SILVER: Generating Persuasive Chinese Product Pitch

Yunsen Hong[1], Hui Li[1], Yanghua Xiao[2], Ryan McBride[3], and Chen Lin[1](✉)

[1] School of Informatics, Xiamen University, Xiamen, China
chenlin@xmu.edu.cn
[2] School of Computer Science, Fudan University, Shanghai, China
[3] School of Computing, Simon Fraser University, Burnaby, BC, Canada

Abstract. Building a silver-tongued salesbot is attractive and profitable. The first and pivotal step is to generate a product pitch, which is a short piece of persuasive text which both convey product information and deliver persuasive explanations related to customer demand. Recent advances in deep neural networks have empowered text generation systems to produce natural language descriptions of products. However, to produce persuasive product pitches, deep neural networks need to be fed with massive amounts of persuasive samples, which are not available due to huge labelling cost. This paper proposes SILVER, a persuasive Chinese product pitch generator, which addresses the issue of insufficient labeled data with data-level, knowledge-level and model-level solutions. At the data level, SILVER employs statistic analysis to automatically derive weak supervision rules that correlate with persuasive texts. At the model level, SILVER apply the weak supervision rules to re-rank outputs from an ensemble of models to enhance pitch generation performance. Finally, at the knowledge level, SILVER incorporates attribute hierarchy to embed product information in the pitch. Both automatic and human-involved evaluations on real data demonstrate that SILVER is able to produce more fluent, catchy and informative snippets than state-of-the-art text generation approaches.

Keywords: Persuasive product pitch · Text generation · Creative text generation

1 Introduction

No behavior is more human than selling. The statement comes from the fact that selling is not only unique to the human species but also a very common social behavior. According to [14], an incredible 40% of our daily time is spent on selling, not only objects, but also ideas and techniques. When we are taking efforts to make machines more human, the large expenditure of selling time we spend and the human nature of the selling behavior bring us an interesting yet challenging question: *Can a machine function like a salesperson?*

K. Karlapalem et al. (Eds.): PAKDD 2021, LNAI 12713, pp. 652–663, 2021.
https://doi.org/10.1007/978-3-030-75765-6_52

Table 1. An example of persuasive snippets for a bookcase under different consumption contexts.

Input	Product Attributes	书架，木质，自然 bookcase, wooden	
	Consumption Context	现代家居 modern house decor	奢华家居 luxury house decor
Output	Regular Pitch	木质书架，适合现代风格。 Wood bookshelf, suitable for modern style.	木质书架，适合奢华风格。 Wood bookshelf, suitable for luxurious style.
	Persuasive Pitch	一款现代简约木质书架，采用经典的原木材质，成就充满风格的都市生活。 A modern wooden bookcase that speaks to your style, simple and classic log materials designed for urban living.	自然原木制造的书架，设计典雅富贵，创造舒适奢华的家居环境，感受尊崇。 The bookshelf is made of natural logs. Its design has incorporated the elements of fortune, grace and fashion, making the of luxurious and comfortable living environment, which will surely make you feel esteemed.

Successful selling is complex. In selling technique, a sales *pitch* (i.e. a product snippet) is the most important step that initiates a sale. To help our industrial partner (i.e., an online E-commerce platform in China) to build a salesbot, we study the problem of generating persuasive Chinese product pitches. Planning the pitch requires powerful insights into the customer needs, great wisdom to connect client demands with product attributes, and conversational talents to convince the customer. Thus, we formalize the problem of persuasive pitch generator as follows: *Given a consumption context keyword that describes the customer needs, a set of product attributes, generate a persuasive snippet in natural language that relates the consumption context to the product.* Table 1 illustrates the difference between a regular pitch and a persuasive pitch. In general, the desirable pitch must be (1) informative: the product attributes are selected from the input to convey product information; (2) relevant: product attributes are expressed in a manner that achieves maximal relevance to the consumption context; (3) persuasive: the power of persuasion is enhanced by a catchy sentence that is enjoyable to read.

The problem falls in the broad class of language generation. Recently, end-to-end deep neural frameworks (DNN) [6,8,23,24], i.e., models that directly transform input of product attributes and consumption context to output of product pitch, have shown promising progress in this field. End-to-end frameworks have the advantage that errors do not accumulate across separate stages, i.e., in choosing the appropriate attributes and expressing the attributes. However, the success of neural frameworks is based on massive training data. In the problem of product pitch generation, it is difficult to obtain large amounts of training samples that pair the input of product attributes and consumption context with the output of informative, relevant and persuasive product pitch. The challenges include (1) human labeling is not only labor-costly but also subjective. Different people may have inconsistent opinions about which product snippets are persuasive. (2) the almost infinite space of product attributes lead to many out-of-vocabulary tokens which will affect the quality of the generated snippets. This problem is more severe for cold start products, i.e., products without enough training samples.

