2021 IEEE 37th International Conference on Data Engineering (ICDE) | 978-1-7281-9184-3/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICDE51399.2021.00226

Description Generation for Points of Interest

Meng Zhou[†]
Academy for Advanced Interdisciplinary Studies
Peking University, China
zhoumeng15@pku.edu.cn

Jingbo Zhou[†*]
Business Intelligence Lab
Baidu Research, China
zhoujingbo@baidu.com

Yanjie Fu
Department of Computer Science
University of Central Florida,United States
yanjie.fu@ucf.edu

Zhaochun Ren
School of Computer Science and Technology
Shandong University, China
zhaochun.ren@sdu.edu.cn

Xiaoli Wang
School of Software
Xiamen University, China
xlwang@xmu.edu.cn

Hui Xiong
Information Systems Department
Rutgers University, United States
xionghui@gmail.com

***Abstract*—Description of Points of Interest (POIs) plays an important role to enhance the quality of many location-based services, such as displaying concentrated information of POIs for user-friendly experience and leading to successful POI recommendation. However, only a few popular POIs have enough description on the web. Collecting or writing high-quality descriptions for many unpopular or long-tail POIs remains a huge challenge for online map services, especially considering there are numerous new appeared POIs every day. Unlike existing studies about automatic product description generation, the POI description is quite diverse across different locations over a country, and requires high expert knowledge. To address this issue, we first study the POI description generation problem by proposing a novel model, named as Multi Mode Description Generator (MMDG), to automatically generate description based on POIs' reviews and other features. To extract key information for POI description generation, MMDG is equipped with a multi-mode encoder and a transformer-based decoder. Besides user reviews, the multi-mode encoder also considers the category and spatial context information of target POIs, and integrate them with a fusion function. We have conducted an extensive experimental evaluation on a large-scale real-world dataset to demonstrate its effectiveness and superiority over state-of-the-art baselines in terms of various metrics.**

***Index Terms*—Point of Interest, Description Generation**

I. Introduction

A high-quality description is critical to help user understand the functionality of a Point of Interest (POI). Here POI refers to a specific point location on an online map, like a restaurant, a shop and a park. The description of POIs has great potential to improve the quality of location-based services [1]. For example, when a user is browsing a POI on an online map, the POI description can help user quickly capture the key information about the POI. Another useful application of POI description is to increase the user experience for POI recommendation. Traditional POI recommendation generated by a black-box model confronts a lack of persuasion [2]–[4]. An informative description may alleviate this dilemma. Many studies have reported that a high-quality description can have a substantial influence on recommendation results [5]–[7].

Whereas, a lot of POIs lack high-quality textual description in the real world. Though the description of very popular POIs can be crafted by users, it is hard to collect or write description for all POIs, especially considering that there are many unpopular and long-tail POIs with little information. According to our statistics, there are hundreds of thousands of new POIs appearing in China everyday, and many of them cannot live more than half a year. Thus, it is a challenging problem to automatically generate POI description for the online map service.

Traditional methods for product description generation mainly focus on the templates and statistical models [8], [9]. Due to the simplicity and practicality, template-based methods are still the main solution to description generation in many applications. Whereas, since these methods just express in the patterns which already existed in templates, the generated description is often dull and unnatural, making template-based methods inherently flawed.

With the recent advancement in deep learning based models in natural language processing, generating product description using neural networks becomes a promising solution. Classic Seq2Seq and transformer models, such as [10], [11], have been investigated in this task. However, as shown in our experimental evaluation section, these models perform well in product description generation task, but become degraded when they turn to POI description generation. At first, the main factor for such degradation is that POI has a unique spatial characteristic different from the general product. For example, the accessibility of a hotel is affected by traffic of its location; and the rate of the same food is quite different in different restaurants. Thus, the description of POI is diverse in different locations and requires high expert knowledge. Second, another challenging problem is that POI has multiple types of information, such as text (user reviews), category and location information. All the information is critical for description generation. Thus, compared to the general NLG task, it is more challenging to incorporate multiple types of information in POI description generation.

