Query Generation for Multimodal Documents

Kyungho Kim[1], **Kyungjae Lee**[1], **Seung-won Hwang**[2]*,
Young-In Song[3] and **Seungwook Lee**[3]
[1]Yonsei University, [2]Seoul National University, [3]NAVER
[1]{ggdg12, lkj0509}@yonsei.ac.kr, [2]seungwon.hwang@gmail.com,
[3]{song.youngin, swook.lee}@navercorp.com

Abstract

This paper studies the problem of generating likely queries for multimodal documents with images. Our application scenario is enabling efficient "first-stage retrieval" of relevant documents, by attaching generated queries to documents before indexing. We can then index this expanded text to efficiently narrow down to candidate matches using inverted index, so that expensive reranking can follow. Our evaluation results show that our proposed multimodal representation meaningfully improves relevance ranking. More importantly, our framework can achieve the state of the art in the first-stage retrieval scenarios.

1 Introduction

As more documents on the web are generated and consumed by mobile devices with cameras, documents are often multimodal, containing information in both text and image modalities. This poses a new challenge of finding relevance documents across modalities. More formally, the relevance of document, consisting of text t and image i, to the given query keywords q, should be modeled as a trimodal function $f(q, t, i)$, rather than a simple lexical match between q and t (or, **BM25** baseline) assuming the semantics of image i is fully represented by the surrounding text (or, **paired-text** assumption).

Prior research observes that paired-text assumption is often violated (Henning and Ewerth, 2017) – for example, some semantics can be better captured in image modality, and may not (or, cannot) be described in text. Meanwhile, BM25 baseline (Robertson et al., 1994) would fail to serve queries for such semantics.

To overcome this limitation of BM25, one may model relevance from query as a trimodal function $f(q, t, i)$ (Nian et al., 2017; Kordan and Kotov, 2018) instead, but they require runtime invocation of f for the given query q with all potential document matches. This incurs a prohibitive runtime overhead, unacceptable for search engines finding results online. A common practice is to use a cheap BM25 ranking as a "first-stage retrieval", efficiently supported by inverted index, to quickly narrow down to a few candidate documents, then evaluate $f(q, t, i)$. However, due to simple nature of BM25 using exact term matching, a document will be missed, if the query term is absent in t, even though it semantically matches image or another term in the text.

Our contribution is to keep first-stage retrieval as efficient as BM25, but enable multimodal semantic matching, using Query Generation **(QG)** before indexing. More specifically, we generate a likely query q from a joint modeling of t and i, to create an expanded text $t' = q \cup t$ such that (q, t') pair has more lexical overlaps, or better paired than (q, t), for first-phase retrieval. Specifically, we train a sequence-to-sequence model, such that given the representation of multimodal document, this model generates possible queries that users may ask to retrieve such document. This is analogous to doc2query (Nogueira et al., 2019) approach used for first-stage retrieval of textual relevance ranking, though this model, dealing with text modality only, cannot apply to our problem of retrieving multimodal documents.

For such multimodal representation for **QG**, a naive baseline is bimodal representation shown in Figure 1a: We may assume (t, i), even when lexical overlaps are low, is semantically paired in the embedding space. Note this is a relaxed version of paired-text assumption. Given this relaxed assumption, common architecture of bimodal representa-

*correspond to seungwon.hwang@gmail.com

Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics, pages 659–668
April 19 - 23, 2021.

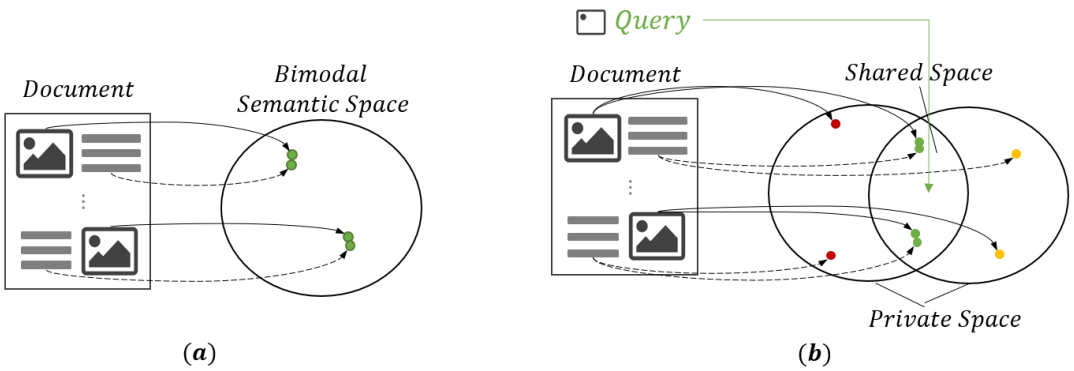

Figure 1: (a) bimodal and (b) trimodal representation for QG, query is used for disentangling shared and private space and relaxing paired-text assumption of bimodal representation.