In this paper, we propose a persuasive product snippet generator **SILVER** (snippet loading via interest relevance) which functions as the pivotal component in a silver-tongued salesbot. SILVER addresses the challenges by data-level, model-level and knowledge-level solutions. As it is easy to obtain conventional product description data set (which is *not* persuasive) either publicly or from our industry partner, a practical and easy-to-implement alternative is to train DNNs on the conventional data set and post-process the outputs based on some persuasion rules. The contributions of our work are as follows:

- **At the data level** (Sect. 2), SILVER proposes a strategy to derive weak persuasion rules and avoid bias and subjectivity. The rules are automatically derived from comparative statistic analysis on two different data source regarding rhetoric, syntactic and vocabulary features.
- **At the model level** (Sect. 3), SILVER presents an ensemble-rerank framework to apply the automatically derived rules to enhance snippet generation performance.
- **At the knowledge level** (Sect. 3), SILVER incorporates knowledge of product attribute hierarchy to understand structural associations among product attributes and tackle the out-of-vocabulary product attributes.

Experiments on real data demonstrate the competency of SILVER. Our work not only brings economic benefits but also sheds insights on other efforts that make machines more human, e.g., creative text generation. Furthermore, the solutions we provide are practical to solve data scarcity problems that many other AI systems face.

The organization of this paper is as follows: Section 2 illustrates the method used for persuasion rule derivation in SILVER. Section 3 depicts the ensemble-rerank architecture used in SILVER. Section 4 presents the experimental study which shows the competency of our method. Section 5 briefly surveys related work and Sect. 6 concludes our work.

2 Persuasion Rule Derivation

Though our understanding of the art of persuasion in Chinese goes back at least as far as the *Ming school* (名学, 250 BCE) [7], we have not seen a fair and objective study of what linguistic factors contribute to successful persuasion in E-commerce. Toward this end, SILVER performs a comparative study on persuasive and non-persuasive product snippets and derives several labeling rules.

Persuasive Product Snippets: The PH Data Set. We crawl 48, 320 headlines from blogs in the *"shopaholic's choice"* section on the largest E-commerce platform in China. This section collects purchase recommendations from the leading bloggers on the platform. It is reported[1] that bloggers in this section are regularly accessed based on content quality, numbers of views/followers/ hot blogs, Click-Through-Rates, trending topics and numerous other metrics. Therefore, its persuasiveness is verified to be effective in marketing. Product snippet

[1] http://news.mydrivers.com/1/596/596411.htm.

can be extracted from the body and the headlines of these blogs. Compared to the body of blogs, headlines tend to be more catchy and convey the most important information. Furthermore, the body usually consists of multimedia elements such as pictures while our focus in this work is purely textual. Consequently, we extract the headlines to learn essential language patterns. This collection is called Persuasive Headline (PH) data set hereafter.

Non-Persuasive Product Snippets: The Review Data Set. We compare the PH data set with regular product descriptions (i.e., they are not persuasive) obtained from a public Chinese online review data set[2]. As these reviews are not intended for advertising, we consider them as non-persuasive. This collection is called the review data set.

2.1 Features

Table 2. Binary (B) and numerical (N) features. Significant features that pass Bonferroni correction and their derived rules are highlighted.