In this paper, we propose a novel **M**ulti **M**ode **D**escription **G**enerator (MMDG), which can generate POI description

[†] Meng Zhou and Jingbo Zhou contributed equally to this paper. This work is done when Meng Zhou was an intern at the Baidu Research.

[*] Jingbo Zhou is the corresponding author.

978-1-7281-9184-3/21/$31.00 ©2021 IEEE
DOI 10.1109/ICDE51399.2021.00226

from POIs' reviews with other POI features to address these shortcomings. Generally, the MMDG is built on the encoder-decoder structure with a multi-mode encoder and a transformer-based decoder. The multi-mode encoder and decoder are equipped with the transformer network [12] to extract the information from general reviews and output the description. In addition, since POIs are closely connected to their location, a Convolutional Neural Network (CNN) is applied in MMDG to reinforce its ability by incorporating spatial context information from nearby areas. Thus, we propose the Context-Aware Convolutional Neural Network (CACNN), which is the key component of the multi-mode encoder, to incorporate the spatial context information around the POI to facilitate the description generation.

Furthermore, we built a large-scale real-world dataset from Baidu Maps[1] which is one of the largest online map service platforms in China. We conducted a series of extensive experimental evaluations by various metrics, of which the results demonstrate that our novel MMDG model outperforms all existing state-of-the-art models in the POI description generation task.

In general, we can summarize our contributions as follows:

- We first investigate the POI description generation problem which aims to automatically generate high-quality description of POIs from user reviews and other POI features.
- We devise a novel end-to-end generation model, named MMDG, to generate high-quality description for POIs. The proposed method utilizes multiple types of information to improve the quality of generated description.
- We build a large-scale real-world dataset from an online map service, Baidu Maps. A set of extensive experimental evaluations demonstrate the effectiveness of our model, which shows that MMDG achieves a higher score than existing state-of-the-art baselines in several metrics including BLEU and ROUGE.

The rest of paper is organized as follow. Next, we discuss the related work in Section II, followed by a problem formulation and framework overview in Section III. Then we present the details of MMDG in Section IV. Finally, we evaluate our method in Section V, and conclude the paper in Section VI

II. Related Works

For a long time, the description generation for recommendation system was implemented by template-based model. These template-based models usually combined statistical models and templates to generate accurate description, such as [8], [9]. Though these template based methods are easy to practice, the limitation of the templates makes the generated description too rigid and dull. Thus, with the success of the seq2seq structure and attention mechanism in Natural Language Generation (NLG), template-based methods are seldom adopted in advanced text generation models.

The seq2seq neural network framework based on encoder-decoder was proved to be effective for many Natural Language Generation (NLG) problems in [13]. Bahdanau et al. [14] further adopted attention mechanism to reinforce seq2seq framework. Such attention-based Seq2Seq models have become very popular for NLG applications [15]–[19]. In recent years, there are also many works to use transformer [12] to build their seq2seq model instead of RNN [20]–[24].

These advanced neural network models for NLG have also been introduced for description generation. For example, Chen et al. [10] proposed a transformer-based model for product description generation with external knowledge. Zhang et al. [11] adopted a hybrid network to achieve pattern controlled description generation.

However, to the best of our knowledge, the POI description generation has not been studied systematically. Existing models do not perform well in POI description generation for two reasons. On the one hand, there is a lack of consideration of spatial information by existing methods. On the other hand, how to incorporate multiple types of information remains a great challenge for existing models. Motivated by these challenges, we design our MMDG model with a multi-mode encoder and a transformer-based decoder, which have not been investigated in previous studies.

III. Preliminaries And Overview

In this section, we first formulate the problem of POI description generation and present preliminaries for this paper. Then we briefly introduce a framework overview of MMDG.

