Automatic Generation of Chinese Short Product Titles for Mobile Display

Yu Gong[1 *], **Xusheng Luo**[2 * †], **Kenny Q. Zhu**[3 ‡], **Shichen Liu**[4], **Wenwu Ou**[5]
[1 4 5] Search Algorithm Team, Alibaba Group
[2 3] Shanghai Jiao Tong University
{gongyu.gy, lxs140564, shichen.lsc}@alibaba-inc.com,
kzhu@cs.sjtu.edu.cn, santong.oww@taobao.com

Abstract

This paper studies the problem of automatically extracting a short title from a manually written longer description of E-commerce products for display on mobile devices. It is a new extractive summarization problem on short text inputs, for which we propose a feature-enriched network model, combining three different categories of features in parallel. Experimental results show that our framework significantly outperforms several baselines by a substantial gain of 4.5%. Moreover, we produce an extractive summarization dataset for E-commerce short texts and will release it to the research community.

1 Introduction

Mobile Internet is fast becoming the primary venue for E-commerce. People have got used to browsing through collections of products and making transactions on the relatively small mobile phone screens. All major E-commerce giants such as Amazon, eBay and Taobao offer mobile apps that are poised to supersede the conventional websites.

When a product is featured on an E-commerce website or mobile app, it is often associated with a textual title which describes the key characteristics of the product. These titles, written by merchants, often contain gory details, so as to maximize the chances of being retrieved by user search queries. Therefore, such titles are often verbose, over-informative, and hardly readable. While this is okay for display on a computer's web browser, it becomes a problem when such longish titles are displayed on mobile apps. Take Figure 1 as an example. The title for a red sweater on an E-commerce mobile app is "ONE MORE文墨2017夏装新款印花连帽上衣长袖短款喇叭袖百搭卫衣女". Due to the limited display space on mobile phones, original long titles (usually more than 20 characters) will be cut off, leaving only the first several characters "ONEMORE文墨2017夏装... (ONE MORE 2017 summer woman...)" on the screen, which is completely incomprehensible, unless the user clicks on the product and load the detailed product page.

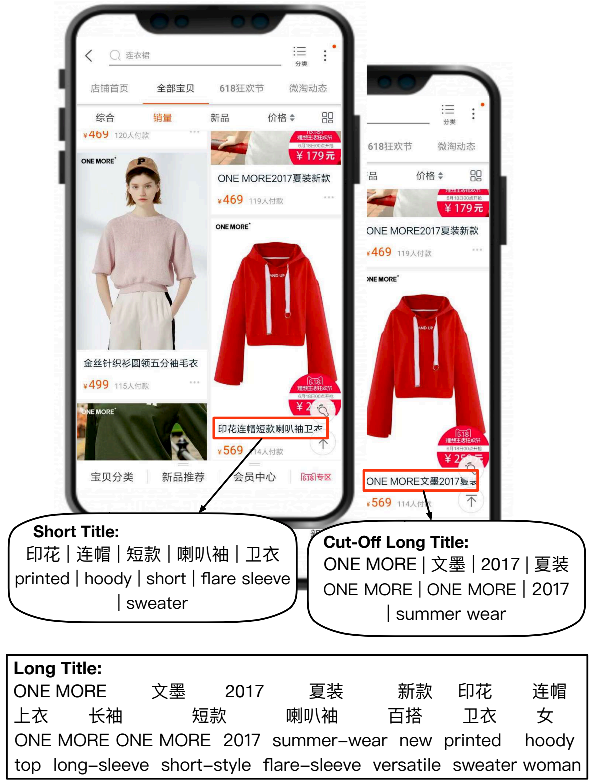

店铺首页
全部宝贝
618狂欢节
微淘动态
综合
销量
新品
价格
ONE MORE2017夏装新款
¥469
金丝针织衫圆领五分袖毛衣
¥499
印花连帽短款喇叭袖卫衣
¥569
ONE MORE文墨2017夏装
宝贝分类
新品推荐
会员中心
Short Title:
印花 | 连帽 | 短款 | 喇叭袖 | 卫衣
printed | hoody | short | flare sleeve | sweater
Cut-Off Long Title:
ONE MORE | 文墨 | 2017 | 夏装
ONE MORE | ONE MORE | 2017 | summer wear
Long Title:
ONE MORE 文墨 2017 夏装 新款 印花 连帽 上衣 长袖 短款 喇叭袖 百搭 卫衣 女
ONE MORE ONE MORE 2017 summer–wear new printed hoody top long–sleeve short–style flare–sleeve versatile sweater woman

Figure 1: A cut-off long title on an E-commerce mobile app, vs. a corresponding short title.

