PaintTeR: Automatic Extraction of Text Spans for Generating Art-Centered Questions

**Sujatha Das Gollapalli,[1] See-Kiong Ng,[1] Ying Kiat Tham,[1]
Shan Shan Chow,[2] Jia Min Wong[2], Kevin Lim[2]**
[1]Institute of Data Science, National University of Singapore
[2]CoLab X-Innovation Lab, National Gallery Singapore
{idssdg,seekiong,idstyk}@nus.edu.sg; {shanshan.chow,jiamin.wong,kevin.lim}@nationalgallery.sg

Abstract

We propose *PaintTeR*, our **Paint**ings **Te**xt**R**ank algorithm for extracting art-related text spans from passages on paintings. *PaintTeR* combines a lexicon of painting words curated automatically through distant supervision with random walks on a large-scale word co-occurrence graph for ranking passage spans for artistic characteristics. The spans extracted with *PaintTeR* are used in state-of-the-art Question Generation and Reading Comprehension models for designing an interactive aid that enables gallery and museum visitors focus on the artistic elements of paintings. We provide experiments on two datasets of expert-written passages on paintings to showcase the effectiveness of *PaintTeR*. Evaluations by both gallery experts as well as crowdworkers indicate that our proposed algorithm can be used to select relevant and interesting art-centered questions. To the best of our knowledge, ours is the first work to effectively fine-tune question generation models using minimal supervision for a low-resource, specialized context such as gallery visits.

Introduction

Human-led docent tours represent the most popular way for gallery and museum visitors to experience the interesting artworks or historical artefacts on exhibition, but they are often oversubscribed. Many galleries and museums have started to experiment with chatbots to bridge the gap between the highly engaging human docents and passive classical audioguides (Schaffer et al. 2018; Boiano et al. 2018). For example, the chatbot Arthena[1] was piloted at National Gallery Singapore[2] to enable its visitors to learn more before, during, and after their visits of an exhibition. Emerging AI applications such as chatbots for GLAMs (Galleries, Library, Archives and Museums) have the potential to unlock the treasure troves of content for the general public by automatically assimilating the available rich content with personal interests and interacting naturally and intelligently with their visitors using AI (Strien et al. 2021).

Most current chatbots for GLAMs are based on Question Answering (QA) systems that combine retrieval of relevant answer passages with Reading Comprehension (RC), namely, the task of extracting an answer from a given passage for a question. However, it is unrealistic to carry out an interactive conversation with GLAM visitors expecting that they will always have specific questions to ask about the objects. Experienced gallery docents often use carefully crafted art-centered questions to guide their visitors in discovering and appreciating an artwork. As such, Automatic Question Generation (QG), namely the task of generating natural language questions for a given input text passage, is a key capability for designing effective GLAM chatbots.

Research in RC and QG has garnered significant focus in the AI and NLP communities in the last decade (Kim et al. 2019; Tuan, Shah, and Barzilay 2020; Ram et al. 2021; Huang et al. 2021). These two tasks are useful in applications for education and tutoring (Lindberg et al. 2013) and for designing dialog systems and chatbots (Wang et al. 2020a). Currently, state-of-the-art performance for QG task is attained by "answer-aware" neural models (Qi et al. 2020) whereas transformer-based ensembles have outperformed humans on the RC task on some datasets (Zhang, Yang, and Zhao 2021).

Answer-aware QG models focus on "how to ask" (constructing questions) given "what to ask" (content selection) in contrast with answer-unaware models that also incorporate a content selection module (Pan et al. 2019). In this paper, we address content selection for QG models in the specialized context of supporting viewers of paintings during their gallery visits. Paintings and historical artefacts exhibited in galleries and museums are often accompanied by text passages written by experts provided alongside the installation or in separate brochures. In the gallery setting, these passages, in addition to descriptions of the content and subject matter of the artwork, often include details pertaining to the artist and background context such as when and where it was created, how it was acquired by the gallery, and so on and are aimed at providing a deeper understanding of various aspects of the artwork to the viewers.

Given an expert-written passage on an artwork, can we choose content that refers to the visual and artistic aspects of the same? How can we engage viewers via questions to focus on artistic elements of a painting? We address these two questions in context of *ArtQuest*, our question-based expository aid to generate art-centered questions that can be used by a gallery chatbot in engaging with visitors.

[1] https://www.facebook.com/chatbotarthena

[2] https://www.nationalgallery.sg/

Passage A moth sits beside a wicker basket containing a profusion of roses, tulips, primulas, and daisies. It is the work of a female artist, not in itself uncommon in 18th-century Holland, and the subject matter, a floral still life, was highly popular. The paintings, as here, often had insects such as caterpillars and butterflies included to enhance the naturalism of the image. Rachel Ruysch was the most celebrated Dutch flower painter of her day. . . .
Art-centered Questions 1. What type of still life was Rachel Ruysch inspired by? 2. Where does the moth sit in Ruysch's painting? 3. Why were insects such as caterpillars and butterflies included in paintings?
Other Candidate Questions 4. In what century was floral still life popular in Holland? 5. What was Rachel Ruysch's career? 6. What type of classes were women not allowed to attend?