A. Preliminaries

A Point of interest, or POI for short, is a specific point location that someone may find useful or interesting in online maps which is denoted as p_i hereafter. We use $P = \{p_i\}_{i=1}^{N}$ to denote a list of POIs. Each POI is labeled with a fine-grained category (e.g. Express Inn or Chinese food restaurant) in our dataset. We use a one-hot encoded vector $\boldsymbol{t}^{p_i}$ with 0-1 values to indicate the category of POI p_i where the dimension of $\boldsymbol{t}^{p_i}$ is the number of categories m (i.e. $|\boldsymbol{t}^{p_i}| = m$). In an online map service, each POI may have a set of reviews written by users. Given a review list of a POI $\{r_1, r_2, ..., r_u\}$,we concatenate these reviews into one long sequence $[r_1; r_2; , ...r_u]$, and denote the word sequence of POI p_i as $\boldsymbol{x^{p_i}} = \{x_1, x_2, ..., x_n\}$.

Considering that POIs are closely connected to their location, we also introduce spatial context information into our description generation task. We propose a novel method for context information extraction: compress the nearby map into a tensor $\boldsymbol{C}$ to retain spatial structure. The detailed discussion of context information extraction is presented in Section IV-C.

Given the above definitions, we finally formulate our problem as generating a description $\hat{\boldsymbol{y}}$ for a POI, based on its category $\boldsymbol{t}$, spatial context information $\boldsymbol{C}$ and reviews of the target POI $\boldsymbol{x}$. We measure the quality of a description by multiple automatic metrics which are discussed in section V.

[1] https://maps.baidu.com/

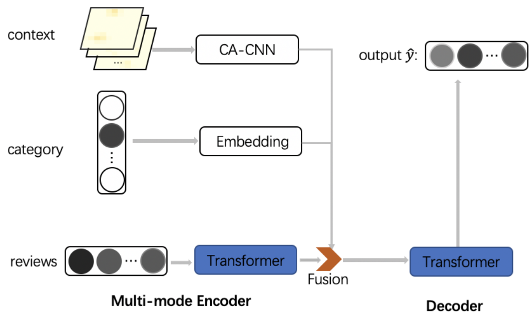

Fig. 1. An overview of the MMDG Model. MMDG is built on the encoder-decoder structure with a multi-mode encoder and a transformer-based decoder.

B. Framework Overview

Figure 1 shows an overview of our MMDG model. Composed of a multi-mode encoder and decoder, our model can extract information from reviews, category and spatial context information of POI, and then to generate description for target POI. As we see from Figure 1, the first part of our model is the proposed multi-mode encoder, which is composed with a transformer network, category embedding and Context-Aware Convolutional Neural Network (CACNN). The multi-mode encoder takes multiple types of information as input, and integrate them into a latent representation with the fusion function. Then, a transformer-based decoder takes the latent representation from multi-mode encoder as its input and generate a sequence of description for POI. Different from the multi-mode encoder, the decoder is only composed of a transformer network to generate description. In IV, we present more details about our proposed model.

IV. Model

In this section, we discuss more details on the transformer network, category embedding and CACNN, which are applied in our multi-mode encoder and decoder. The fusion function for information integration is also introduced in the end.

A. Transformer Networks

The transformer network [12], which is characterized by the self-attention mechanism, has been demonstrated effective in many NLG tasks. Considering the effectiveness of the transformer network, we apply transformer networks in our model to build not only the multi-mode encoder but also the decoder.

1) Multi-Mode Encoder: The transformer network is an important component of our multi-mode encoder as it can extract key information from massive user reviews of POI. In multi-mode encoder, we implement the transformer network with stacked one embedding layer and six self-attention layers. The transformer network takes the reviews of target POI $\boldsymbol{x} = \{x_1, x_2, ..., x_n\}$ as input, and calculates a latent encoded representation $\boldsymbol{e}_{rev}$ as its output. In transformer network, the input word sequence $\boldsymbol{x} = \{x_1, x_2, ..., x_n\}$ passes through the embedding layer and the six identical self-attention layers sequentially. The embedding layer converts any given word to a continuous vector. For example, given the input review words sequence $\boldsymbol{x} = \{x_1, x_2, \ldots, x_n\}$, the embedding layer returns a matrix $\boldsymbol{e} = \{e_1, e_2, \ldots, e_n\} \in R^{n \times d_{model}}$, of which each element e_j is a d_{model}-dim vector.