Thus, in order to properly display product listing on a mobile screen, one has to significantly simplify (e.g., to under 10 characters) the long titles while keeping the most important information. This way, user only has to glance through the search result page to make quick decision whether they want to click into a particular product. Figure 1 also shows as comparison an alternate display of a shortened title for the same product. The short title in the left snapshot is "印花|连帽|短款|喇叭袖|卫衣", which means "printed|hoody|short|flare sleeve|sweater".

In this paper, we attempt to extract short titles from their longer, more verbose counterparts for E-commerce products. To the best of our knowledge, this is the first attempt that attacks the E-commerce product short title extraction problem.

[*] Equally contribution.

[†] Work performed while interning at Alibaba Group.

[‡] Corresponding author.

This problem is related to text summarization, which generates a summary by either *extracting* or *abstracting* words or sentences from the input text. Existing summarization methods have primarily been applied to news or other long documents, which may contain irrelevant information. Thus, the goal of traditional summarization is to identify the most essential information in the input and condense it into something as fluent and readable as possible.

We would attack our problem with an *extractive* summarization approach, rather than an *abstractive* one for these reasons. First, our input title is relatively shorter and contains less noise, (average 27 characters; see Table 1). Some words in the long title may not be important but they are all relevant to the product. Thus, it is sufficient to decide if each word should or should not stay in the summary. Second, the number of words in the output is strictly constrained in our problem due to size of the display. Generative (abstractive) approaches do not perform as well when there's such a constraint. Finally, for E-commerce, it is better for the words in the summary to come from the original title. Using different words may lead to a change of original intention of the merchant.

State-of-the-art neural summarization models [Cheng and Lapata, 2016; Narayan *et al.*, 2017] are generally based on attentional RNN frameworks and have been applied on news or wiki-like articles. However, in E-commerce, customers are not so sensitive to the order of the words in a product title. Besides using deep RNN with attention mechanism to encode word sequence, we believe other single-word level semantic features such as NER tags and TF-IDF scores will be as just as useful and should be given more weights in the model. In this paper, we propose a feature-enriched neural network model, which is not only deep but also wide, aiming to effectively shorten original long titles.

The contributions of this paper are summarized below:

- We collect and will open source a product title summary dataset (Section 2).
- We present a novel feature-enriched network model, combining three different types of word level features (Section 4), and the results show the model outperforms several strong baseline methods, with $ROUGE\text{–}1_{F1}$ score of 0.725 (Section 5.4).
- By deploying the framework on an E-commerce mobile app, we witnessed improved online sales and better turnover conversion rate in the popular 11/11 shopping season (Section 5.5).

2 Data Collection

Publicly available large-scale summarization dataset is rare. Existing document summarization datasets include DUC2[1], TAC3[2] and TREC4[3] for English, and LCSTS[4] for Chinese. In this work, we create a dataset on short title extraction for E-commerce products. This dataset comes from a module in **Taobao** named “有好货”(Youhaohuo)[5]. *Youhaohuo* is a collection of high-quality products on Taobao. If you click a product in Youhaohuo, you will be redirected to the detailed product page (including product title). What is different from ordinary Taobao products is that online merchants are required to submit a short title for each Youhaohuo product. This short title, written by humans, is readable and describes the key properties of the product. Furthermore, most of these short titles are directly extracted from the original product titles. Thus, we believe Youhaohuo is a good data source of extractive summarization for product descriptions.