tion for QG consists of the encoders for image i and text t. Each generates vector representation, which is later fused into a multimodal space, then decoded into a textual caption. Specifically, we consider two strong baselines: (a) cross-modal representation, pretrained from a large-scale paired corpus of image and caption, such as LXMERT (Tan and Bansal, 2019), ViLBERT (Lu et al., 2019), and VisualBERT (Li et al., 2019), finetuned for our task, and (b) memory network structure (Park et al., 2017). Based on these baselines, we design Bimodal QG combining the advantages of the two as a strong baseline. Then, we extend into Trimodal QG leveraging text, image and query (q, t, i).

Alternatively, we may further relax paired-text assumption and propose Trimodal QG in Figure 1b: (t, i) can be partially paired, where some semantics is conveyed in one modality. To deal with that challenge, the query given at training, helps "disentangle" shared and private semantics as additional loss terms. Another role of query is improving image representation, to de-emphasize semantics not discussed in either text or query.

In summary, our contributions are as follows:

- We study QG for multimodal documents, as an enabler for efficient first-stage ranking.
- We build a multi-task model, for query generation and representation learning, to generate effective queries for offline indexing.
- We improve the QG model by considering query as third modality in order to work well without paired-text assumption.
- We validated that our model outperforms all baselines in both public dataset and real-life web search query logs and quality annotation for multimodal documents.

2 Related Work

Our work is closely related to the following three areas of research.

2.1 Web search with images

Most efficient way to treat multimodal document ranking has been making paired-text assumption (Coelho et al., 2004; Azilawati and Meriam, 2008), such that simply matching q with t is sufficient. Our work is as efficient, by incurring no additional runtime overhead, but does not make such assumption. Alternatively, Rodríguez-Vaamonde et al. (2015) adds a reranking phase, checking if the images are relevant to the query, supervised by whether the given image is correlated with clicks.

Our distinction: We do not build on paired-text assumption, and can be viewed as generating a better-paired document by adding likely queries.

2.2 Image captioning

Another closely related work is the task of generating textual captions to the given query. As overviewed in Section 1, state-of-the-art models include bimodal joint representation of image i and text t (Kiros et al., 2014b; You et al., 2016; Park et al., 2017). Alternatively, such joint models can also be transferred from pretrained models, such as LXMERT (Tan and Bansal, 2019), ViLBERT (Lu et al., 2019), and VisualBERT (Li et al., 2019). Section 3 will compare and contrast these two approaches, and discuss why these models are

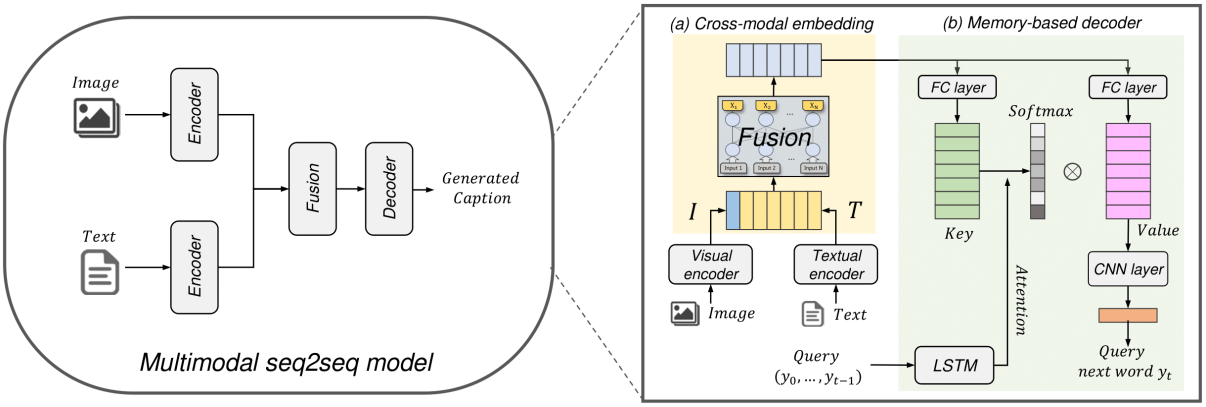

Figure 2: Proposed bimodal QG baseline, combining (a) cross-modal and (b) memory-based state-of-the-arts

