$ are calculated via a bilinear scoring function and softmax normalization:

$$a_{i,t} = \frac{\exp(\mathbf{h}_t^T \mathbf{W}_b \mathbf{q}_i)}{\sum_j \exp(\mathbf{h}_t^T \mathbf{W}_b \mathbf{q}_j)}$$

Apart from the bilinear score, alternative options for computing the attention can also be used (e.g. dot product). Readers can refer to Luong et al. (2015) for more details.

During inference, beam search is used to predict the question. The decoded `UNK` token at time step t, is replaced with the token in the input sentence with the highest attention score, the index of which is $\arg\max_i a_{i,t}$.

Henceforth, we will refer to our sentence-level **N**eural **Q**uestion **G**eneration system as **NQG**. Note that generating answer-specific questions would be easy for this architecture — we can append answer location features to the vectors of tokens in the sentence. To better mimic the real life case (where questions are generated with no prior knowledge of the desired answers), we do not use such location features in our experiments.

4 Experimental Setup and Results

4.1 Dataset and Implementation Details

We use the SQuAD dataset (Rajpurkar et al., 2016) for training and evaluation for both important sentence selection and sentence-level NQG. The dataset contains 536 curated Wikipedia articles with over 100k questions posed about the articles. The authors employ Amazon Mechanical Turk crowd-workers to generate questions based on the article paragraphs and to annotate the corresponding answer spans in the text. Later, to make the evaluation of the dataset more robust, other crowd-workers are employed to provide additional answers to the questions.

We split the public portion of the dataset into training (~80%), validation (~10%) and test (~10%) sets at the paragraph level. For the sentence selection task, we treat sentences that contain at least one answer span (question-worthy sentences) as positive examples ($y = 1$); all remaining sentences are considered negative ($y = 0$). Not surprisingly, the training set is unbalanced: 52332 (~60%) sentences contain answers, while 29693 sentences do not. Because of the variabil-

	Model	BLEU 1	BLEU 2	BLEU 3	BLEU 4	METEOR
Conservative	LREG$_{\text{(C\&L)}}$ + NQG	38.30	23.15	15.64	10.97	15.09
	Ours + NQG	40.08	24.26	16.39	11.50	15.67
Liberal	LREG$_{\text{(C\&L)}}$ + NQG	51.55	40.17	34.35	30.59	24.17
	Ours + NQG	52.89	41.16	35.15	31.25	24.76

Table 2: Results for the full QG systems using BLEU 1–4, METEOR. The first stage of the two pipeline systems are the feature-rich linear model (LREG) and our best performing selection model respectively.

	conservative eval.		liberal eval.	
Gold Data System Output	w/ Q	w/o Q	w/ Q	w/o Q
w/ Q	matching	zero	matching	full
w/o Q	zero	-	zero	-

Table 3: For a source sentence in SQuAD, given the prediction from the sentence selection system and the corresponding NQG output, we provide conservative and liberal evaluations.

ity of human choice in generating questions, it is the case that many sentences labeled as negative examples might actually contain concepts worth asking a question about. For the related important sentence detection task in text summarization, Yang et al. (2017) therefore propose a two-stage approach (Lee and Liu, 2003; Elkan and Noto, 2008) to augment the set of known summary-worthy sentences. In contrast, we adopt a conservative approach rather than predict too many sentences as being question-worthy: we pair up source sentences with their corresponding questions, and use just these sentence-question pairs to training the encoder-decoder model.

We use the `glove.840B.300d` pre-trained embeddings (Pennington et al., 2014) for initialization of the embedding layer for our sentence selection model and the full NQG model. `glove.6B.100d` embeddings are used for calculating sentence similarity feature of the baseline linear model (LREG). Tokens outside the vocabulary list are replaced by the `UNK` symbol. Hyperparameters for all models are tuned on the validation set and results are reported on the test set.