Table 1: Example questions for the passage on the painting "Flowers and Insects" by Rachel Ruysch[3]

The objective of *ArtQuest* is to enable gallery visitors focus on the top art-centered aspects amongst the other factual aspects of a painting. *ArtQuest* uses *PaintTeR*, our novel **Paint**ings **Te**xt**R**ank algorithm for extracting text spans pertaining to the subject matter and interpretation of paintings and uses them as cues in state-of-the-art Question Generation (QG) models. For illustration, Table 1 shows sample questions generated by the ProphetNet QG model (Qi et al. 2020) for an example passage available in the public-domain.[3] We posit that questions 1-3 are more pertinent to understanding the visual nature of this artwork in contrast to questions 4-6, that pertain to facts related to the artist, or the historical context of this artwork.

Contributions: (1) We present *PaintTeR* (**Paint**ings **Te**xt**R**ank), a PageRank-style algorithm that incorporates word associations with painting documents and random walks on a large-scale, word co-occurrence graph to effectively rank artistic text spans from expert-written painting passages. We apply distant supervision to automatically curate a list of a painting-specific words from Wikipedia in order to enable the art-centered focus required in the gallery context. (2) We highlight the effectiveness of *PaintTeR* via the QG task inside our interactive system, *ArtQuest*. Human annotation results from gallery experts and crowdworkers on two datasets illustrate the effectiveness of *PaintTeR*-extracted spans in generating art-centered questions. To the best of our understanding, ours is the first work to apply minimal supervision for content selection used for generating "faceted" questions (in the gallery context).

Methods

Problem Formulation: Given a text document d describing a painting $\mathcal{P}$, our objective is to extract a ranked list of text spans (passage segments, sequences of words) from d that best represent the art-centered aspects of $\mathcal{P}$. We adopt the following steps:

1. Extract a set of candidate spans S_c from d.
2. Score each $s \in S_c$ for its art-centeredness.
3. Choose the top-scoring spans $S_a \subseteq S_c$ as the set of predictions.

The above steps are similar to those adopted in unsupervised keyphrase extraction (KE) methods (Mihalcea and Tarau 2004; Wan and Xiao 2008) except that unlike KE, where the objective is to choose S_a that summarizes the topical content of d, in our setting, S_a represents the artistic content of d.[4] To meet this objective, we leverage two resources: (1) A word co-occurrence graph $G = (V, E)$ built from a large-scale, representative corpus, and (2) A lexicon $\mathcal{L}$ of words commonly associated with paintings.

Word co-occurrence graphs are known to capture contextual and latent language information and were successfully applied in various NLP tasks such as keyphrase extraction, emotion detection, and summarization (Wan and Xiao 2008; Rozenshtein, Gollapalli, and Ng 2020). Every $v \in V$ corresponds to a unique word in the corpus vocabulary $\mathcal{V}$ and the lexicon $\mathcal{L} \subset \mathcal{V}$. An edge between two word vertices is weighted by its co-occurrence frequency in the underlying corpus. We characterize the text similarity between a text span s and the dictionary $\mathcal{L}$ using *PaintTeR*, our **R**anking algorithm for scoring **Te**xt spans in **Paint**ings.

The *PaintTeR* Algorithm

Inspired by earlier works (Duan et al. 2018), we compute *PaintTeR* scores for a given text span via random walks in G. To define such a random walk, we need a transition matrix whose each entry represents the probability of moving from one word to another. We use a co-occurrence matrix $A_{|V| \times |V|}$ derived from some large corpus, representative of general world knowledge (e.g., Wikipedia), where $A[i, j]$ is the number of times words i and j appear in the same context window. The entries of A are row-normalized to convert A to a stochastic matrix. The **random walk with restarts** model is defined by an imaginary walker who starts walking from a randomly chosen vertex $v \in V$. At each time step, the walker moves to another word vertex (neighbors of v in G according to the transition matrix) with probability $\alpha \in [0, 1]$ or stops with the probability $1 - \alpha$. This walk behaviour is defined by the matrix (Haveliwala 2003; Duan et al. 2018):

$$P = (1 - \alpha) \sum_{k=0}^{\infty} \alpha^k A^k,$$

where k is the length of the walk.