limited for our task setting. (Jeon et al., 2003) is a non-neural model trained to annotate images with textual description, though requiring expensive supervision of segmented image with term.

Our distinction: We propose and validate trimodal joint representation for higher-quality captioning. Meanwhile, we do not require segment-level annotation, though our query-guided trimodal image representation naturally emphasizes important segments.

2.3 QG for first-stage retrieval

QG for first-stage text retrieval was pioneered by doc2query (Nogueira et al., 2019), generating likely queries for the document for indexing purposes in text modality. Our work can be viewed as query generation for multimodal documents: For each document, the task is to predict a set of likely queries. We train a sequence-to-sequence transformer using a data set of (query, relevant document) pairs. Alternatively, inverted index can be built for a latent term (Zamani et al., 2018), though it cannot be human-interpreted or reweighed. In contrast, we focus on inverted index on actual terms, as it is human interpretable and combines more naturally with legacy tf-idf ranker and reweighting module.

Our distinction: We validate the effectiveness of trimodal QG over bimodal state-of-the-art methods.

3 Bimodal Baselines: LXMERT and Memory-Based Generator

This section compares and contrasts two bimodal baselines: LXMERT [1] and Memory-Based Generator encode text t and image i into vector representations (Section 3.1), then the two are aggregated into a multimodal representation (Section 3.2), such that this vector can feed a decoder to generate a text sequence (Section 3.3). Specifically, we build **Bimodal QG** baseline, combining LXMERT representation and Memory-Based Generator decoding. Figure 2 shows overall architecture of our Bimodal QG baseline.

3.1 Text and image encoder

LXMERT and Memory-Based Generator generate text and image vectors, using transformer and memory network structure respectively. Both can be explained as **key** memory, aggregating **value** memory representation with proper self-attention, denoted as key and val, respectively, following the conventions of prior literature (Sukhbaatar et al., 2015).

Formally, we encode textual context words $C = \{w_1, w_2, ..., w_j\}$ obtained by concatenating the $n-$dimensional word embedding ($w \in R^n$) of the

[1] Out of cross-modal representations discussed in Section 1, we empirically found LXMERT performs the best in our problem setting and thus adopt it as a baseline.

top-j words with the highest TF-IDF weights [2].

$$T_i^{key} = \text{ReLU}(W_1 w_i + b_1), \quad i \in [1, j]$$
$$T_i^{val} = \text{ReLU}(W_2 w_i + b_2), \quad i \in [1, j]$$
$$T^{key} = [T_1^{key}; ...; T_j^{key}]$$
$$T^{val} = [T_1^{val}; ...; T_j^{val}]$$

where $W_1, W_2 \in R^{m \times n}$, $b_1, b_2 \in R^m$ are trainable linear transformation parameters where m is the dimension of memory. When parameter, such as T or I, is denoted without superscript (key or val), it refers both key and val vectors.

Similarly, an image input $U \in R^{2048}$ is generated from pool5 feature vector of Resnet-101 CNN encoder, which is similarly embedded into:

$$I^{key} = \text{ReLU}(W_3 U + b_3)$$
$$I^{val} = \text{ReLU}(W_4 U + b_4)$$

where $W_3, W_4 \in R^{m \times 2048}$ and $b_3, b_4 \in R^m$ are trainable matrices for tuning on given dataset. Final image embedding is generated with key and value vector, following the convention of attention network (Kiros et al., 2014a). Note only the representation of text and image is used at this point, and using query representation will be discussed in Section 4. Also, LXMERT extract image features by using Faster-RCNN (Ren et al., 2015) and transformer structure, which is pre-trained on large dataset combined by MSCOCO (Lin et al., 2014) and Visual Genome (Krishna et al., 2016). Therefore, both methods can be applicable on general images.