Then, the self-attention layers are applied to learn deeper representation where the dependency between words is considered. Given the input matrix $\boldsymbol{e} = \{e_1, e_2, \ldots, e_n\}$, the self-attention layer projects $\boldsymbol{e}$ to three different matrices: key $K \in R^{n \times d_{model}}$, value $V \in R^{n \times d_{model}}$ and query $Q \in R^{n \times d_{model}}$. These projections are implemented by three different linear functions as below:

$$K = \boldsymbol{e}W_K + b_K, V = \boldsymbol{e}W_V + b_V, Q = \boldsymbol{e}W_Q + b_Q, \quad (1)$$

where the parameters W_K, W_V, W_Q are matrices, b_K, b_V, b_Q are vectors. With K, V and Q, the self-attention equation can be formulated as below:

$$\alpha = softmax(\frac{QK^T}{\sqrt{d_{model}}}) \quad (2)$$

$$\boldsymbol{e}_{att} = \alpha \boldsymbol{e} \quad (3)$$

where α is the attention weight and $\boldsymbol{e}_{att}$ is the attention representation. We also implement a multi-head version of self-attention to enhance the transformer network. In multi-head attention, n_{head} self-attention functions (Equation (1)-(3)) are performed in parallel, and the output $\boldsymbol{e}_{att}$ is a concatenation of the attention representation in each-head

$$\boldsymbol{e}_{att} = \{\boldsymbol{e}_{att}^{(i)}\}_{i=1}^{n_{head}} \quad (4)$$

Then, the attention representation $\boldsymbol{e}_{att}$ is sent to a Point-Wise Feed-Forward Network (FFN) to get the output representation $\boldsymbol{e}_{next}$. The output $\boldsymbol{e}_{next}$ is also the input of next self-attention layer.

$$\boldsymbol{e}_{next} = FFN(\boldsymbol{e}_{att}) = W_2 ReLU(W_1 \boldsymbol{e}_{att} + b_1) + b_2 \quad (5)$$

where the parameters W_1, W_2 are matrices and the parameters b_1 and b_2 are vectors. Note that there are six self-attention layers in our encoder. We use $\boldsymbol{e}_{rev}$ to denote the output of the last self-attention layer, which is also the output of the transformer network.

2) Decoder: Different from the encoder, our decoder only consists of the transformer network. In the decoder, the transformer is implemented with two self-attention layers described above and a softmax layer to generate the final output. We input the fused representation $\tilde{e}_{fused}$ to decoder, which is the output of the fusion function (see IV-D). With the calculation in self-attention layers and softmax layer, the decoder generates a sequence of probabilities $\hat{\boldsymbol{y}}$ on vocabulary to estimate the description of target POI. Then the probabilities $\hat{\boldsymbol{y}}$ can be converted to a word sequence (description) by beam search. For simplicity, we use $\hat{\boldsymbol{y}}$ to denote both the generated description and the predicted probabilities of the decoder if without confusing in the context.

Our optimization is driven by maximum likelihood estimation based on the ground-truth $\boldsymbol{y}$ and the output of decoder $\hat{\boldsymbol{y}}$.

We adopt KL divergence as our loss function, which can be formulated concisely as below.

$$l = KL(\hat{\boldsymbol{y}}|\boldsymbol{y}) \tag{6}$$

To improve the accuracy and BLEU score, the label smoothing with value $\epsilon_{ls} = 0.1$ [25] is also applied in our optimization.

B. Category Embedding

The category embedding is also one of the components of our multi-mode encoder. In an online Map Service, each POI is always assigned a fine-grained category such as "hotel", "restaurant" and "shop". Such categories not only indicate the business scope of a POI, but also determine the main service of the POI. Thus, a POI's category has an important relation with the description of the POI. For example, a hotel description tends to emphasize environment and transportation while a restaurant description tends to emphasize taste and price.