Our goal is to measure the similarity between the two vectors corresponding to text span s and the words in $\mathcal{L}$. Therefore, we restrict the walk to restart only inside the vertices corresponding to the span s instead of all vertices in G. Assuming a uniform probability for choosing words in s, the

[3] https://www.britannica.com/list/25-famous-paintings-to-see-the-next-time-youre-in-florence

[4] Thus we also extend the set of candidates S_c from the list of noun phrases used in KE works to include other common collocation patterns: https://en.wikipedia.org/wiki/English_collocations

probability that a random walker stops at a word $w \in \mathcal{V}$ restarting from the words in s is given by:

$$\textit{PPR}(x_s, w) = (1-\alpha)\frac{x_s^T}{\|x_s\|_1}\sum_{k=0}^{\infty}\alpha^k A^k e_w,$$

where e_w is a one-hot vector with 1 at the position corresponding to the word w and x_s refers to the binary bag-of-words column vector of length $|V|$ corresponding to the text span. Aggregating over all words in the lexicon, our similarity function can be written as:

$$\textit{PaintTeR}(s) = \textit{PPR}(x_s)^T x_{\mathcal{L}} \tag{1}$$

where $x_{\mathcal{L}}$ is the binary bag-of-words column vector for the lexicon. To summarize, the $\textit{PaintTeR}(s)$ score is the aggregate probability that a random walk with restarts in given text span s ends in the words from the dictionary $\mathcal{L}$. We note that $\textit{PPR}(x_s, w)$ is the classic Personalized or Topic-Sensitive extension to the PageRank score of a word w for a personalization (topic) vector $\frac{x_s^T}{\|x_s\|_1}$ and these variations of PageRank were first studied for webpages (Page et al. 1999; Haveliwala, Kamvar, and Jeh 2003; Haveliwala 2003).

Wikipedia as a Knowledge Resource: We used the textual content in Wikipedia articles for computing our word co-occurrence graph as well as for curating our lexicon. Wikipedia constitues a free online "encyclopedia" constructed through collaborative effort of contributors and includes articles on various topics. The articles are organized as a graph with category labels (from a taxonomy) assigned to them. Due to its scale, reliability, and coverage on diverse topics (Anthony, Smith, and Williamson 2009), several NLP works use Wikipedia as a knowledge resource for solving problems such as semantic term relatedness, entity disambiguation, and text classification (Witten and Milne 2008; Gattani et al. 2013; Nguyen, Matsuo, and Ishizuka 2007).

Compiling a Lexicon of Painting Words

One of the crucial resources for computing *PaintTeR* scores is the lexicon $\mathcal{L}$ containing words discriminative of painting documents. However, to the best of our knowledge, such an *art expert*-compiled lexicon is not available. We address this deficiency by using "distant supervision" (Mintz et al. 2009) to compile such a lexicon automatically from Wikipedia.

Distant supervision is often employed to generate noisy labels in lieu of a large set of manually labeled data (Ji et al. 2017; Xie et al. 2020). For example, crowd-annotated labels for news documents were used to compute scores at word-level in DepecheMood (Staiano and Guerini 2014). Following this paradigm, we use document-level labels available in a reliable corpus such as Wikipedia to compile painting association scores for words and automatically curate a lexicon using the following steps:

We obtain $\mathcal{D}_P$, the subset of Wikipedia articles with the category label "Category:Painting". The list of words seen in $d \in \mathcal{D}_P$ forms our candidate list of words L_c. Our lexicon for painting documents $\mathcal{L} \subset L_c$, must contain words that are discriminative of paintings. That is, for a word to be part of $\mathcal{L}$, its probability of appearing in articles assigned to the paintings category should be greater than its probability of appearing uniformly across all categories. Based on this notion, we compute: $P_u(w) = \frac{DF(w)}{N}$ and $P_p(w) = \frac{DF_P(w)}{N_P}$ where $DF(w)$ refers to the document frequency of word w in Wikipedia, $DF_P(w)$ its document frequency in the subset D_P and N, N_p the total number of documents in Wikipedia and the subset $\mathcal{D}_P$, respectively. Our lexicon comprises of all words $\forall w \in \mathcal{L}$, $P_p(w) > P_u(w)$.

The *ArtQuest* Interface

Background: An experimental chatbot called Arthena for answering questions on artworks was recently deployed through the Facebook messenger for the National Gallery Singapore's "Georgette Chen: At Home in the World" exhibition from 27 November 2020 to 26 September 2021.[5] Arthena employed "evoking" questions manually crafted by Gallery experts for this exhibition. A user study conducted by the Gallery for Arthena revealed several interesting findings: (1) Through handcrafted evoking questions, viewers' curiosity of a painting's artistic aspects was piqued and they started to observe the visual elements of the paintings more; (2) The visitors, in general, did not carefully peruse the passages accompanying an artwork, but when interested in a painting, they seek to understand more of the contextual information (regarding the artist, intent and feelings) by reading the passage; and (3) Most visitors did not know how to initiate questions on an artwork ("what to ask").