3.2 Multimodal fusion

The goal of multimodal fusion module in Figure 2 is to get text and image representations as input, and create a joint representation as output. When paired-text assumption holds (Figure 1a), this can be achieved by simply concatenating or adding two input modalities, and the future layers will be tuned for a proper alignment of the two. However, such concatenation is less effective when paired-text assumption does not hold as in Figure 1b.

One solution is transfer learning from pretrained joint representation model trained from a large-scale paired resources, such as LXMERT (Tan and Bansal, 2019). LXMERT is a transformer-based architecture for cross-modal representation learning, for predicting masked words from the text modality, and vice versa. This auxiliary task, known as masked cross-modality language model, helps building connections across modalities. In our problem setting, this option can be considered for public English dataset, as pre-trained LXMERT with a large scale paired training resources is readily available to generate a joint representation M_{lxm}, to replace simple concatenated fusion embedding M_{fusion} ($[I; T]$).

3.3 Bimodal QG: Joint text decoding for query generation

We now observe the two baselines: LXMERT, focuses on the problem of joint representation, but does not consider a decoder of generating a query sequence from such representation (Figure 2a). Meanwhile, Memory-Based Generator has the advantage of tightly coupling the key-value encoder and CNN decoder, by concatenating the representations of all modalities, co-attended based on the query keywords generated thus far, as illustrated in Figure 2b. This multimodal vector is calculated in each time step and used to decode the next word, which is an effective decoder design adopted for our model:

$$M_{total} = [I; T; Q]$$

With this joint representation, query generation is predicting the output probability of the next word among all vocabularies, by a convolution neural network, denoted as CNN in Figure 2.

For combining with the strength of LXMERT, we can simply replace the cross modal embedding vector in Figure 2 by M_{lxm} for bimodal QG. Alternatively, for ablation purpose, M_{lxm} can directly decoded without memory-based decoder, which we denote as LXMERT QG. Our final loss of bimodal captioning is a seq2seq loss.

$$L_{seq} = -\sum_{t=0}^{l} log(P(y_t | y_{<t}, I, T))$$

where t is the time step and l is the length of caption.

4 Trimodal QG: Query-aware representation

Our proposed bimodal QG partially contributes to relax paired-text assumption, but neither image

[2] C can be a subset of arbitrary size from t, which we empirically tune to $j = 30$.

and text representation is aware of queries. We argue that, queries carry rich semantics and contribute significantly to to relax paired-text assumption, specified as two key contributions C1 and C2 below.

- C1: We use query to improve image and text representation to disentangle into shared/private semantics where q matches the shared semantics.
- C2: As query generation is better trained when t and i are paired, we revise image representation to enhance pairedness with given query.

Motivated, we propose two new loss functions L_1 and L_2, addressing C1 and C2 respectively.

4.1 Query-aware relevance

For addressing **C1**, we model "private" parts of image and text, denoted as P_i and P_t, to relax the paired-text assumption. Our goal is to build joint representation S, aligning only the shared part of image and text, with query q.

$$S = W_5[I^{key}; T^{key}] + b_5$$
$$P_i = W_6 I^{key} + b_6$$
$$P_t = W_7 T^{key} + b_7$$

where $W_5, W_6, W_7 \in R^{m \times 2m}$ and $b_5, b_6, b_7 \in R^m$ are trainable parameters. I^{key} and T^{key} are the image and text input vectors, respectively. These inputs are concatenated into $[I^{key}, T^{key}]$ and multiplied by W_5 so that the combined modality can be projected into same semantic space with private vectors P_i and P_t.

To ensure this joint representation to project closely to the representation of query, query embedding vector Q_v is generated by LSTM with generated query $y_0, ..., y_{t-1}$:

$$Q_v = \mathrm{LSTM}(Q)$$

with the objective loss to keep private vectors away from query, and shared close to query:

$$L_1^{img} = \max\{0, r - (\mathrm{sim}(Q_v, S) - \mathrm{sim}(Q_v, P_i))\}$$
$$L_1^{text} = \max\{0, r - (\mathrm{sim}(Q_v, S) - \mathrm{sim}(Q_v, P_t))\}$$
$$L_1 = L_1^{img} + L_1^{text}$$

where r is the margin parameter. This margin enables our model to relax the decision function in LXMERT, predicting whether t and i are paired, as a binary classification. Unlike such binary prediction, computing zero or one score for partially paired pair (Figure 1b), the above two losses compute a scalar score and make a soft decision based on similarity. The query-aware relevance loss L_1 is defined by combining the two losses for each modality.