Inspired by this phenomenon, we adopt a category embedding layer to extract the information behind categories. In category embedding, the one-hot category vector $\boldsymbol{t} \in R^{d_{model}}$ is embedded by a linear function.

$$\boldsymbol{t}_{emb} = W_t\boldsymbol{t} + b_t, \tag{7}$$

where W_t is a weight matrix and the b_t is a vector.

C. CACNN

A unique feature of POI description generation is that spatial context information has a great impact on the final description of POIs, making it different from general product description generation. In order to utilize such spatial context information, we first define a concept of context tensor $\boldsymbol{C}$, and then equip the multi-mode encoder with CACNN to incorporate the context tensor $\boldsymbol{C}$.

1) Context Tensor Construction: Given a POI p, we extract a map of the square region centered on p and divide extracted map to $l \times l$ grids. Then, the context tensor $\boldsymbol{C} \in R^{l \times l \times m}$ of p is calculated by the count of POIs of each category on each grid. It means the element $\boldsymbol{C}_{i,j,k}$ is the number of POIs on $(i,j)^{th}$ grid under the k^{th} category.

Figure 2(a) illustrates how to construct context tensor $\boldsymbol{C}$. First, taking the location of target POI as the center, we extract a square area with side length h (e.g. 3,000m) on a map, and divide the square area into $l \times l$ grids. Each grid represents a small square area with side length h/l on the map. Then, we calculate m heat maps on each grid where m is the number of categories of POIs. The value $c_{i,j,k}$ of grid $g_{(i,j)}$ in k^{th} heat map equals the number of POIs under the k^{th} category in the area represented by $g_{(i,j)}$. The calculation of value $c_{i,j,k}$ under grid $g_{(i,j)}$ of k^{th} heat maps can be formally defined as below:

$$c_{i,j,k} = |\{p_t|Loc(p_j) \in g_{(i,j)} \; and \; Category(p_t) = k\}|, \tag{8}$$

where $Category(p_t) = k$ means the poi p_t has the k^{th} category, and $Loc(p_t)$ returns the location of p_j ($1 \leq i,j \leq l, 1 \leq k \leq m$). Finally, we stack all the m heat maps to a tensor $\boldsymbol{C} \in R^{l \times l \times m}$.

2) Framework of CACNN: Considering the superiority of CNN [26], [27], we propose to use CNN to extract the information from the context tensor $\boldsymbol{C}$, named it as CACNN. The structure of CACNN is highly dependent on the division of the nearby map. Figure 2(b) provides an implementation of CACNN that is adopted in our implementation. The output tensor from the CACNN is reshaped to a vector that shares the same dimension d_{model} with other tensors. With f_c to denote the CACNN and reshape function, the output $\boldsymbol{c}_{spa}$ can be formulated as below.

$$\boldsymbol{c}_{spa} = f_c(\boldsymbol{C}) \tag{9}$$

To demonstrate the effectiveness of CACNN, we also provide a comparison between CACNN and other methods for processing spatial information in section V.

D. Fusion Function

In the end of multi-mode encoder, we fuse all of the representations including $\boldsymbol{e}_{rev}$,$\boldsymbol{t}_{emb}$ and $\boldsymbol{c}_{spa}$ to $\boldsymbol{e}_{fused}$. We implement the fusion by a linear function over the concatenation $[\boldsymbol{e}_{rev}; \boldsymbol{t}_{emb}; \boldsymbol{c}_{spa}]$.

$$\boldsymbol{e}_{fused} = W_f[\boldsymbol{e}_{rev}; \boldsymbol{t}_{emb}; \boldsymbol{c}_{spa}] + b_f \tag{10}$$

where the parameter W_f is a matrix and the parameter b_f is a vector. The dimension of $\boldsymbol{e}_{fused}$ is also set to d_{model}. Essentially, the output of the multi-mode encoder contains multiple types of information (reviews, category and spatial context) for description generation. All the parameters in MMDG are optimized in an end-to-end manner during the training stage.

V. Experiment

To compare the MMDG model with existing models, we construct a large-scale dataset from Baidu Maps and conduct a series of extensive experiments. In this section, we first introduce our dataset and experimental settings, followed by the discussion on baseline models and evaluation metrics. We also conduct analyses of the experimental results and demonstrate the effectiveness of our model.