Feature	Type	Value $f(s)$	Rule $l(s)$
Rhetoric Features			
Simile 比喻	B	If s uses connecting words such as "仿佛/好像" (i.e., "like"), then $f(s) = 1$	$f(s) = 1 \rightarrow l(s) = 1$
Antithesis 对偶	B	If s contains two clauses with equal lengths, then $f(s) = 1$	-
Anadiplosis 顶真	B	If one clause's last word appears at the beginning of the succeeding clause, then $f(s) = 1$	$f(s) = 1 \rightarrow l(s) = 1$
Rhetorical repetition 排比	B	$f(s) = 1$ if s contains three clauses, which have at least one common word, and their positional indexes within three clauses are similar (± 1 offset).	$f(s) = 1 \rightarrow l(s) = 1$
Repetition 反复	B	If one clause in s is followed by an identical cluase, then $f(s) = 1$	-
Rhetoric question 设问	B	If s contains at least a question clause, then $f(s) = 1$	-
Answer question 反问	B	If a question clause contains words such as "难道 (i.e.isn't)",then $f(s) = 1$	-
Regression 回环	B	If s follows a pattern "ABA,ABCBA," etc, then $f(s) = 1$	-
Rhetorical exchanging 互文	B	If s contains two cluases with equal lengths and their POS tag sequences are identical, then $f(s) = 1$	-
Enumeration 枚举	B	If the words in s seperated by commas are with identical POS tags, then $f(s) = 1$	-
Syntatic Features			
Normalized tree depth	N	$f(s) = d(s)/\max_s d(s)$, where $d(s)$ is the depth of the syntax tree of s	-
#clauses	N	$f(s) = c(s)/\max_s c(s)$, where $c(s)$ is the number of clauses in s	$c(s) \in [3, 9] \rightarrow l(s) = 1$
#tokens	N	$f(s) = \|s\|/\max_s \|s\|$, where $\|s\|$ is the number of tokens in s	$\|s\| \in [9, 51] \rightarrow l(s) = 1$
Causative verb 使动词	B	If a comma in s is followed by a causative verb such as "让/使/为/给" (i.e., "make/let/have/get"), then $f(s) = 1$	$f(s) = 1 \rightarrow l(s) = 1$
Sentence Entropy	N	$f(s) = \sum_{w \in s} \frac{n_s(w)}{\|s\|} \log \frac{n_s(w)}{\|s\|}$, where $w \in s$ is a word in s, $n_s(w)$ is the term frequency of w in s	-
Maximal tfidf	N	$f(s) = \max_{w \in s} n_s(w)/df(w)$, where $w \in s$ is a word in s, $n_s(w)$ is frequency of w in s, $df(w)$ is the document frequency of w	-
Position tfidf	N	$f(s) = 1/pos(w)$ where w is the word in s with maximal tfidf value	-
Volcaburary Features			
Chinese modal partic 语气词	B	If s's POS tag consequence has at least one modal particle, then $f(s) = 1$	-
Passive verb 被动词	B	If s's POS tag consequence has at least one passive verb, then $f(s) = 1$	-
Chinese auxiliary wor 助词	B	If s's POS tag consequence has at least one auxiliary word, then $f(s) = 1$	-
Chinese auxiliary wor 助词	B	If s contains at least one conjunction word, then $f(s) = 1$	-
4-word phrases 成语	B	$f(s) = 1$, if s contains at least a four-word Chinese phrase	-
Numerals	B	If s contains at least one numeral, then $f(s) = 1$	-
Book title mark	B	If s contains at least one book title mark, then $f(s) = 1$	-
Time words	B	If s's POS tag consequence has at least one time word, then $f(s) = 1$	-
Misused marks	B	If two punctuation marks are adjacent in s, then $f(s) = 1$	$f(s) = 1 \rightarrow l(s) = 0$
Quote	B	If s contains at least one phrase quoted with """", then $f(s) = 1$	-
Without adj. adv.	B	If s does not contain any adjective or adverb, then $f(s) = 1$	$f(s) = 1 \rightarrow l(s) = 0$
Exclamation	B	If s contains at least one exclamation mark, then $f(s) = 1$	$f(s) = 1 \rightarrow l(s) = 1$
#Adverbs	N	$f(s) = v(s)/\max_s v(s)$, where $v(s)$ is the number of adverbs in s	$v(s) \in [1, 2] \rightarrow l(s) = 1$
#Adjective	N	$f(s) = j(s)/\max_s j(s)$, where $j(s) = \sum_{w \text{is an adjective} n_s(w)}$ is the number of adjectives	-

Inspired by [16], we firstly pair the PH headlines with regular reviews under similar products. We filter trivial cases, i.e., products that have less than 300 snip-

[2] https://github.com/SophonPlus/ChineseNlpCorpus/blob/master/datasets/yf_amazon/intro.ipynb.

pets in any data set. We then explore 33 linguistic features, including rhetoric, syntactic and vocabulary features.

Many think that effective persuasion involves rhetorical skills, hence, we conduct a comprehensive study on traditional Chinese rhetorical skills according to [7], such as *Simile* (比喻), *Antithesis* (对偶), *Repetition* (反复), *Rhetoric Question* (设问), *Answer a question with a question* (反问), *Regression* (回环), *Rhetorical Exchanging* (互文), etc. For syntactic features, we include normalized syntax tree depth, sentence entropy, distribution of the word TFIDF, and so on. For vocabulary features, we use binary and numerical measures of the appearance of different POS tags in a snippet. We use a Chinese processing tool[3] for POS tagging. The features and their definitions are provided in Table 2.