4.2 Query-aware alignment

For **C2**, we revise image representation to highlight query-related semantics, to make it semantically pair better with text representation. To reflect a (possibly nonlinear) relation with the query and the image, a fully connected neural network is applied to each modality before gating. Formally, query-aware image embedding V_p is described below:

$$A_q = \sigma(w_8 Q_v + b_8)$$
$$V_p = (w_9 I^{key} + b_9) \odot A_q$$

where Q_v is the query embedding, A_q is the attention derived from query with sigmoid σ, $\odot$ means element-wise multiplication and $w_9 \in R^{m \times m}$ and $b_9 \in R^m$ are trainable parameters. $w_8 \in R^{m \times m}$ and $b_8 \in R^m$ are learned with m-dimension query embedding. We apply the fully connected layer (first term before element-wise multiplication) to project image near query embedding.

We apply query-aware image representation to the multimodal space learning:

$$L_2 = \max\{0, r - (\mathrm{sim}(V_p, T_+^{key}) - \mathrm{sim}(V_p, T_-^{key}))\}$$

where r is the margin and $+$ means positive text where text belongs into same document with given image and $-$ means negative text from different document.

Finally, we combine the two loss functions as a final query-aware loss L_q:

$$L_q = L_1 + L_2$$

Our final loss of trimodality captioning is the sum of seq2seq loss and query aware alignment loss:

$$L_{final} = L_{seq} + L_q$$

	R@1	R@10	R@30
BM25	0.166	0.672	0.809
LXMERT QG	0.175	0.684	0.813
Bimodal QG	0.207	0.718	0.816
Ours (Trimodal QG)	**0.213**	**0.723**	**0.823**

Table 1: Public dataset results for first-stage retrieval

5 Experiment

The goal of our evaluations is to validate the effectiveness of our approach in public dataset and real-world Web search queries and settings. In particular, we have two research questions:

- **RQ1** Would QG task benefit from LXMERT model? How does our approach compare with BM25 or Bimodal for first-stage retrieval?
- **RQ2** Would our approach improve real-life Web search queries?

We use public dataset for RQ1 and real-life queries and quality annotation for ad-hoc web search task from NAVER for RQ2.

5.1 RQ1: Public reproducible scenarios

5.1.1 Dataset

As there is no public dataset with query workloads and multimodal documents, we repurpose a public dataset of instructional videos (Kim et al., 2020) by transforming videos into multimodal documents, which consists of 2000 query-video pairs, where each video is a recipe instruction from YouCook2 dataset[3]. We first sample an image corresponding to each sentence in the transcript, by capturing a center frame. As this may create too many (image,sentence) pairs, we propose to cluster into more natural boundaries using temporal and semantic aspects: A pair of successive frames, each with textual transcript and a set of objects [4], will be merged if more than $clip\%$ objects overlap, which is empirically tuned for each experiment. When $clip\%$ is set to 100%, it is our initial setting without merging, and this can be tuned to better fit the target scenario. Figure 3 shows the example of video clipping, where associated frames are merged into multimodal paragraphs with images and transcripts. The lengths of extracted multimodal paragraphs are from 1 to 20 according to video. Therefore it can correspond to short to long length of actual documents.

5.1.2 Experiment settings

To preprocess text and image input data, we use NLTK text tokenizer (Sukhbaatar et al., 2015) and Resnet-101 CNN respectively. When learning a query generator, the dimensionality of image and word embedding vector is set to 2048 (by following the size of pool5 vector of ResNet) and 100 respectively. The dimensionality of memory m and query embedding are empirically determined to 256. Mini-batch stochastic gradient descent method is used to learn our query generator. Specifically, we used Adam optimizer (Kingma and Ba, 2014) with the default setting. The initial learning rate is set as 0.001 and is divided by 1.2 at every five epoch until it reaches 30 epochs. The number of generated query is up to 8.

In this study, we follow a standard two-stage document retrieval scenario: First, top 30 candidate documents are ranked and selected from the index using a first-stage ranker, namely BM25, LXMERT, Bimodal, and Trimodal in Table 1. Then, the second-stage ranker follows, which is generally more sophisticated and expensive, such as BERT-based ranker (Nogueira et al., 2019). However, we stress that our work is orthogonal to second-stage ranker and focus on first-stage results.