We designed *ArtQuest*, an experimental prototype, motivated by the above findings. *ArtQuest* is a simple, quiz-style interface combining Question Generation (QG) and Reading Comprehension (RC) modules to interact with viewers of paintings. Gallery visitors engage with *ArtQuest* by asking questions that are answered by the machine (using the RC module) while also attempting to answer machine-generated questions on a specific painting (using the QG module). This interaction is scored over a session to provide a fun, play-based experience for the viewer.

Through an appropriate choice of art-centered questions derived from the accompanying text passages, *ArtQuest* not only encourages the visitors to discover relevant contextual details provided in the texts, but also indirectly guides the visitors to observe the painting more by focusing on its key visual aspects and artistic elements as mentioned in the texts. We include more details on the *ArtQuest* interface with example snapshots and the list of models considered for QG and RC in the Appendix.[6] A demo of *ArtQuest* is accessible at http://artquestapp.herokuapp.com/app.

Experiments

Datasets: Due to novelty of our context, benchmark datasets are not available for evaluating our proposed algorithms. Therefore, we compiled the following two datasets: (1) *GCG*: a proprietary collection of articles describing paintings by <u>G</u>eorgette <u>C</u>hen, a pioneer Singapore artist, provided

[5] https://www.nationalgallery.sg/see-do/programme-detail/422534596/chatbot-arthena-delve-deeper-into-georgette-chen's-world

by National Gallery Singapore, and (2) *RAB*: a set of articles describing famous paintings by Renaissance Artists included in Britannica and publicly available.[3] Our two datasets are summarized in Table 2. Across the two datasets, the shortest passage has about 3 sentences whereas the largest number of questions generated for one of the passages was 39. The last column (#TotalQs) refers to questions generated using all extracted candidate spans before ranking with *PaintTeR*.

Question Annotation

For evaluating the quality of machine-generated questions, we obtained manual annotations for subsets of about ten articles for each of the datasets. About 153 questions were manually examined by two experts from the National Gallery Singapore for the *GCG* dataset whereas crowdworkers were employed on the Amazon Mechanical Turk (AMT) platform. for obtaining annotations for a subset of 310 questions for the *RAB* dataset.

In line with recent QG works (Gao et al. 2019; Pan et al. 2020; Wang et al. 2020a), all questions were scored along four dimensions: (1) **Fluency**: Is the question grammatically correct, natural sounding, and semantically valid for the given passage context, (2) **Answerability**: Is the answer to the generated question present in the passage, (3) **Correctness**: is the answer extracted by our RC module correct and complete, and (4) **Relevance**: is the question centered on the subject, content, or artistic aspects of the painting. Fluency, Answerability, and Correctness provide evaluation for the QG and RC modules employed in *ArtQuest*, while Relevance is a measure of how well *PaintTeR* does in choosing artistic text spans for subsequently producing art-centered questions. A rating scale with three values: “Yes” (1), “No” (0), “Acceptable/Partially Correct” (0.5) was employed.

For the *GCG* dataset, the two experts annotated five different articles each and one common article (18 questions) that was used to compute the interannotator agreement. The Cohen’s Kappa value was 0.553 on this dataset indicating moderate agreement (Landis and Koch 1977).

For the *RAB* dataset, the crowdworkers on AMT were required to have greater than 95% HIT approval rate, a minimum of 10,000 HITs, and be located in the United States. We offered a remuneration of $0.40 for each question per worker and each question was annotated by three crowdworkers. These settings are in line with previous QA/QG data collection efforts to ensure the *quality* and *ethical* considerations while obtaining crowd-annotated data. A total of 31 workers helped in creating this dataset, with about 41% of the workers labeling less than 5 questions each. More details on the AMT task are provided in the Appendix.[6]

Baselines

We compared *PaintTeR* scores with two straightforward unsupervised baselines used by closely-related works on measuring relevance of short text segments with a given concept. Let W_s refer to the words in a given text span s.

Dataset	#Articles	#Sents	#Qs	#TotalQs
GCG	21	3-10	6-21	294
RAB	25	7-15	19-39	772

Table 2: Dataset Summary: #Sents and #Qs refer to the number of sentences and questions, respectively.

- (**WN-BL**) We use WordNet (Miller 1995), a database of words linked as a graph via semantic relations and a popular resource in NLP, for computing the score for span s. In WordNet, the synonyms for a word (for a given part-of-speech and sense) are grouped together to form synsets. We used the set of painting-related synsets: P_{syn}= {‘painting.n.01’, ‘painting.n.02’, ‘art.n.01’, ‘art.n.02’} to define WN-score(s)=$\frac{1}{min(D)+1}$, where D is the set of shortest path distances between the set P_{syn} and $syns(w)$, $\forall w \in W_s$. Here $syns(w)$ refers to all synsets of word w (Budanitsky and Hirst 2006).
- (**LO-BL**) The paintings lexicon $\mathcal{L}$ (from the **Methods** section) can be directly used to compute a score for s (Staiano and Guerini 2014). Let O refer to the set of overlapping words in the sets: W_s and $\mathcal{L}$. The lexicon-based score, LO-score(s)=$\frac{\sum\limits_{w \in O} P_p(w)}{|O|}$ where $P_p(w)$ refers to the association probability of the word with painting documents.