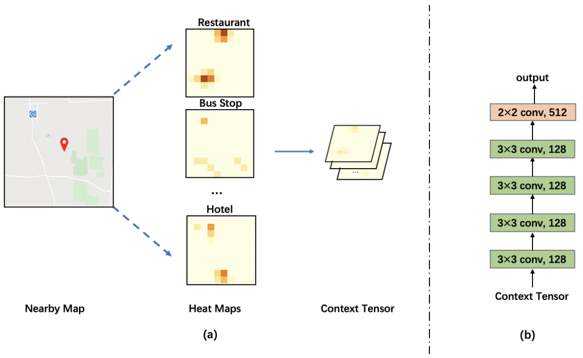

Fig. 2. (a) Context Tensor Construction. (b) Implementation of CACNN.

A. Dataset

We first constructed a new dataset based on Baidu Maps. However, it is a great challenge that the original POI does not have a description in map data. Reference to the research conducted by Novgorodov et al. [28], some high-quality reviews can be used directly as ground-truth for description generation. We combined the machine learning method proposed in [28] with some simple filtering rules to improve precision, and extract 691,224 descriptions from reviews data. A detailed introduction about the dataset construction can be found in the appendix.[2] A product manager manually assessed 1000 random samples, and concluded that 95.2% of the descriptions are high-quality. Besides the ground-truth description, each sample contains the category, context tensor and the reviews of the target POI. The basic information is reported in Table I. Note that such ground-truth construction method can only generate description for a small portion of whole POIs. The aim of MMDG is to generate POI description as much as possible.

B. Environment and Settings

1) Environment: All experiments in this paper are conducted on a GPU-CPU platform. The GPU is Nvidia Tesla P40. The program and baselines are implemented in Python 2.7.

2) Settings: Following [12], we empirically set the words embedding size d_{model} to be 256, and the size of FFN inner Representation d_{ff} to be 1024. The sizes of other representations (including categories and context) are also set to be 256. We applied dropout layers [29] with $p_{drop} = 0.1$ both in our model and baseline models for regularization, and all of the activation functions are set to be ReLU function [30].

In training stage, We adopt the Adam optimizer [31] with the parameters $\beta_1 = 0.9$, $\beta_2 = 0.998$ and $\epsilon = 1 \times 10^{-9}$. The learning rate is initialized to 1×10^{-4} and gradient clipping is adopted with the threshold value 5. In testing stage, we adopt the beam-search trick with beam size 8. The batch size is set to 48 for both training and testing.

C. Baseline Models

To demonstrate the effectiveness of our feature selection and network design, we compare our model with rich baseline models which have achieved state-of-the-art results in description generation task. All the models evaluated in experiment are summarized below.

- **Seq2Seq**: Following Bahdanau et al. [14], we implement the attentional Seq2Seq model with LSTM [32] and Bahdanau attention as the first baseline model. This baseline model only takes the reviews of target POI as input without expert knowledge such as category and context information.
- **Trans** (transformer): We implement a vanilla transformer model which is the same as [12]. Similar to our Seq2Seq baseline model, the Trans only takes the reviews of target POI as input.
- **+C** (Trans + Category): Besides reviews, this modified version of the transformer also takes category information as input. In this baseline model, the one-hot category vector is incorporated by the same embedding layer as MMDG.
- **+CV** (Trans + Context Vector): We implement this modified version of the transformer to incorporate reviews and context information of the target POI. Different from the context tensor discussed in section IV-C, this baseline model extracts a context vector from nearby map without division to grids. Each element in the context vector represents the number of nearby POIs under a category. The context vector is incorporated in a fully connected neural network to learn the deep representation. We design this baseline model to demonstrate the effectiveness of our context tensor construction and CACNN.
- **+CT** (Trans + Context Tensor): This baseline model takes the context tensor (see SectionIV-C) and reviews of the target POI as input. In this baseline model, the context tensor is incorporated by the same CACNN component as MMDG.
- **MMDG**: This model is our proposed model. As discussed before, our model takes target POI reviews, category information and context tensor as input. Enhanced by the category embedding and CACNN, our model can incorporate multi-mode information efficiently.