2.2 Labeling Rules

We first compute the feature value $f(s)$ for each snippet s under each product, as defined in Table 2. Then, for binary features, we perform a two-tailed test of population proportion under each product. For numerical features, we perform Welch's t-test under each product, as it is more reliable when the numbers of headlines and reviews are generally not equivalent. Finally, for each feature, we perform Bonferroni correction to adjust significance level among all products. In this manner, we identify ten features that are statistically different in headlines and reviews with p-value less than 0.05.

Based on the statistic analysis, we derive a set $L = \{l\}$ of ten labeling rules, each of which consists of a precedent conditioned on a feature that passes Bonferroni corrected significance test, and a labeling rule $l(s) \in \{0, 1\}$. The precedent condition of the labeling rule is related to the feature value. For binary features, the precedent condition is $f(s) = 1$. For numeral features, we make the feature value fall in the range of $[\mu(f) \pm \sigma(f)]$, where $\mu(f)$ and $\sigma(f)$ are the mean value and variance of the feature on PH. A labeling rules $l(s)$ assigns either a positive label 1 to a persuasive snippet s, or a negative label 0 to a non-persuasive snippet. The sign of the label is determined by the z-score of the hypothesis test, i.e., more $f(s) = 1$ for binary feature or higher $f(s)$ for numerical feature in PH results in a positive labeling rule.

3 SILVER: Ensemble-Rerank

The overall architecture of SILVER, which is shown in Fig. 1, follows an encoder-decoder framework [2].

Input. Each training sample $\langle x^i, y^i \rangle$, where i indicates the index of the sample, contains a set of input segments $x^i = \{x_1^i, \cdots, x_J^i\}$ and an output sequence $y^i = \langle y_1^i, \cdots, y_T^i \rangle$. Let $1 \le j \le J$ and $1 \le t \le T$. Each x_j^i of the the input segments represents a consumption context or a product attribute. All tokens in the input and output are from the vocabulary $\mathcal{V}$, i.e., $x_j^i, y_t^i \in \mathcal{V}$.

[3] https://github.com/fxsjy/jieba.

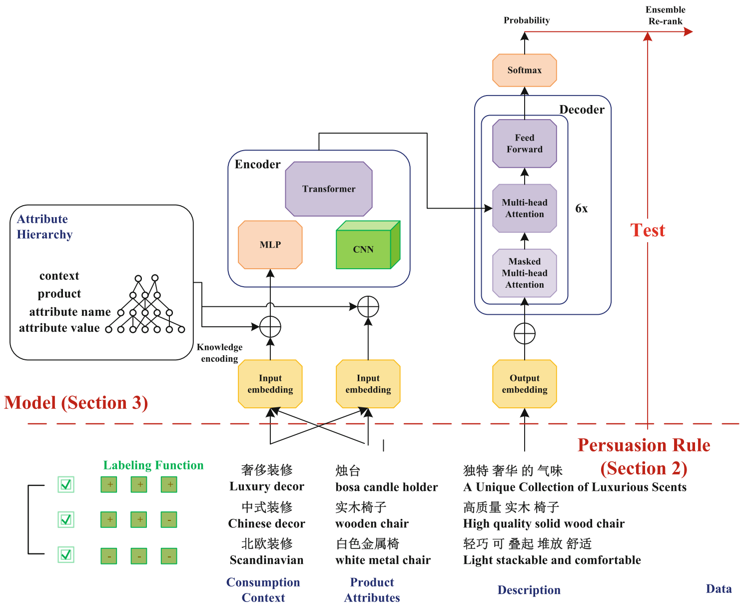

Fig. 1. Framework of SILVER

Knowledge. We construct a four-layer attribute hierarchy for each consumption, which we call "*knowledge*", using users' search logs on our industrial partner's E-commerce platform. The node in the first layer of each hierarchy is the consumption context, the nodes of the second layer are products, the nodes of the third layer are product attributes associated with each product in the second layer, and the leaf nodes in the forth layer indicate the attribute values of the corresponding product attribute in the third layer. Different consumption contexts have their own hierarchical structure, but some nodes are shared among different hierarchies as the same products, attribute names and attribute values may exist in multiple consumption contexts. Therefore, all the hierarchies can be globally viewed as a graph.

To leverage the knowledge, we use DeepWalk [13], a graph embedding method, to learn the representations (i.e., embeddings) of each word for context (e.g., "北欧装修"/"Scandinavian house decor"), product (e.g., "书柜"/"bookcase"), attribute, (e.g., "风格"/"style") or attribute value (e.g., "现代"/"modern") from the structural associations contained in the graph. Then, SILVER used the learned word embedding as the *knowledge enhanced* input embedding. Note that many other graph embedding learning approaches [3] can be employed for this step. We choose DeepWalk, since its robust performance makes it a standard method used in many tasks [3].