For both the baselines, the score assigned to a given span is zero if no words in the span are found in WordNet (WN-BL) or there is no overlap between the span and the lexicon (LO-BL). Note that *PaintTeR* is able to handle zero overlap due to the other linked words in the co-occurrence graph.

Implementation Details

Our Wikipedia collection has approximately 5.2M documents.[7] We applied standard text normalization by converting all text to lowercase and removing stopwords as well as terms that do not meet term and document frequency thresholds of 100 and 5, respectively, while collecting the vocabulary. Moreover, we only keep edges between words that occur within a context window of 5 appearing over 200 times. The final co-occurrence matrix contains 39K words and 1.7M non-zero entries. We used the SparseLib++ library[8] to efficiently compute *PaintTeR* scores on this word co-occurrence graph via matrix operations (Haveliwala et al. 2003). The QG and RC models used in *ArtQuest* are from ProphetNet and AllenNLP, respectively.[9] We chose them based on their state-of-the-art performance on SQuAD (Rajpurkar et al. 2016), the widely-used dataset for QG and RC research. The text processing pipeline and baselines were implemented in Python v3.7.5 while the NLP taggers are from Stanza.[10]

[6] The appendix, resources, and code are made available for academic purposes at https://github.com/NUS-IDS/painter.

[7] Collected in Feb 2020

[8] https://math.nist.gov/sparselib++/

[9] https://github.com/microsoft/ProphetNet/tree/master/ProphetNet_En and https://demo.allennlp.org/

[10] https://stanfordnlp.github.io/stanza/

word	word (w)	$P_p(w)$	$P_u(w)$
predella	painting	0.564	0.011
foreshortening	paintings	0.269	0.008
triptychs	background	0.136	0.013
brushstroke	painter	0.133	0.006
brushwork	canvas	0.121	0.003
illusionistic	portrait	0.116	0.006
brushstrokes	composition	0.111	0.009
grisaille	landscape	0.084	0.007
impasto	panel	0.078	0.007
putto	foreground	0.065	0.001

Table 3: The top-10 words in our lexicon based on their Pointwise Mutual Information scores (left column) and association scores with painting documents (second column)

Precision	LO-BL	WN-BL	*PaintTeR*
k=1	0.360	0.440	**0.760**
k=3	0.467	0.613	**0.773**
k=5	0.472	0.624	**0.800**

Table 4: Span-extraction Performance on the *RAB* dataset.

PaintTeR	**WN-BL**	**LO-BL**
Baroque floral	focused on paintings	life paintings
wicker basket	life paintings	female artist
paint portraits	historical scenes	same time
floral still life	viewed as activities	focused on paintings
emptiness of life	botanical illustrations	subject matter

Table 5: Top spans extracted for the passage from Table 1

Results and Observations

Lexicon Extraction From our Wikipedia collection, the number of documents assigned the "Category:Painting" label was 5.2K. We applied the standard text processing pipeline which includes filters based on stopwords, term and document frequency thresholds (2 and 3, respectively). In addition, only words with parts-of-speech from nouns, adjectives, and adverbs are retained in this step. After filtering based on association probabilities (**Methods** section), we obtained a list of about 6K words for our lexicon.

The distant supervision provided by document-level labels are harnessed effectively by our algorithm to yield a discriminative list of words closely associated with painting documents. For illustration, the top-10 words based on their Pointwise Mutual Information values (Church and Hanks 1990) are shown in the left column in Table 3 along with the list of top-10 words with the highest $P_p(w)$ values that are at least ten times greater than uniform probability across categories ($P_u(w)$). The top-scoring words based on PMI values describe various painting styles, poses, and genres whereas the words in the second column are indeed highly representative of painting passages. We evaluate our lexicon quality indirectly through performance of *PaintTeR* and *ArtQuest.*

Text Span Extraction For the *RAB* dataset, we manually examined all candidate text spans for their art-centeredness. That is, in the context of a given passage, a candidate text span is marked as 'positive' if it is descriptive of the subject, content, or artistic aspects of the painting and 'negative' otherwise. The performance of *PaintTeR* as well as the two baseline scorers is illustrated in Table 4 using Precision@K (where $k = 1, 3, 5$) for the *RAB* dataset.