TABLE I
BASIC INFORMATION ABOUT THE DATASET

Attribute	Value
Train dataset size	667,224
Valid dataset size	12,000
Test dataset size	12,000
Average length of Reviews	243.6
Average length of Description	115.5
Vocabulary size of Reviews	12007
Vocabulary size of Description	9973

D. Metrics

We adopt some standard metrics to evaluate model performance in the POI description generation.

- **BLEU**: BLEU [33] is widely used in NLG task such as translation and summarization. It compares the co-occurrences lexical n-grams between generated text and ground-truth text, and the n-grams BLEU score is calculated as the rate of n-grams in candidate text which can be found in reference text. To put into a nutshell, BLEU measures the precision of the generated text. In our experiments, we report the geometric mean of 1,2,3,4-grams BLEU score.
- **ROUGE**: ROUGE [34] is also popular in NLG task. It calculates the overlapping n-grams between the generated text and the target text. Different from BLEU, ROUGE

[2] http://zhoujingbo.github.io/paper/poigenicde2021apx.pdf

TABLE II
PERFORMANCE OF MMDG AND BASELINE MODELS

Model	BLEU	ROUGE-1	ROUGE-2	ROUGE-L
seq2seq	12.05	35.12	14.87	31.22
V-Trans	14.88	35.81	16.72	32.14
+C	15.49	36.86	17.23	32.83
+CV	15.18	37.00	16.64	32.47
+CT	15.46	36.86	17.07	32.68
MMDG	**15.84**	**36.72**	**17.53**	**32.93**

also considers the recall rate in the generated text. We report the F1 scores of ROUGE-1, ROUGE-2 and ROUGE-L in our experiments.

E. Result Analysis

As shown in Table II, the transform network (**Trans**) is more effective than LSTM model (**Seq2Seq**) in our POI description generation task. Besides, both the category information and context information improve generation performance. The gap between the baseline model **+CV** and **+CT** demonstrates the effectiveness of our proposed CACNN. Compared with the simple context vector, the context tensor can retain the spatial structure, and perform better with CACNN in POI description generation. Moreover, We also observe that our proposed model achieved the best results in our experiments. In means that our combination of category embedding and CACNN is successful.

VI. CONCLUSION

In this paper, we systematically study the description generation task for Points of Interest. We propose a novel neural model, called MMDG, to tackle the description generation for POIs. Consisted of a multi-mode encoder and a transformer-based decoder, our proposed MMDG model can incorporate multi-mode information efficiently and generate high-quality description. Furthermore, we build a large-scale real-life dataset for POI description generation and conducted a series of extensive experiments. The experimental result demonstrated the effectiveness of our proposed model in terms of various metrics including BLEU and ROUGE.

REFERENCES

[1] J. Zhou, S. Gou, R. Hu, D. Zhang, J. Xu, A. Jiang, Y. Li, and H. Xiong, "A collaborative learning framework to tag refinement for points of interest," in *KDD*, 2019, pp. 1752–1761.

[2] X. He, T. Chen, M.-Y. Kan, and X. Chen, "Trirank: Review-aware explainable recommendation by modeling aspects," in *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, 2015, pp. 1661–1670.

[3] Y. Zhang and C. Xu, "Explainable recommendation: A survey and new perspectives," *arXiv preprint arXiv:1804.11192*, 2018.

[4] H. Luo, J. Zhou, Z. Bao, S. Li, J. S. Culpepper, H. Ying, H. Liu, and H. Xiong, "Spatial object recommendation with hints: When spatial granularity matters," in *SIGIR*, 2020, pp. 781–790.

[5] L. A. Jiang, Z. Yang, and M. Jun, "Measuring consumer perceptions of online shopping convenience," *Journal of Service Management*, vol. 24, 2013.