Ensemble and Rerank. The overall architecture of SILVER follows an encoder-decoder framework [23,24] and the idea of *ensemble-rerank*. We create

Table 3. Statistics of data

Context	Modern	Luxury	Scandinavian
#snippets	15,294	6,306	12,725
#product categories	66	24	6
#attributes	168	158	168
#unique tokens (segmented words)	17,014	10,154	15,237

several neural networks with different encoder blocks including Multi-Layer Perception (MLP), CNN and Transformer [19], and identical Transformer [19] decoder blocks. We train each model individually with early stopping, i.e., a network stops when the loss does not decrease on a validation set. We accumulate the top-k predicted candidates from each model and assign each candidate a persuasive score $g(i) = \frac{\sum_{l \in L} l(y^i)}{|L|} + \frac{\sum_j I(x^i_j = y^i_j)}{|x^i|}$ where $l(y^i)$ is the output of each labeling rule, $|L|$ is the number of rules, $\sum_j I(x^i_j = y^i_j)$ is the number of attributes in input x^i which appear in the output snippet y^i, $|x^i|$ is the number of input attributes. We then rerank all candidates together by $g(i)$ and return the global top-k candidates as the output.

4 Experiments

In this section, we provide an experimental study to demonstrate the effectiveness of SILVER.

4.1 Experimental Setup

The data we use contains a set of product descriptions collected from online house decoration stores on the E-commerce platform of our industrial partner. Before weak-supervised labeling, it contains approximately 0.2 million descriptions for 91 different products. After labeling, 82% positive descriptions remain and each description is associated with 1 product and 3.52 product attributes on average. The average length of description before and after labeling is 103 tokens and 76 tokens, respectively. After labeling, 94.65% of the remaining descriptions do not contain any consumption context keywords. Statistics of the data are shown in Table 3.

We focus on four consumption contexts: modern house decor, luxury house decor, Chinese house decor, and Scandinavian house decor. For each consumption context, we randomly select approximate 80% descriptions (including instances without any context keyword) as the training set, the remaining 20% is used for testing.

4.2 Competitors

We compare SILVER with several state-of-the-art methods:

Table 4. Objective evaluations of all methods with best results shown in bold

Context	Model	BLEU-1	BLEU-2	BLEU-3	ROUGE-1	ROUGE-2	ROUGE-3	ROUGE-L	ROUGE-S	CRF
Modern	NPLM	0.0528	0.0232	0.0087	0.1188	0.0209	0.0020	0.1077	0.0338	39.9970
	SC-LSTM	0.2344	0.1370	0.0676	0.2232	0.0395	0.0064	0.1770	0.0609	76.1197
	MLP	0.2299	0.1737	0.1063	0.3232	0.0618	0.0106	0.2383	0.0839	75.3715
	MLP+K	0.1926	0.1480	0.0978	0.2867	0.0709	0.0144	0.2262	0.0839	77.1888
	ResCNN	0.2055	0.1524	0.0967	0.2605	0.0584	0.0090	0.1940	0.0565	73.0549
	Transformer	0.1024	0.0759	0.0380	0.1741	0.0255	0.0036	0.1315	0.0314	75.4345
	Transformer+K	0.1714	0.1215	0.0566	0.2606	0.0287	0.0033	0.1886	0.0459	75.5515
	SILVER-1	**0.2550**	**0.1917**	**0.1245**	**0.3482**	**0.0798**	**0.0155**	**0.2565**	**0.0950**	**77.2031**
	SILVER-2	0.2412	0.1813	0.1171	0.3326	0.0758	0.0145	0.2477	0.0899	76.9532
Luxury	NPLM	0.0622	0.0283	0.0105	0.1068	0.0182	0.0006	0.0963	0.0264	31.3454
	SC-LSTM	0.2208	0.1245	0.0584	0.2117	0.0344	0.0055	0.1686	0.0561	76.0965
	MLP	0.2239	0.1688	0.1020	0.3266	0.0640	0.0133	0.2455	0.0827	73.7697
	MLP+K	0.1853	0.1407	0.0882	0.2768	0.0594	0.0122	0.2214	0.0758	**76.2510**
	ResCNN	0.1967	0.1435	0.0833	0.2622	0.0499	0.0069	0.2050	0.0543	72.4910
	Transformer	0.0977	0.0725	0.0358	0.1642	0.0217	0.0036	0.1283	0.0292	74.9017
	Transformer+K	0.1633	0.1148	0.0487	0.2529	0.0252	0.0028	0.1847	0.0421	74.9978
	SILVER-1	**0.2515**	**0.1885**	**0.1174**	**0.3521**	**0.0767**	**0.0164**	**0.2641**	**0.0939**	75.6342
	SILVER-2	0.2382	0.1783	0.1099	0.3326	0.0723	0.0154	0.2523	0.0876	75.5868
Scandinavian	NPLM	0.0608	0.0284	0.0110	0.1004	0.0162	0.0010	0.0901	0.0244	29.7617
	SC-LSTM	0.2208	0.1257	0.0646	0.2018	0.0358	0.0055	0.1598	0.0557	75.6598
	MLP	0.2000	0.1510	0.0927	0.2952	0.0572	0.0106	0.2191	0.0691	73.9286
	MLP+K	0.1633	0.1239	0.0786	0.2591	0.0551	0.0113	0.2028	0.0635	**76.3754**
	ResCNN	0.2090	0.1497	0.0916	0.2663	0.0566	0.0085	0.2013	0.0611	72.2239
	Transformer	0.0712	0.0528	0.0286	0.1488	0.0201	0.0026	0.1145	0.0233	75.3881
	Transformer+K	0.1497	0.1066	0.0531	0.2428	0.0243	0.0027	0.1750	0.0395	74.8781
	SILVER-1	**0.2454**	**0.1818**	**0.1161**	**0.3291**	**0.0729**	**0.0140**	**0.2428**	**0.0841**	75.6040
	SILVER-2	0.2283	0.1683	0.1059	0.3127	0.0656	0.0117	0.2336	0.0772	75.5444