Figure 2 shows how we collected the data. On the left is a web page in Youhaohuo displaying several products, each of which contains an image and a short title below. When clicking on the bottom right dress, we jump to the detailed page on the right. The title next to the picture in red box is the manually written short title, which says “MIUCO针织马甲假两件收腰连衣裙” (MIUCO tight dress with knit vest). This short title is extracted from the long title below in the blue box. Notice that all the characters in the short tile are directly extracted from the long title (red boxes inside blue box). In addition to the characters in the short title, the long title also contains extra information such as “女装2017冬新” (woman's wear brand new in winter 2017). In this work, we segment the original long titles and short titles into Chinese words by jieba[6].

The dataset consists of 6,481,623 pairs of original and short product titles, which is the largest short text summarization dataset to date. We call it large extractive summary dataset for E-commerce (LESD4EC), whose statistics is shown in Table 1. We believe this dataset will contribute to the future research of short text summarization[7].

Table 1: Statistics of LESD4EC dataset.

No. of summaries	6,481,623
No. of words per text	12
No. of chars per text	27
No. of words per summary	5
No. of chars per summary	11

3 Problem Definition

In this section we formally define the problem of short title extraction. A char is a single Chinese or English character. A segmented word (or term) x is a sequence of several chars such as “Nike” or “牛仔裤”(jean). A product title, denoted as X, is a sequence of words $\{x_1, x_2, ..., x_n\}$. Let Y be a sequence of labels $\{y_1, y_2, ..., y_n\}$ over X, where $y_i \in \{0, 1\}$. The corresponding short title is a subsequence of X, denoted as $S = \{x_i\}$, where $y_i = 1$ and $|S| \leq n$.

We regard short title extraction task as a sequence classification problem. Each word is sequentially visited in the orig-

[1] http://duc.nist.gov/data.html

[2] http://www.nist.gov/tac/2015/KBP/

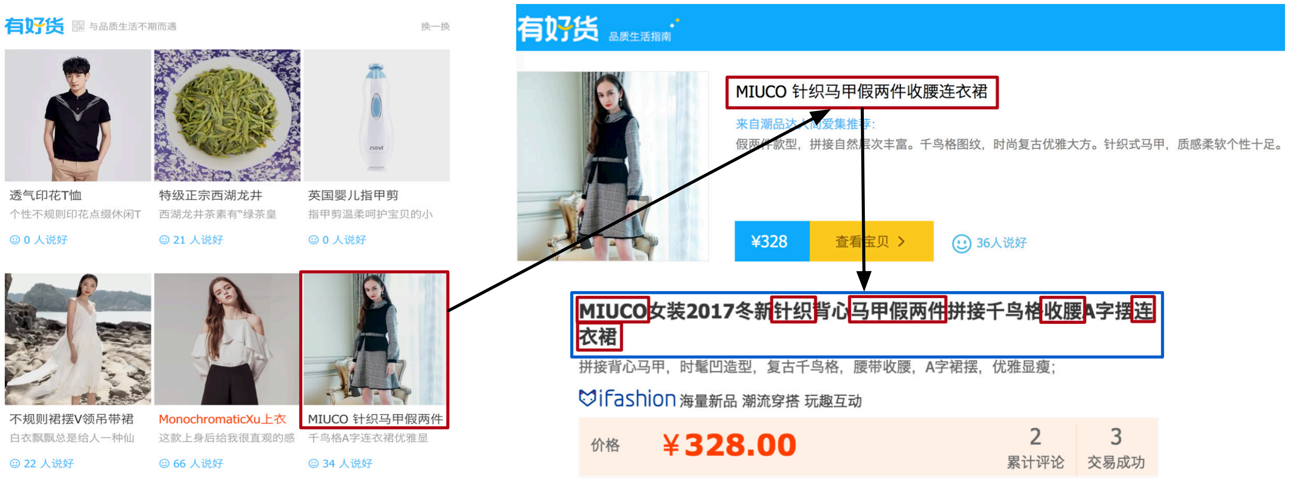
[3] http://trec.nist.gov/

[4] http://icrc.hitsz.edu.cn/Article/show/139.html

[5] https://h5.m.taobao.com/lanlan/index.html

[6] https://pypi.python/pypi/jieba/

[7] The dataset will be published after the paper is accepted. Notice all the original data can be crawled online.