5.1.3 Results

First-stage ranking results on this public English dataset are shown in Table 1. In this dataset, evaluation metrics is limited to R@K, due to binary nature of relevance annotation: the ratio of ground truth videos that appear in our top-K results, when $K = 30$ is returning all results. In real-life evaluation in the next section, graded relevance annotations will be collected to evaluate rank accuracy. BM25 in the table uses raw BM25 score on text t itself. The other QG models (LXMERT QG, Bimodal QG, and Trimodal QG) are implemented as BM25 scoring on expanded text t' with queries, generated from each QG model respectively. In all evaluations, relevance scores on t and t' are aggregated with linear weighting, which we empirically tune $\lambda = 0.9$ for t' (and $1 - \lambda$ for t). In all metrics, Bimodal and Trimodal outperform BM25 ranking, validating our hypothesis that considering image for joint document representation is effective.

Based on this result, our evaluations from this point on focus on evaluating Trimodal, with real-

[3] http://youcook2.eecs.umich.edu/

[4] extracted from each frame using Faster-RCNN (Ren et al., 2015)

(a) Raw video (b) Multimodal document

Figure 3: An example of video clipping to show how we transform a video into a multimodal document. Each multimodal document is clustered into paragraphs, with images and transcripts, shown as red boxes.

Video			
Query	Fried Chicken	Macaroni and Cheese	Mashed potato
Context	Chicken, Crunchy, Powder, Oil	Bread, Sandwich, Cheese, Melt	Potato, Butter, Mix, Smooth
Caption	Fried	Macaroni	Corn, Hummus

Figure 4: Qualitative examples demonstrating caption generation of our model on public dataset. Successful cases are highlighted by green. Failure case is highlighted by red.

life settings, allowing multilinguality, graded relevance annotation, and a realistic ranker.

5.1.4 Qualitative results

The example images, contexts, and captions are presented in Figure 4. It shows an improvement on search by generating queries for multimodal documents. An example of correctly generating query is shown in the first and second image of Figure 4. The images and contexts are highly related to the name of cooking, but it does not exist in the context where the words are selected by applying TF-IDF to transcript. In this case, our model could make a considerable contribution to search performance by directly generating the query itself like "fried" and "macaroni". The failure case is shown in the third image of Figure 4. The recipe for hummus and mashed potato both have a mashing step and similar-looking ingredient. If the cooking method and the appearance, color, and texture of the ingredients are similar, the model has a probability of generating other queries. As shown above, our model does great for generating query words to support first-level retrieval.

5.2 RQ2: Real-world ad-hoc web search scenarios

5.2.1 Dataset

The source dataset used in our experiments is the evaluation set of the web search ranking task from the real-life commercial search engine. This dataset contains about 28,000 queries, for each of which 60 document URLs from search engine results are found. In real-time commercial dataset, annotating the relevance of all query-document pairs is impractical. Instead, we pooled top 60 documents, as used widely in IR evaluation to reduce annotation efforts, where only top ranked documents from a small set of retrieval runs are manually assessed for relevance to investigate the impact of first-stage retrieval. These documents are labeled by expert query annotators into one of five graded relevance score, ranging from 1 (poor) to 5 (excellent), or left unlabelled. Since unlabelled documents were randomly sampled from low ranks of search results, we treat all unlabelled documents as irrelevant ones (score 1). Additionally each query is classified into domains by NAVER.we evaluate real world dataset in such experiment setting.

5.2.2 Experiment settings

Out of all domain areas, we observe five main categories where the image information is expected to complement missing information from text–namely, Fashion, Place, Entertainment, Commerce, and Food/Recipe. We select query-document pairs annotated as described above for these categories. More specifically, Table 2 shows the selected categories and the number of queries in each category.

As a realistic ranker, we replace BM25 and train LambdaMart, as implemented in LightGBM (Ke et al., 2017; Meng et al., 2016), which is gradient boosting framework developed by Microsoft that uses tree based learning algorithms. The ranking model is trained and tested separately for each

Category	# of queries
Fashion	119
Entertainment	347
Food/Recipe	205
Place	971
Commerce	1665

Table 2: Statistics of categories selected from real-life data

category. For evaluation metrics, we follow the convention of prior work, to use NDCG@1, 5, 10, evaluated using a 5-fold cross-validation. A detailed description of the text features and image feature used to learn the ranking model is as follows.