As can be seen in Table 4, *PaintTeR* is able to combine the word co-occurrence graph effectively with the lexicon to choose more number of 'positive' text spans at top of the ranked list resulting in significantly better precision scores compared to the baselines. For anecdotal illustration, the top-5 spans extracted with the three scoring algorithms are shown for the passage from Table 1 in Table 5. Compared to the baselines that rank candidates such as "focused on paintings" and "same time" among the top spans, *PaintTeR*-selected spans are more reflective of the artistic content.

Question Generation A summary of annotator evaluation for the questions generated by the ProphetNet model using candidate spans as "answer cues" is shown in Table 6. Since we use a rating scale of three from the set "{No/0.0, Acceptable/0.5, Yes/1.0}", a higher score indicates a better performance. The table shows micro-averaged annotation scores over the top-5 questions from different articles ranked by the three scoring algorithms.

For both the datasets, the scores of questions generated using *PaintTeR*-spans are significantly better than the baselines on all measures highlighting the impact of choosing appropriate spans on the end-to-end performance of our pipeline. In particular, the "Relevance" score improvements range between 11-32% highlighting the effectiveness of both our random-walk based scoring function as well as the lexicon. Note that on the *RAB* dataset though the baseline "LO-BL" generates fewer art-centered questions (lower "Relevance" score), the generated questions in fact score better on the other dimensions. The high "Fluency" values for all scoring algorithms on this dataset indicate that the ProphetNet model performs rather well in generating natural-sounding, human-like questions.

We also note that the "Answerability" scores are often significantly higher than the "Correctness" scores indicating that though the answer is present in the passage, our RC module is unable to correctly extract the same. In this regard, model tuning for ensuring round-trip consistency (match between the answer cues in QG with the answers generated by RC) needs further investigation (Alberti et al. 2019).

Method	Ans	Flu	Rel	Corr
		RAB dataset		
PaintTeR	**0.943**	**0.933**	**0.837**	**0.897**
WN-BL	0.899	0.896	0.726	0.865
LO-BL	0.924	0.927	0.659	0.875
		GCG dataset		
PaintTeR	**0.500**	**0.698**	**0.510**	**0.510**
WN-BL	0.469	0.656	0.458	0.427
LO-BL	0.438	0.583	0.385	0.406

Table 6: Human Evaluation results. **Ans**: Answerability, **Flu**: Fluency, **Rel**: Relevance, **Corr**: Correctness

Overall, the expert annotation scores on the *GCG* dataset are lower than the crowd-annotated scores obtained for *RAB*.

While the best scores for *RAB* are, on average, closer to "1" indicating that most annotators choose "Yes", the best scores are closer to "0.5", on average, for the *GCG* dataset, indicating that "Acceptable" was the common choice. This difference in overall scores between the two datasets is probably indicative of what an expert perceives as fluent or art-centered compared to an "average person" since the writing styles of both sets of articles were similar from our examination. We hope to gain more insights on this aspect through a user study after *ArtQuest* is deployed.

Collectively, our results suggest that a reasonable performing, interactive interface can be designed using QG and RC modules for simple pedagogical contexts.

Related Work

We refer our readers to recent survey articles for an overview on the challenges, existing approaches, as well as evaluation metrics for the question generation and reading comprehension tasks (Pan et al. 2019; Zeng et al. 2020). For QG specifically, building on early research with attention and the basic encoder-decoder setup (Zhou et al. 2018), recent works have started exploring transformers (Chan and Fan 2019), variational encoders (Lee et al. 2020), reinforcement learning (Wang et al. 2020b), semantic information (Pan et al. 2020) and future n-gram prediction (Qi et al. 2020). However, research addressing content selection and answer unaware QG is still preliminary with some previous works employing supervision for training an answer span selection module alongside QG (Du and Cardie 2017; Subramanian et al. 2018) or by simply treating noun phrases and named entities as potential answer cues for QG (Lewis, Denoyer, and Riedel 2019; Kumar et al. 2019).

In contrast, we are the first to exploit content selection for QG through unsupervised, PageRank-style approaches. Previously, PageRank was employed on word graphs for solving keyphrase extraction, summarization and emotion detection (Wan and Xiao 2008; Mihalcea and Tarau 2004; Rozenshtein, Gollapalli, and Ng 2020).

Conclusions

We presented *PaintTeR*, our novel, graph-based algorithm for scoring art-centeredness of text spans. We also showed how a lexicon of words highly associated with painting articles can be extracted automatically using distant supervision. Our experiments illustrate the effectiveness of *PaintTeR* in enabling art-centered QG for use in *ArtQuest*, our game-style interface for gallery visitors.

Our proof-of-concept *ArtQuest* prototype illustrates the practicality of building simple conversation support around artworks using zero context-specific training data and harnessing existing 'off-the-shelf' QG and RC modules and instead fine-tuning the input. Going forward, we will explore incorporating *ArtQuest*'s QG capabilities to enhance our Arthena chatbot and conduct user studies to evaluate it in operation. Given our low-resource context, we also would like to study some recent approaches such as data augmentation (Alberti et al. 2019), few-shot learning (Lewis, Denoyer, and Riedel 2019; Chen et al. 2020) and reinforcement learning using feedback from user studies (Wang et al. 2020b).