1. NPLM [10]: an unsupervised framework that expands a set of keywords to creative product descriptions.
2. SC-LSTM [22]: a supervised framework which is based on a semantically controlled LSTM structure. SC-LSTM has the advantage of scaling sentence generation to cover multiple domains (e.g.,, the consumption contexts in our problem).
3. Transformer [19]: a supervised text generation framework which is purely based on attention mechanism.
4. ResCNN [5]: text generation framework consists of CNN encoder with residual learning and transformer decoder.
5. MLP: text generation framework with MLP encoder and transformer decoder.

For transformer and MLP, we also test their performance with knowledge (denoted as K) incorporated. If knowledge is not leveraged, the input embedding will be randomly initialized.

We set the dimensions of the embedding and hidden units to be 128. The size of mini-batch is set to be 32. We use 1000 descriptions as validation for early stopping. Codes and data are available at https://shorturl.at/suvxI.

4.3 Objective Evaluation

We evaluate the snippets generated by the best result output by competitors and the top-2 results output by SILVER. We use two metrics for objective evaluation, i.e., BLEU [12] and ROUGE [9]. BLEU is a standard metric for machine translation task. ROUGE is a commonly adopted metric for multi-document summarization task. For each training instance (i.e., a pair of input and output $\langle x^i, y^i \rangle$), BLEU and ROUGE calculate a score based on how close the system output is to the ground truth. We exclude BLEU-4, because BLEU-4 is based on 4-gram match and is only meaningful in the corpus level. We also adopt a Chinese Readability Formula (CRF) [20] as a compensatory evaluation metric.

We report the evaluations on the first candidate and the second candidate from the top-2 results of SILVER as SILVER-1 and SILVER-2, respectively. From the results in Table 4, we can observe that:

1. Supervised methods are better than the unsupervised method.
2. SILVER constantly produces the best performances in terms of all BLEU and ROUGE measures. Furthermore, the second result output by SILVER achieves the second best performances. This shows that, the labeling rules and the rerank scoring functions which are based on them, are effective.
3. SILVER produces comparable results in CRF. In fact, the CRF values are close to each other, indicating that the results output by different methods are of the same difficulty level.

4.4 Evaluation by Human

Previous study has acknowledged that automatic metrics do not consistently agree with human perceptions [22], especially when they are not designed for assessing the persuasiveness. To gain better insights into how and why SILVER produces more persuasive snippets, we conduct an evaluation involving human participants to assess the performance of SILVER.

Evaluation Protocol. Five judges are recruited to evaluate the quality of 30 randomly selected snippets generated by different methods on five metrics. For a fair evaluation, the method name (i.e., the instance is generated by which approach) is invisible to judges. Furthermore, the judges do not directly give an overall score of the corresponding method. Instead, they are asked to score on three aspects of each method. The range of the score is between 0 and 5. The three criteria are as follows:

- **Fluency** [21] measures whether the snippet is smooth. The judge is asked to focus on repeated terms and grammar mistakes. A score of 5 stands for zero mistakes, while 0 will be given if there are more than five mistakes.
- **Catchyness** [10] measures whether the snippet is attractive. The judge is asked to find attractive words. The score is given based on the ratio of attractive words, i.e., the number of attractive words divided by the number of total tokens.
- **Informative** measures whether the snippet is informative. The judge is asked to look at product attributes which are distinguishing. The score is given based on the number of distinguishing attributes. A score of 5 will be given if more than five distinguishing attributes exist. If the product is not mentioned, the snippet is assigned with a score of zero.