To handle Korean text, we replace a tokenizer from KoNLPy [5]. Except this, all other experiment settings, including configurations to learn our query generator, remain unchanged from previous experiments.

BM25F score of query-document
BM25 score of query-document title
BM25 score of query-highlighted text
Exact matching of query-document
Query proximity score on document
Query proximity score on document title
Covered query term ratio of document title
Covered query term ratio of front section
Covered query term ratio of highlighted text

Table 3: Descriptions of real-life text features selected

Each document in real-life search engine is represented by hundreds of pre-computed features. Among them, we select nine widely used features related to textual similarity between a query and document. The selected features are shown in Table 3. Those features are used for the text baseline in Table 4.

5.2.3 Results

Table 4 reports accuracy gains in the five selected categories. For the four of five categories, our proposed approach achieved up to 10.8% gain on NDCG@1.

The category seeing the highest gain has been Food/Recipe, where images can be informative and complement textual instructions in this domain, as consistently observed empirically.

On the other hand, Commerce domain, though we expected showing the image of actual goods would complement text information, was the worst performing category. Our analysis shows that expert annotation was biased to highly rated official sites, while the same item can be sold in millions of sites with lower authority. Meanwhile, our models focusing on document relevance only, following the convention of ad-hoc retrieval scenarios, could not distinguish such difference.

Table 5 shows the search performance of each category when a document is ranked using only trimodal-aware image feature. The best search performance category is the Food/Recipe, which had the highest performance gain in Table 4, and the other categories show a similar performance. The score of ranking model using only image feature can achieve performance about 62% of that of using all features, with respect to NDCG@5.

Table 6 reports the accuracy gains of all categories over strong baselines. Only our trimodal query generation shows positive results on all domains. This demonstrates that our proposed query-aware trimodal loss contributes to capturing the query-relevant semantic of images.

6 Conclusion

We study the problem of representing a multimodal document to be indexable for efficient first-stage. Our contribution is posing the problem as trimodal QG to augment the given text, by proposing a trimodal joint representation of image, text, and query without paired-text assumption. We validate our approach over both public dataset and real-life web search data collected from commercial search engines.

Acknowledgements

This research was supported by the MSIT, under IITP-2017-0-01779; A machine learning and statistical inference framework for explainable artificial intelligence) and the ITRC support program (IITP-2021-2020-0-01789), supervised by the IITP.

References

Azmi Azilawati and Siti Meriam. 2008. Exploiting surrounding text for retrieving web images.

[5] Korean Natural Language Processing in python (Park and Cho, 2014)

Category	Model	NDCG@1		NDCG@5		NDCG@10	
Fashion	Text	0.5194	5.9%	0.6542	4.2%	0.7546	2.3%
	Trimodal QG	**0.5502***		**0.6819***		**0.7719***	
Place	Text	0.4806	6.0%	0.5772	4.1%	0.7031	2.0%
	Trimodal QG	**0.5093***		**0.6008****		**0.7175****	
Entertainment	Text	0.4009	10.8%	0.5695	4.8%	0.6916	3.3%
	Trimodal QG	**0.4441***		**0.5970***		**0.7144****	
Commerce	Text	**0.5117***	-3.9%	0.6182	0.1%	**0.7282**	-0.1%
	Trimodal QG	0.4917		**0.6190**		0.7277	
Food/Recipe	Text	0.4862	8.4%	0.6069	5.6%	0.7223	3.4%
	Trimodal QG	**0.5268***		**0.6411****		**0.7468****	

Table 4: First-stage results for real-life dataset. Percentage shows the rate of increase, compared with text-only feature. * and ** indicate statistically significant improvements at $p < 0.05$ level and $p < 0.01$ level respectively.

Category	NDCG@1	NDCG@5	NDCG@10
Fashion	0.2157	0.3616	0.5375
Place	0.2637	0.3778	0.5398
Entertainment	0.2581	0.3878	0.5512
Commerce	0.2539	0.3843	0.5477
Food/Recipe	0.3837	0.4473	0.5904

Table 5: Ablation study of using Trimodal feature only.