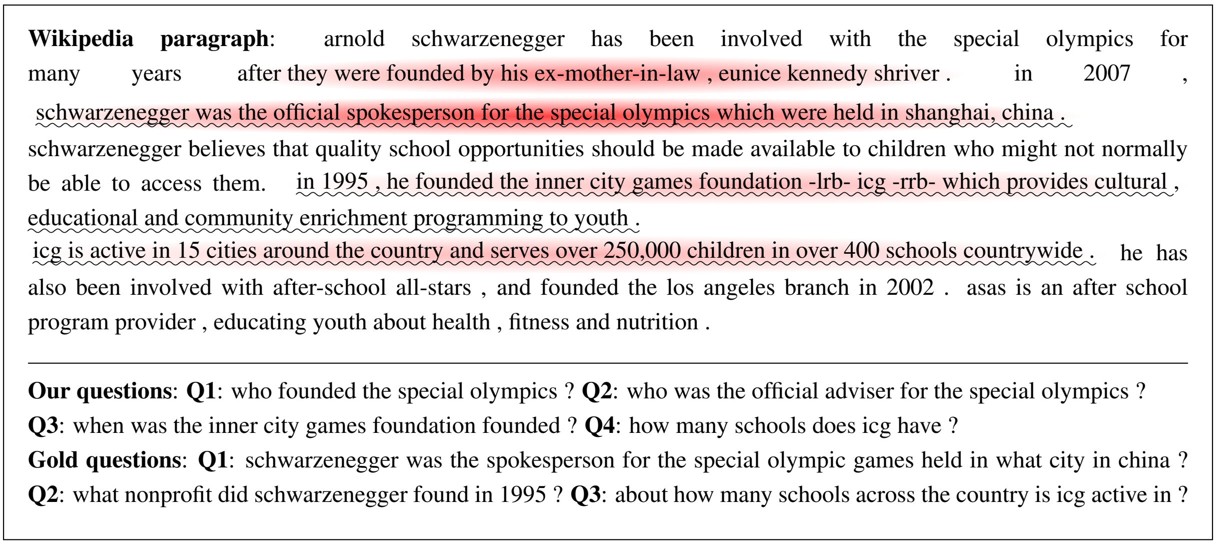
4.2 Sentence Selection Results

We compare to a number of baselines. The **Random** baseline assigns a random label to each sentence. The **Majority** baseline assumes that all sentences are question-worthy. The convolutional neural networks (**CNN**) sentence classification model (Kim, 2014) has similar structure to our CNN sentence encoder, but the classification is done only at the sentence-level rather than jointly at paragraph-level. **LREG$_{\text{w/ BOW}}$** is the logistic regression model with bag-of-words features. **LREG$_{\text{w/ para.-level}}$** is the feature-rich LREG model designed by Cheng and Lapata (2016); the features include: sentence length, position of sentence, number of named entities in the sentence, number of sentences in the paragraph, sentence-to-sentence cohesion, and sentence-to-paragraph relevance. Sentence-to-sentence cohesion is obtained by calculating the embedding space similarity between it and every other sentence in the paragraph (similar for sentence-to-paragraph relevance). In document summarization, graph-based extractive summarization models (e.g. TGRAPH Parveen et al. (2015) and URANK Wan (2010)) focus on global optimization and extract sentences contributing to topical coherent summaries. Because this does not really fit our task — a *summary-worthy* sentence might not necessarily contain enough information for generating a good question — we do not include these as comparisons.

Results are displayed in Table 1. Our models with sum or CNN as the sentence encoder significantly outperform the feature-rich LREG as well as the other baselines in terms of F-measure.