In this work, we only explored extractive questions whose answers can be found in the accompanying passages. In GLAM contexts however, abstractive questions that require reasoning from multiple passages (Mitra 2017) and reflective questions on artistic elements ("Can you imagine yourself..." or "Why do you think this color...") that trigger further perspectives for examining artworks form exciting research challenges for future investigation.

Acknowledgments

This research/project is supported by the National Research Foundation, Singapore under its Industry Alignment Fund – Pre-positioning (IAF-PP) Funding Initiative. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore.

Ethics Statement

This research was conducted in conformance with the AAAI Code of Professional Ethics and Conduct.

References

Alberti, C.; Andor, D.; Pitler, E.; Devlin, J.; and Collins, M. 2019. Synthetic QA Corpora Generation with Roundtrip Consistency. In *ACL*.

Anthony, D.; Smith, S. W.; and Williamson, T. 2009. Reputation and Reliability in Collective Goods: The Case of the Online Encyclopedia Wikipedia. *Rationality and Society*.

Boiano, S.; Borda, A.; Gaia, G.; Rossi, S.; and Cuomo, P. 2018. Chatbots and New Audience Opportunities for Museums and Heritage Organisations. In *Proceedings of the Conference on Electronic Visualisation and the Arts*.

Budanitsky, A.; and Hirst, G. 2006. Evaluating WordNet-based Measures of Lexical Semantic Relatedness. *Computational Linguistics*.

Chan, Y.-H.; and Fan, Y.-C. 2019. A Recurrent BERT-based Model for Question Generation. In *EMNLP Workshop on Machine Reading for Question Answering*.

Chen, Z.; Eavani, H.; Chen, W.; Liu, Y.; and Wang, W. Y. 2020. Few-Shot NLG with Pre-Trained Language Model. In *ACL*.

Church, K. W.; and Hanks, P. 1990. Word Association Norms, Mutual Information, and Lexicography. *Computational Linguistics*.

Du, X.; and Cardie, C. 2017. Identifying Where to Focus in Reading Comprehension for Neural Question Generation. In *EMNLP*.

Duan, J.; Cui, J.; Wu, M.; and Wang, H. 2018. Capturing Semantic Similarity for Words in Wikipedia with Random Walk. In *International Conference on Cloud Computing and Intelligence Systems (CCIS)*.

Gao, Y.; Bing, L.; Chen, W.; Lyu, M.; and King, I. 2019. Difficulty Controllable Generation of Reading Comprehension Questions. In *IJCAI*.

Gattani, A.; Lamba, D. S.; Garera, N.; Tiwari, M.; Chai, X.; Das, S.; Subramaniam, S.; Rajaraman, A.; Harinarayan, V.; and Doan, A. 2013. Entity Extraction, Linking, Classification, and Tagging for Social Media: A Wikipedia-Based Approach. *Proc. VLDB Endow.*

Haveliwala, T.; Kamvar, S.; and Jeh, G. 2003. An analytical comparison of approaches to personalizing pagerank. Technical report, Stanford.

Haveliwala, T.; Kamvar, S.; Klein, D.; Manning, C.; and Golub, G. 2003. Computing PageRank using power extrapolation. Technical report.

Haveliwala, T. H. 2003. Topic-sensitive pagerank: A context-sensitive ranking algorithm for web search. *TKDE*.

Huang, Q.; Fu, M.; Mo, L.; Cai, Y.; Xu, J.; Li, P.; Li, Q.; and Leung, H.-f. 2021. Entity Guided Question Generation with Contextual Structure and Sequence Information Capturing. *AAAI*.

Ji, G.; Liu, K.; He, S.; and Zhao, J. 2017. Distant Supervision for Relation Extraction with Sentence-Level Attention and Entity Descriptions. *AAAI*, 31.

Kim, Y.; Lee, H.; Shin, J.; and Jung, K. 2019. Improving Neural Question Generation Using Answer Separation. In *AAAI*.

Kumar, V.; Muneeswaran, S.; Ramakrishnan, G.; and Li, Y.-F. 2019. ParaQG: A System for Generating Questions and Answers from Paragraphs. In *EMNLP*.

Landis, J.; and Koch, G. 1977. The measurement of observer agreement for categorical data. In *Biometrics*.

Lee, D. B.; Lee, S.; Jeong, W. T.; Kim, D.; and Hwang, S. J. 2020. Generating Diverse and Consistent QA pairs from Contexts with Information-Maximizing Hierarchical Conditional VAEs. In *ACL*.

Lewis, P.; Denoyer, L.; and Riedel, S. 2019. Unsupervised Question Answering by Cloze Translation. In *ACL*.