Table 5. Evaluation results from human judges. "Trans" indicates Transformer. Best results are shown in bold.

Metrics	MLP	MLP+K	ResCNN	Trans	Trans+K	SILVER
Fluency	1.96	2.79	1.37	3.02	3.03	**3.05**
Catchyness	2.20	2.71	1.55	2.66	2.72	**2.76**
Informative	4.07	4.11	3.89	2.18	3.11	**4.33**

Results. We report the evaluation results, which are average scores from all judges, for the basic SILVER and three other supervised methods in Table 5. From Table 5, we can conclude that SILVER produces more fluent, catchy, and informative snippets than state-of-the-art text generation approaches.

5 Related Work

We briefly survey two lines of research related to our work, i.e., language generation and learning with weak supervision.

Natural Language Generation (NLG) task is one of the most widely studied problems in the area of natural language processing. We identify two types of NLG tasks: data-to-document generation and creative text generation.

Data-to-document generation (DDG) is a classic NLG task. Given some structured data (e.g., a table), DDG produces text, such as a sentence or a paragraph, that adequately and fluently describes the input data. Early DDG systems typically consist of two separate stages: a content selection stage to decide "what to say", and a surface realization stage to decide "how to say". The recent success of Deep Neural Network (DNN) models [15] has motivated research on end-to-end systems that blur the distinction between the two stages. Most of the DNN-based systems employ an encoder-decoder framework. Frequently adopted encoders include Multi-Level Perception (MLP) [1,23] or a hierarchical form of LSTM [24]. In the decoder layers, RNN [23] and LSTM [24] are common choices.

Creative text generation (CTG) has received considerably more attention, from a commercial point of view. In CTG, the generated text must reveal more human characteristics. DNN-based methods are also appealing in CTG when supervision is accessible. For example, most state-of-the-art research on poetry generation is based on the encoder-decoder framework [4,21,25]. However, training collections are difficult to obtain for other types of creative text, due to the inherent complexity of the cognitive process. In this case, unsupervised methods are the mainstream solution. Most of them are heavily dependent on syntactic templates, e.g., word substitution [11,17,18], and can only generate short headline style sentences or slogans. A recent work [10] explores the possibility of generating a complete persuasive sentence by an unsupervised approach.

Our work is different from existing NLG work on the following two aspects: (1) While most NLG tasks focus on the output's fluency and fidelity to references, we emphasize on the persuasiveness and relevance of the output. (2) End-to-end DNN-based models require a tremendous amount of training data in order to obtain promising results. When it is impossible to generate labeled corpus, CTG systems often resort to unsupervised approaches. On the contrary, our work attempts to exploit the superior learning power of DNNs by utilizing weak supervisions.

Recently, a surge of works has been proposed that aims to address the data scarcity issue using weak supervision [26], which is the opposite of strong supervision [26]. The collection of weak supervision can be obtained by either an unsupervised model (with possibly worse performance) or a set of manually constructed heuristics. As weak supervisions are often incomplete (i.e., only a small fraction of training set is labeled), inaccurate (i.e., only coarse-grained labels are given) and/or inexact (i.e., given labels are not always correct), an adaption of the model is necessary for optimizing performance. However, this is not fully explored in the literature of DNN, especially for NLG. Most previous works simply treat weak supervision signals as normal labels.

6 Conclusion

In this paper, we propose a persuasive product snippet generator SILVER. SILVER leverages data-level, model-level and knowledge-level solutions to overcome the data scarcity problem and generate persuasive product snippets. The evaluations on real data from our industrial partner demonstrate that SILVER is able to produce persuasive snippets like a persuasive salesman. In the future, we plan to employ more sophisticated graph embedding approaches to improve SILVER.

Acknowledgment. Chen Lin is supported by the Natural Science Foundation of China (No. 61972328), Joint Innovation Research Program of Fujian Province China (No. 2020R0130). Hui Li is supported by the Natural Science Foundation of China (No. 62002303), Natural Science Foundation of Fujian Province China (No. 2020J05001). Yanghua Xiao is supported by NSFC (No. 61732004, No. 61472085, No. U1509213, No. U1636207), National Key R&D Program of China (No. 2017YFC0803700, No. 2017YFC1201200), Shanghai Municipal Science and Technology project (No. 16JC1420401), Shanghai STCSMs R&D Program (No. 16JC1420400).