4.3 Evaluation of the full QG system

To evaluate the full systems for paragraph-level QG, we introduce in Table 3 the "conservative" and "liberal" evaluation strategies. Given an input source sentence, there will be in total four possibilities: if both the gold standard data and prediction include the sentence, then we use its n-gram matching score (by BLEU (Papineni et al., 2002) and METEOR (Denkowski and Lavie, 2014)); if neither the gold data nor prediction include the sentence, then the sentence is discarded from the evaluation; if the gold data includes the sentence while the prediction does not, we assign a score of 0 for it; and if gold data does not include the sentence while prediction does, the generated question gets a 0 for conservative, while it gets full



Wikipedia paragraph: arnold schwarzenegger has been involved with the special olympics for many years after they were founded by his ex-mother-in-law , eunice kennedy shriver . in 2007 , schwarzenegger was the official spokesperson for the special olympics which were held in shanghai, china . schwarzenegger believes that quality school opportunities should be made available to children who might not normally be able to access them. in 1995 , he founded the inner city games foundation -lrb- icg -rrb- which provides cultural , educational and community enrichment programming to youth . icg is active in 15 cities around the country and serves over 250,000 children in over 400 schools countrywide . he has also been involved with after-school all-stars , and founded the los angeles branch in 2002 . asas is an after school program provider , educating youth about health , fitness and nutrition .

Our questions: **Q1**: who founded the special olympics ? **Q2**: who was the official adviser for the special olympics ? **Q3**: when was the inner city games foundation founded ? **Q4**: how many schools does icg have ?
Gold questions: **Q1**: schwarzenegger was the spokesperson for the special olympic games held in what city in china ? **Q2**: what nonprofit did schwarzenegger found in 1995 ? **Q3**: about how many schools across the country is icg active in ?

Figure 2: Sample output from our full NQG system, the four questions correspond to the four highlighted sentences in the paragraph in the same order. Darkness indicates sentence importance, the score for deciding the darkness is obtained from the softmax results. Wave-lined sentences bear label $y = 1$, and 0 otherwise. The three gold questions also correspond to the wave-lined sentences in the same order. Please refer to the appendix for sample output on more Wikipedia articles.

score for liberal evaluation. Table 2 shows that the QG system incorporating our best performing sentence extractor outperforms its LREG counterpart across metrics. Note that to calculate the score for the matching case, similar to our earlier work (Du et al., 2017), we adapt the image captioning evaluation scripts of Chen et al. (2015) since there can be several gold standard questions for a single input sentence.

In Figure A, we provide questions generated by the full NQG system (Q1–4) and according to the gold standard (Q1–3) for the selected Wikipedia paragraph. The sentences they were drawn from are shown with wavy lines (gold standard) and via highlighting (our system). Darkness of the highlighting is proportional to the softmax score provided by the sentence extractor.

5 Conclusion

In this work we introduced the task of identifying important sentences — good sentences to ask a question about — in the reading comprehension setting. We proposed a hierarchical neural sentence labeling model and investigated encoding sentences with sum and convolution operations. The question generation system that uses our sentence selection model consistently outperforms previous approaches and achieves state-of-the-art paragraph-level question generation performance on the SQuAD data set.

In future work, we would like to investigate approaches to identify question-worth *concepts* rather than question-worthy sentences. it would be interesting to see if the generated questions can be used to help improve question answering systems.

Acknowledgments

We thank the reviewers for helpful comments and Victoria Litvinova for proofreading.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *International Conference on Learning Representations Workshop (ICLR)*.

Taylor Berg-Kirkpatrick, Dan Gillick, and Dan Klein. 2011. Jointly learning to extract and compress. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, Portland, Oregon, USA, pages 481–490. http://www.aclweb.org/anthology/P11-1049.

Danqi Chen and Christopher Manning. 2014. A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Doha, Qatar, pages 740–750. http://www.aclweb.org/anthology/D14-1082.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and

C Lawrence Zitnick. 2015. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325* .

Jianpeng Cheng and Mirella Lapata. 2016. Neural summarization by extracting sentences and words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Berlin, Germany, pages 484–494. http://www.aclweb.org/anthology/P16-1046.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research* 12(Aug):2493–2537.