Lindberg, D.; Popowich, F.; Nesbit, J.; and Winne, P. 2013. Generating Natural Language Questions to Support Learning On-Line. In *European Workshop on Natural Language Generation*.

Mihalcea, R.; and Tarau, P. 2004. TextRank: Bringing Order into Text. In *EMNLP*.

Miller, G. A. 1995. WordNet: A Lexical Database for English. *Commun. ACM*.

Mintz, M.; Bills, S.; Snow, R.; and Jurafsky, D. 2009. Distant supervision for relation extraction without labeled data. In *ACL*.

Mitra, R. 2017. An Abstractive approach to Question Answering. *CoRR*, abs/1711.06238.

Nguyen, D. P. T.; Matsuo, Y.; and Ishizuka, M. 2007. Relation Extraction from Wikipedia Using Subtree Mining. In *AAAI*.

Page, L.; Brin, S.; Motwani, R.; and Winograd, T. 1999. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab.

Pan, L.; Lei, W.; Chua, T.; and Kan, M. 2019. Recent Advances in Neural Question Generation. *CoRR*, abs/1905.08949.

Pan, L.; Xie, Y.; Feng, Y.; Chua, T.-S.; and Kan, M.-Y. 2020. Semantic Graphs for Generating Deep Questions. In *ACL*.

Qi, W.; Yan, Y.; Gong, Y.; Liu, D.; Duan, N.; Chen, J.; Zhang, R.; and Zhou, M. 2020. ProphetNet: Predicting Future N-gram for Sequence-to-Sequence Pre-training. In *EMNLP Findings*.

Rajpurkar, P.; Zhang, J.; Lopyrev, K.; and Liang, P. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In *EMNLP*.

Ram, O.; Kirstain, Y.; Berant, J.; Globerson, A.; and Levy, O. 2021. Few-Shot Question Answering by Pretraining Span Selection. In *ACL*.

Rozenshtein, P.; Gollapalli, S. D.; and Ng, S.-K. 2020. ESTeR: Combining Word Co-occurrences and Word Associations for Unsupervised Emotion Detection. In *EMNLP Findings*.

Schaffer, S.; Gustke, O.; Oldemeier, J.; and Reithinger, N. 2018. Towards chatbots in the museum. In *mobileCH@Mobile HCI*.

Staiano, J.; and Guerini, M. 2014. Depeche Mood: a Lexicon for Emotion Analysis from Crowd Annotated News. In *ACL*.

Strien, D. v.; Bell, M.; McGregor, N. R.; and Trizna, M. 2021. An Introduction to AI for GLAM. In *PMLR Teaching in Machine Learning Workshop*.

Subramanian, S.; Wang, T.; Yuan, X.; Zhang, S.; Trischler, A.; and Bengio, Y. 2018. Neural Models for Key Phrase Extraction and Question Generation. In *ACL Workshop on Machine Reading for Question Answering*.

Tuan, L. A.; Shah, D. J.; and Barzilay, R. 2020. Capturing Greater Context for Question Generation. In *AAAI*.

Wan, X.; and Xiao, J. 2008. Single Document Keyphrase Extraction Using Neighborhood Knowledge. In *AAAI*.

Wang, J.; Liu, J.; Bi, W.; Liu, X.; He, K.; Xu, R.; and Yang, M. 2020a. Improving Knowledge-Aware Dialogue Generation via Knowledge Base Question Answering. *AAAI*.

Wang, L.; Xu, Z.; Lin, Z.; Zheng, H.; and Shen, Y. 2020b. Answer-driven Deep Question Generation based on Reinforcement Learning. In *COLING*.

Witten, I.; and Milne, D. 2008. An effective, low-cost measure of semantic relatedness obtained from Wikipedia links. In *AAAI Workshop on Wikipedia and Artificial Intelligence: an Evolving Synergy*.

Xie, Y.; Yang, W.; Tan, L.; Xiong, K.; Yuan, N. J.; Huai, B.; Li, M.; and Lin, J. 2020. Distant Supervision for Multi-Stage Fine-Tuning in Retrieval-Based Question Answering. In *WWW*.

Zeng, C.; Li, S.; Li, Q.; Hu, J.; and Hu, J. 2020. A Survey on Machine Reading Comprehension: Tasks, Evaluation Metrics, and Benchmark Datasets. *CoRR*, abs/2006.11880.

Zhang, Z.; Yang, J.; and Zhao, H. 2021. Retrospective Reader for Machine Reading Comprehension. *AAAI*.

Zhou, Q.; Yang, N.; Wei, F.; Tan, C.; Bao, H.; and Zhou, M. 2018. Neural Question Generation from Text: A Preliminary Study. In *Natural Language Processing and Chinese Computing*, 662–671. Cham: Springer International Publishing.