Model	NDCG@1	NDCG@5	NDCG@10
LXMERT QG	-0.30%	-0.12%	-0.09%
Bimodal QG	-0.25%	-0.08%	-0.05%
Trimodal QG	+0.30%	+0.12%	+0.08%

Table 6: Result of real-life dataset on all domain category. Numbers show the percentage rate of increases, when compared to using text-only feature.

Tatiana A. S. Coelho, Pável Calado, Lamarque Vieira Souza, Berthier A. Ribeiro-Neto, and Richard R. Muntz. 2004. Image retrieval using multiple evidence ranking. *IEEE Transactions on Knowledge and Data Engineering*, 16:408–417.

Christian Andreas Henning and Ralph Ewerth. 2017. Estimating the information gap between textual and visual representations. In *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval*, ICMR '17, page 14–22, New York, NY, USA. Association for Computing Machinery.

Jiwoon Jeon, Victor Lavrenko, and R. Manmatha. 2003. Automatic image annotation and retrieval using cross-media relevance models. In *SIGIR '03*.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. Lightgbm: A highly efficient gradient boosting decision tree. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 3146–3154. Curran Associates, Inc.

Kyungho Kim, Kyungjae Lee, and Seung-won Hwang. 2020. Instructional video summarization using attentive knowledge grounding. In *Proceedings of the 2020 European Conference on Machine Learning*.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Ryan Kiros, R. Salakhutdinov, and R. Zemel. 2014a. Unifying visual-semantic embeddings with multimodal neural language models. *ArXiv*, abs/1411.2539.

Ryan Kiros, Ruslan Salakhutdinov, and Richard S. Zemel. 2014b. Multimodal neural language models. In *ICML*.

Saeid Balaneshin Kordan and Alexander Kotov. 2018. Deep neural architecture for multi-modal retrieval based on joint embedding space for text and images. In *WSDM '18*.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, Michael Bernstein, and Li Fei-Fei. 2016. Visual genome: Connecting language and vision using crowdsourced dense image annotations.

Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. 2019. Visualbert: A simple and performant baseline for vision and language. *ArXiv*, abs/1908.03557.

Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *ArXiv*, abs/1908.02265.

Qi Meng, Guolin Ke, Taifeng Wang, Wei Chen, Qiwei Ye, Zhi-Ming Ma, and Tie-Yan Liu. 2016. A communication-efficient parallel algorithm for decision tree. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 1279–1287. Curran Associates, Inc.

Fudong Nian, Bing-Kun Bao, Teng Li, and Changsheng Xu. 2017. Multi-modal knowledge representation learning via webly-supervised relationships mining. In *Proceedings of the 25th ACM International Conference on Multimedia*, MM '17, page 411–419, New York, NY, USA. Association for Computing Machinery.

Rodrigo Nogueira, Wei Yang, Jimmy Lin, and Kyunghyun Cho. 2019. Document expansion by query prediction. *ArXiv*, abs/1904.08375.

Cesc Chunseong Park, Byeongchang Kim, and Gunhee Kim. 2017. Attend to you: Personalized image captioning with context sequence memory networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6432–6440.

Eunjeong L. Park and Sungzoon Cho. 2014. Konlpy: Korean natural language processing in python. In *Proceedings of the 26th Annual Conference on Human & Cognitive Language Technology*, Chuncheon, Korea.

Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:1137–1149.

S. Robertson, S. Walker, Susan Jones, M. Hancock-Beaulieu, and Mike Gatford. 1994. Okapi at trec-3. In *TREC*.

Sergio Rodríguez-Vaamonde, Lorenzo Torresani, and Andrew W. Fitzgibbon. 2015. What can pictures tell us about web pages? improving document search using images. *IEEE transactions on pattern analysis and machine intelligence*, 37 6:1274–85.

Sainbayar Sukhbaatar, Arthur Szlam, Jason Weston, and Rob Fergus. 2015. Weakly supervised memory networks. *CoRR*, abs/1503.08895.

Hao Hao Tan and Mohit Bansal. 2019. Lxmert: Learning cross-modality encoder representations from transformers. In *EMNLP/IJCNLP*.

Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo. 2016. Image captioning with semantic attention. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4651–4659.

Hamed Zamani, Mostafa Dehghani, W. Bruce Croft, Erik G. Learned-Miller, and Jaap Kamps. 2018. From neural re-ranking to neural ranking: Learning a sparse representation for inverted indexing. *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*.