Michael Denkowski and Alon Lavie. 2014. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*. Association for Computational Linguistics, Baltimore, Maryland, USA, pages 376–380. http://www.aclweb.org/anthology/W14-3348.

Cícero Nogueira dos Santos and Bianca Zadrozny. 2014. Learning character-level representations for part-of-speech tagging. In *ICML*. pages 1818–1826.

Xinya Du, Junru Shao, and Claire Cardie. 2017. Learning to ask: Neural question generation for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*. To appear.

Charles Elkan and Keith Noto. 2008. Learning classifiers from only positive and unlabeled data. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, New York, NY, USA, KDD '08, pages 213–220. https://doi.org/10.1145/1401890.1401920.

Michael Heilman and Noah A. Smith. 2010. Good question! statistical ranking for question generation. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, Los Angeles, California, pages 609–617. http://www.aclweb.org/anthology/N10-1086.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9(8):1735–1780.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Doha, Qatar, pages 1746–1751. http://www.aclweb.org/anthology/D14-1181.

Wee Sun Lee and Bing Liu. 2003. Learning with positive and unlabeled examples using weighted logistic regression. In *Proceedings of the Twentieth International Conference on International Conference on Machine Learning*. AAAI Press, ICML'03, pages 448–455. http://dl.acm.org/citation.cfm?id=3041838.3041895.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Lisbon, Portugal, pages 1412–1421. http://aclweb.org/anthology/D15-1166.

Prashanth Mannem, Rashmi Prasad, and Aravind Joshi. 2010. Question generation from paragraphs at upenn: Qgstec system description. In *Proceedings of QG2010: The Third Workshop on Question Generation*. pages 84–91.

Rada Mihalcea. 2005. Language independent extractive summarization. In *Proceedings of the ACL Interactive Poster and Demonstration Sessions*. Association for Computational Linguistics, Ann Arbor, Michigan, pages 49–52. https://doi.org/10.3115/1225753.1225766.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Philadelphia, Pennsylvania, USA, pages 311–318. https://doi.org/10.3115/1073083.1073135.

Daraksha Parveen, Hans-Martin Ramsl, and Michael Strube. 2015. Topical coherence for graph-based extractive summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Lisbon, Portugal, pages 1949–1954. http://aclweb.org/anthology/D15-1226.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Doha, Qatar, pages 1532–1543. http://www.aclweb.org/anthology/D14-1162.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Austin, Texas, pages 2383–2392. https://aclweb.org/anthology/D16-1264.

Vasile Rus, Brendan Wyse, Paul Piwek, Mihai Lintean, Svetlana Stoyanchev, and Cristian Moldovan. 2010. The first question generation shared task evaluation challenge. In *Proceedings of the 6th International Natural Language Generation Conference*. Association for Computational Linguistics, Stroudsburg, PA, USA, pages 251–257. http://dl.acm.org/citation.cfm?id=1873738.1873777.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Seattle, Washington, USA, pages 1631–1642. http://www.aclweb.org/anthology/D13-1170.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems (NIPS)*. pages 3104–3112.

Xiaojun Wan. 2010. Towards a unified approach to simultaneous single-document and multi-document summarizations. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*. Coling 2010 Organizing Committee, Beijing, China, pages 1137–1145. http://www.aclweb.org/anthology/C10-1128.

Yinfei Yang, Forrest Bao, and Ani Nenkova. 2017. Detecting (un)important content for single-document news summarization. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*. Association for Computational Linguistics, Valencia, Spain, pages 707–712. http://www.aclweb.org/anthology/E17-2112.

Qingyu Zhou, Nan Yang, Furu Wei, Chuanqi Tan, Hangbo Bao, and Ming Zhou. 2017. Neural question generation from text: A preliminary study. *arXiv preprint arXiv:1704.01792* .

A Supplementary Materials

Below we provide more sample outputs from our system.