[6] E.-J. Lee and S. Y. Shin, "When do consumers buy online product reviews? effects of review quality, product type, and reviewer's photo." *Computers in Human Behavior*, vol. 31, 2014.

[7] M. Limayem, M. Khalifa, and A. Frini, "What makes consumers buy from internet? a longitudinal study of online shopping," *IEEE Transactions on systems, man, and Cybernetics-Part A: Systems and Humans*, vol. 30, no. 4, pp. 421–432, 2000.

[8] I. Langkilde and K. Knight, "Generation that exploits corpus-based statistical knowledge," in *ACL*, vol. 1, 1998, pp. 704–710.

[9] J. Wang, Y. Hou, J. Liu, Y. Cao, and C.-Y. Lin, "A statistical framework for product description generation," in *IJCNLP*, vol. 2, 2017, pp. 187–192.

[10] Q. Chen, J. Lin, Y. Zhang, H. Yang, J. Zhou, and J. Tang, "Towards knowledge-based personalized product description generation in e-commerce," in *KDD*. ACM Press, 2019, pp. 3040–3050.

[11] T. Zhang, J. Zhang, C. Huo, and W. Ren, "Automatic generation of pattern-controlled product description in e-commerce," in *WWW*, May 2019, pp. 2355–2365.

[12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Łukasz Kaiser, and I. Polosukhin, "Attention is all you need," in *NIPS*, 2017, pp. 5998–6008.

[13] S. Ilya, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *NIPS*, 2014, pp. 3104–3112.

[14] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *ICLR*, Jan. 2015.

[15] A. M. Rush, S. Chopra, and J. Weston, "Model for abstractive sentence summarization," in *EMNLP*, 2015, p. 379–389.

[16] S. Chen, "A general model for neural text generation from structured data," *E2E NLG Challenge System Descriptions*, 2018.

[17] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, Y. C. Maxim Krikun, Q. Gao, and K. M. et al, "Google's neural machine translation system: Bridging the gap between human and machine translation," *arXiv preprint arXiv:1609.08144*, 2016.

[18] R. Nallapati, B. Zhou, C. N. dos Santos, Çaglar Gulçehre, and B. Xiang, "Abstractive text summarization using sequence-to-sequence rnns and beyond," in *CoNLL*, 2016, p. 280–290.

[19] Y. Xia, J. Zhou, Z. Shi, C. Lu, and H. Huang, "Generative adversarial regularized mutual information policy gradient framework for automatic diagnosis," in *AAAI*, vol. 34, no. 01, 2020, pp. 1062–1069.

[20] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[21] Y.-H. H. Tsai, S. Bai, P. P. Liang, J. Z. Kolter, L.-P. Morency, and R. Salakhutdinov, "Multimodal transformer for unaligned multimodal language sequences," in *ACL*, 2019.

[22] Y. Liu and M. Lapata, "Hierarchical transformers for multi-document summarization," in *ACL*, 2019, pp. 5070–5081.

[23] P. Zhang, N. Ge, B. Chen, and K. Fan, "Lattice transformer for speech translation," in *ACL*, 2019, pp. 6475–6484.

[24] H. Le, D. Sahoo, N. Chen, and S. Hoi, "Multimodal transformer networks for end-to-end video-grounded dialogue systems," in *ACL*, 2019, pp. 5612–5623.

[25] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[26] N. Deo and M. M. Trivedi, "Convolutional social pooling for vehicle trajectory prediction," in *CVPR*. IEEE, 2018, pp. 1468–1476.

[27] H. Yao, F. Wu, J. Ke, X. Tang, Y. Jia, S. Lu, P. Gong, J. Ye, and Z. Li, "Deep multi-view spatial-temporal network for taxi demand prediction," in *AAAI*, 2018.

[28] S. Novgorodov, I. Guy, G. Elad, and K. Radinsky, "Generating product descriptions from user reviews," in *WWW*, 2019, pp. 1354–1364.

[29] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[30] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *ICML*, 2010, pp. 807–814.

[31] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[32] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[33] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *ACL*, 2002, p. 311–318.

[34] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," 2004.