References

1. Bao, J., Tang, D., Duan, N., Yan, Z., Zhou, M., Zhao, T.: Text generation from tables. IEEE/ACM Trans. Audio Speech Lang. Process. **27**(2), 311–320 (2019)
2. Cho, K., et al.: Learning phrase representations using RNN encoder-decoder for statistical machine translation. In: EMNLP, pp. 1724–1734 (2014)
3. Cui, P., Wang, X., Pei, J., Zhu, W.: A survey on network embedding. IEEE Trans. Knowl. Data Eng. **31**(5), 833–852 (2019)
4. Ghazvininejad, M., Shi, X., Choi, Y., Knight, K.: Generating topical poetry. In: EMNLP, pp. 1183–1191 (2016)

5. Huang, Y.Y., Wang, W.Y.: Deep residual learning for weakly-supervised relation extraction. In: EMNLP, pp. 1803–1807 (2017)
6. Karpathy, A., Li, F.: Deep visual-semantic alignments for generating image descriptions. In: CVPR, pp. 3128–3137 (2015)
7. Kirapatrick, A., Xu, Z.: Chinese Rhetoric and Writing: An Introduction for Language Teachers. Parlor Press, South Carolina (2012)
8. Lebret, R., Grangier, D., Auli, M.: Neural text generation from structured data with application to the biography domain. In: EMNLP, pp. 1203–1213 (2016)
9. Lin, C.Y.: Rouge: a package for automatic evaluation of summaries. In: Text Summarization Branches Out (2004)
10. Munigala, V., Mishra, A., Tamilselvam, S.G., Khare, S., Dasgupta, R., Sankaran, A.: Persuaide! an adaptive persuasive text generation system for fashion domain. In: WWW, pp. 335–342 (2018)
11. Özbal, G., Pighin, D., Strapparava, C.: BRAINSUP: brainstorming support for creative sentence generation. In: ACL (1), pp. 1446–1455 (2013)
12. Papineni, K., Roukos, S., Ward, T., Zhu, W.: Bleu: a method for automatic evaluation of machine translation. In: ACL, pp. 311–318 (2002)
13. Perozzi, B., Al-Rfou, R., Skiena, S.: DeepWalk: online learning of social representations. In: KDD, pp. 701–710 (2014)
14. Pink, D.H.: To Sell is Human: The Surprising Truth About Persuading, Convincing, and Influencing Others. Canongate Books, New York (2013)
15. Sutskever, I., Martens, J., Hinton, G.E.: Generating text with recurrent neural networks. In: ICML, pp. 1017–1024 (2011)
16. Tan, C., Niculae, V., Danescu-Niculescu-Mizil, C., Lee, L.: Winning arguments: interaction dynamics and persuasion strategies in good-faith online discussions. In: WWW, pp. 613–624 (2016)
17. Thomaidou, S., Lourentzou, I., Katsivelis-Perakis, P., Vazirgiannis, M.: Automated snippet generation for online advertising. In: CIKM, pp. 1841–1844 (2013)
18. Valitutti, A., Toivonen, H., Doucet, A., Toivanen, J.M.: Let everything turn well in your wife: generation of adult humor using lexical constraints. In: ACL (2), pp. 243–248 (2013)
19. Vaswani, A., et al.: Attention is all you need. In: NIPS, pp. 6000–6010 (2017)
20. Wang, L.: Research on Chinese readability formula of texts for elementary and intermediate South Korean and Japanese learners. Lang. Teach. Linguist. Stud. **5**(2017), 15–25 (2017)
21. Wang, Z., et al.: Chinese poetry generation with planning based neural network. In: COLING, pp. 1051–1060 (2016)
22. Wen, T., Gasic, M., Mrksic, N., Su, P., Vandyke, D., Young, S.J.: Semantically conditioned LSTM-based natural language generation for spoken dialogue systems. In: EMNLP, pp. 1711–1721 (2015)
23. Wiseman, S., Shieber, S.M., Rush, A.M.: Challenges in data-to-document generation. In: EMNLP, pp. 2253–2263 (2017)
24. Yang, Z., Blunsom, P., Dyer, C., Ling, W.: Reference-aware language models. In: EMNLP, pp. 1850–1859 (2017)
25. Zhang, X., Lapata, M.: Chinese poetry generation with recurrent neural networks. In: EMNLP, pp. 670–680 (2014)
26. Zhou, Z.H.: A brief introduction to weakly supervised learning. Nat. Sci. Rev. **5**(1), 44–53 (2017)