Paragraph 1 in 1939 , coinciding with the start of world war ii , rene dubos reported the discovery of the first naturally derived antibiotic , tyrothricin , a compound of 20 % gramicidin and 80 % tyrocidine , from b. brevis . it was one of the first commercially manufactured antibiotics universally and was very effective in treating wounds and ulcers during world war ii . gramicidin , however , could not be used systemically because of toxicity . tyrocidine also proved too toxic for systemic usage . research results obtained during that period were not shared between the axis and the allied powers during the war .

Our questions:
Q1: what was the name of the compound that was discovered in 1939 ?
Q2: what was one of the first commercially manufactured antibiotics ?

Gold questions:
Q1: (1) what was the first antibiotic developed from nature ? (2) when was tyrothricin created ? (3) what also happened in 1939 besides tyrothricin ?
Q2: what was tyrothricin used for during the war ?

Paragraph 2 after the english civil war the royal citadel was built in 1666 on the east end of plymouth hoe , to defend the port from naval attacks , suppress plymothian parliamentary leanings and to train the armed forces . Guided tours are available in the summer months . further west is smeaton 's tower , which was built in 1759 as a lighthouse on rocks 14 miles -lrb- 23 km -rrb- off shore, but dismantled and the top two thirds rebuilt on the hoe in 1877 . It is open to the public and has views over the plymouth sound and the city from the lantern room . Plymouth has 20 war memorials of which nine are on the hoe including : plymouth naval memorial , to remember those killed in world wars i and ii , and the armada memorial, to commemorate the defeat of the spanish armada .

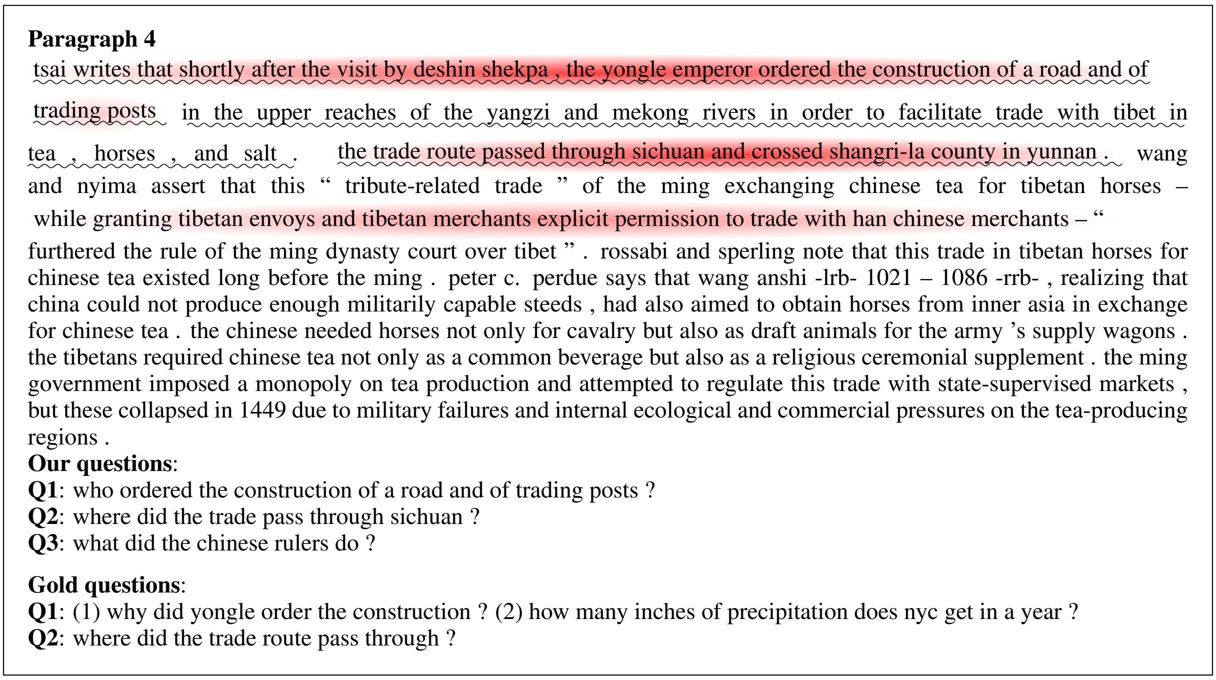
Our questions:
Q1: when was the royal basilica fort built ?
Q2: when was the smeaton 's tower built ?
Q3: how many war memorials did plymouth have ?