有好货 与品质生活不期而遇 换一换

透气印花T恤
个性不规则印花点缀休闲T
☺ 0 人说好

特级正宗西湖龙井
西湖龙井茶素有"绿茶皇
☺ 21 人说好

英国婴儿指甲剪
指甲剪温柔呵护宝贝的小
☺ 0 人说好

不规则裙摆V领吊带裙
白衣飘飘总是给人一种仙
☺ 22 人说好

MonochromaticXu上衣
这款上身后给我很直观的感
☺ 66 人说好

MIUCO 针织马甲假两件
千鸟格A字连衣裙优雅显
☺ 34 人说好

有好货 品质生活指南

MIUCO 针织马甲假两件收腰连衣裙
来自潮品达人阁爱集推荐:
假两件款型，拼接自然层次丰富。千鸟格图纹，时尚复古优雅大方。针织式马甲，质感柔软个性十足。
¥328 查看宝贝 > ☺ 36人说好

MIUCO女装2017冬新针织背心马甲假两件拼接千鸟格收腰A字摆连衣裙
拼接背心马甲，时髦凹造型，复古千鸟格，腰带收腰，A字裙摆，优雅显瘦；
iFashion 海量新品 潮流穿搭 玩趣互动
价格 ¥328.00 2 累计评论 3 交易成功

Figure 2: Procedure of data collection in Youhaohuo.

inal product title order and a binary decision is made. We do this by scoring each word x_i within X and predicting a label $y_i \in \{0, 1\}$, indicating whether the word should or should not be included in the short title S. As we apply supervised training, the objective is to maximize the likelihood of all word labels $Y = \{y_1, y_2, ..., y_n\}$, given the input product title X and model parameters θ:

$$\log p(Y|X,\theta) = \sum_{i=1}^{n} \log p(y_i|X,\theta). \tag{1}$$

4 Feature-Enriched Neural Extractive Model

In this section, we describe our extractive model for product short title extraction. The overall architecture of our neural network based extractive model is shown in Figure 3. Basically, we use a Recurrent Neural Network (RNN) as the main building block of the sequential classifier. However, unlike traditional RNN-based sequence labeling models used in NER or POS tagging, where all the word level features are fed into RNN cell, we instead divide the features into three parts, namely *Content*, *Attention* and *Semantic* respectively. Finally we combine all three features in an ensemble.

4.1 Content Feature

To encode the product title, we first look up an embedding matrix $E_x \in \mathbf{R}^{d \times v}$ to get the word embeddings $\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n)$. Here, d denotes the dimension of the embeddings and v denotes the vocabulary size of natural language words. Then, the embeddings are fed into a bidirectional LSTM networks. To this end, we get two hidden state sequences, $(\overrightarrow{\boldsymbol{h}_1}, \overrightarrow{\boldsymbol{h}_2}, \ldots, \overrightarrow{\boldsymbol{h}_n})$ from the forward network and $(\overleftarrow{\boldsymbol{h}_1}, \overleftarrow{\boldsymbol{h}_2}, \ldots, \overleftarrow{\boldsymbol{h}_n})$ from the backward network. We concatenate the forward hidden state of each word with corresponding backward hidden state, resulting in a representation $[\overrightarrow{\boldsymbol{h}_i}; \overleftarrow{\boldsymbol{h}_i}]$. At this point, we obtain the representation of the product title X.

The *content* feature of current word x_i is then calculated as:

$$\boldsymbol{f}_{cont} = \boldsymbol{W}_{cont}^{T}[\overrightarrow{\boldsymbol{h}_i}; \overleftarrow{\boldsymbol{h}_i}] + \boldsymbol{b}_{cont}, \tag{2}$$

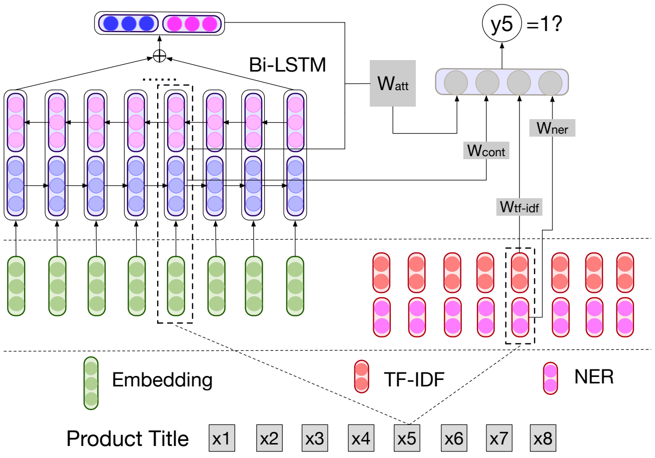


Figure 3: Architecture of Feature-Enriched Neural Extractive Model.

where $\boldsymbol{W}_{cont}$ and $\boldsymbol{b}_{cont}$ are model parameters.

4.2 Attention Feature

In order to measure the importance of each word relevant to the whole product title, we borrow the idea of attention mechanism [Bahdanau *et al.*, 2014; Luong *et al.*, 2015] to calculate a relevance score between the hidden vector of current word and representation of the entire title sequence.

The representation of the entire product title is modeled as a non-linear transformation of the average pooling of the concatenated hidden states of the BiLSTM:

$$\boldsymbol{d} = \tanh\left(\boldsymbol{W}_d \frac{1}{n} \sum_{j=1}^{n} [\overrightarrow{\boldsymbol{h}_i}; \overleftarrow{\boldsymbol{h}_i}] + \boldsymbol{b}_d\right). \tag{3}$$

Therefore, the attention feature of current word x_i is calculated by a Bilinear combination function as:

$$\boldsymbol{f}_{att} = \boldsymbol{d}^{T} \boldsymbol{W}_{att} [\overrightarrow{\boldsymbol{h}_i}; \overleftarrow{\boldsymbol{h}_i}] + \boldsymbol{b}_{att}, \tag{4}$$

where $\boldsymbol{W}_{att}$ is a parameter matrix.

4.3 Semantic Feature

Apart from the two hidden features calculated using an RNN encoder, we design another kind of feature including TF-IDF and NER Tag, to capture the deep semantics of each word in a product title.

TF-IDF

Tf-idf, short for term frequency–inverse document frequency, is a numerical statistic that is intended to reflect how important a word is to a document in corpus or a sentence in a document.

A simple choice to calculate term frequency of current word x_i is to use the number of its occurrences (or count) in the title:

$$\mathbf{tf}(x_i, X) = \frac{cnt_{x_i}}{n}. \tag{5}$$

In the case of inverse document frequency, we calculate it as:

$$\mathbf{idf}(x_i) = \log(1 + \frac{N}{C_{x_i}}), \tag{6}$$

where N is the number of product titles in the corpus and C_{x_i} is the number of titles containing the word x_i.

By combining the above two, the tf-idf score of word x_i in a product title X, denoted as $\mathbf{tf\text{-}idf}(x_i, X)$, is then calculated as the product of $\mathbf{tf}(x_i, X)$ and $\mathbf{idf}(x_i)$.

We design a feature vector containing three values: tf score, idf score and tf-idf score and the calculate a third feature:

$$\boldsymbol{v}_{tfidf} = [\mathbf{tf}(x_i, X); \mathbf{idf}(x_i); \mathbf{tf\text{-}idf}(x_i, X)] \tag{7}$$

$$\boldsymbol{f}_{tfidf} = \boldsymbol{W}_{tfidf}^T \boldsymbol{v}_{tfidf} + \boldsymbol{b}_{tfidf}. \tag{8}$$