Gold questions:
Q1: in what year was the royal citadel constructed ?
Q2: (1) when was smeaton 's tower first constructed ? (2) in kilometers , how far off the coast was smeaton 's tower originally built ?
Q3: what memorial commemorates the naval victory over the spanish armada ?

Paragraph 3 the city receives 49.9 inches -lrb- 1,270 mm -rrb- of precipitation annually , which is fairly spread throughout the year . average winter snowfall between 1981 and 2010 has been 25.8 inches -lrb- 66 cm -rrb- , but this varies considerably from year to year . hurricanes and tropical storms are rare in the new york area , but are not unheard of and always have the potential to strike the area . hurricane sandy brought a destructive storm surge to new york city on the evening of october 29 , 2012 , flooding numerous streets , tunnels , and subway lines in lower manhattan and other areas of the city and cutting off electricity in many parts of the city and its suburbs . the storm and its profound impacts have prompted the discussion of constructing seawalls and other coastal barriers around the shorelines of the city and the metropolitan area to minimize the risk of destructive consequences from another such event in the future .

Our questions:
Q1: how much precipitation does the city receive annually ?
Q2: what is the average winter snowfall between 1981 and 2010 ?
Q3: what caused the storm surge to new york city ?

Gold questions:
Q1: (1) in millimeters , how much precipitation does new york receive a year ? (2) how many inches of precipitation does nyc get in a year ?
Q2: (1) in centimeters , what is the average winter snowfall ? (2) the mean snowfall between 1981 and 2010 in nyc has been how many inches ?
Q3: (1) when did hurricane sandy strike new york ? (2) which natural disaster occurred on october 29 , 2012 in nyc ?



Paragraph 4

tsai writes that shortly after the visit by deshin shekpa , the yongle emperor ordered the construction of a road and of trading posts in the upper reaches of the yangzi and mekong rivers in order to facilitate trade with tibet in tea , horses , and salt . the trade route passed through sichuan and crossed shangri-la county in yunnan . wang and nyima assert that this " tribute-related trade " of the ming exchanging chinese tea for tibetan horses – while granting tibetan envoys and tibetan merchants explicit permission to trade with han chinese merchants – " furthered the rule of the ming dynasty court over tibet " . rossabi and sperling note that this trade in tibetan horses for chinese tea existed long before the ming . peter c. perdue says that wang anshi -lrb- 1021 – 1086 -rrb- , realizing that china could not produce enough militarily capable steeds , had also aimed to obtain horses from inner asia in exchange for chinese tea . the chinese needed horses not only for cavalry but also as draft animals for the army 's supply wagons . the tibetans required chinese tea not only as a common beverage but also as a religious ceremonial supplement . the ming government imposed a monopoly on tea production and attempted to regulate this trade with state-supervised markets , but these collapsed in 1449 due to military failures and internal ecological and commercial pressures on the tea-producing regions .

Our questions:
Q1: who ordered the construction of a road and of trading posts ?
Q2: where did the trade pass through sichuan ?
Q3: what did the chinese rulers do ?

Gold questions:
Q1: (1) why did yongle order the construction ? (2) how many inches of precipitation does nyc get in a year ?
Q2: where did the trade route pass through ?

Figure 3: Sample output from our system on more Wikipedia articles. Our questions correspond to the highlighted sentences in the same order, the gold standard questions correspond to the wave-lined sentences also in the same order. Darkness indicate the importance of the sentence predicted by our system.