NER

We use a specialized NER tool for E-commerce to label entities in a product title. In total there are 36 types of entities, which are of common interest in E-commerce scenario, such as "颜色"(color), "风格"(style) and "尺寸规格 "(size). For example, in segmented product title "包邮 Nike 品牌 的 红色 运动裤" (Nike red sweatpants with free shipping), "包邮 "(Free shipping) is labeled as *Marketing_Service*, "Nike" is labeled as *Brand*, "红色"(Red) is labeled as *Color* and "运动裤"(Sweatpants) is labeled as *Category*. We use one-hot representation to encode NER feature $\boldsymbol{v}_{ner}$ of each word and then integrate it into the model by proposing a fourth feature:

$$\boldsymbol{f}_{ner} = \boldsymbol{W}_{ner}^T \boldsymbol{v}_{ner} + \boldsymbol{b}_{ner}. \tag{9}$$

4.4 Ensemble

We combine all the features above into one final score of word x_i:

$$score_i = \sigma(\boldsymbol{W}_f^T [\boldsymbol{f}_{cont}; \boldsymbol{f}_{att}; \boldsymbol{f}_{tfidf}; \boldsymbol{f}_{ner}] + \boldsymbol{b}_f), \tag{10}$$

where σ is the sigmoid or logistic function, which restrain the score between 0 and 1. Based on that, we set a threshold τ to decide whether we keep word x_i in the short title or not.

Our model is very much like the Wide & Deep Model architecture [Cheng *et al.*, 2016]. While the *content* and *attention* features are deep since they rely on deep RNN structure, the *semantic* features are relatively wide and linear.

5 Experiments

In this section, we first introduce the experimental setup and the previous state-of-the-art systems as comparison to our own model known as Feature-Enriched-Net. We then show the implementation details and the evaluation results before giving some discussions on the results.

5.1 Training and Testing Data

We randomly select 500,000 product titles as our training data, and another 50,000 for testing. Each product title X is annotated with a sequence of binary label Y, i.e., each word x_i is labeled with 1 (included in short title) or 0 (not included in short title). Readers may refer to Section 2 for the details about how we collect the product titles and their corresponding short titles.

5.2 Baseline Systems

Since there are no previous work that directly solves the short title extraction for E-commerce product, we select our baselines from three categories. The first one is traditional methods. We choose a keyword extraction framework known as TextRank [Mihalcea and Tarau, 2004]. It first infers an importance score for each word within the long title by an algorithm similar to PageRank, then decides whether each word should or should not be kept in the short title according to the scores.

The second category is standard sequence labeling systems. We choose the system mentioned by Huang et al.[2015], in which a multi-layer BiLSTM is used. Compared to our system, it does not exploit the attention mechanism and any side feature information. We substitute the Conditional Random Field (CRF) layer with Logistic Regression to make it compatible with our binary labeling problem. We call this system BiLSTM-Net.

The last category of methods is attention-based frameworks, which use encoder-decoder architecture with attention mechanism. We choose Pointer Network [Vinyals *et al.*, 2015] as a comparison and call it Pointer-Net. During decoding, it looks at the whole sentence, calculates the attentional distribution and then makes decisions based on the attentional probabilities.

5.3 Implementation Details

We pre-train word embeddings used in our model on the whole product titles data plus an extra corpus called "E-commerce Product Recommended Reason", which is written by online merchants and is also extracted from YouHaoHuo (Section 2). We use the Word2vec [Mikolov *et al.*, 2013a; Mikolov *et al.*, 2013b] CBOW model with context window size 5, negative sampling size 5, iteration steps 5 and hierarchical softmax 1. The size of pre-trained word embeddings is set to 200. For Out-Of-Vocabulary (OOV) words, embeddings are initialized as zero. All embeddings are updated during training.

For recurrent neural network component in our system, we used a two-layers LSTM network with unit size 512. All product titles are padded to a maximum sentence length of 15.

We perform a mini-batch cross-entropy loss training with a batch size of 128 sentences for 10 training epochs. We use Adam optimizer, and the learning rate is initialized with 0.01.

5.4 Offline Evaluation

To evaluate the quality of automatically extracted short titles, we used ROUGE [Lin and Hovy, 2003] to compare model generated short titles to manually-written short titles. In this paper, we only report ROUGE-1, mainly because linguistic fluency and word order are not of concern in this task. Unlike previous works in which ROUGE is a recall-oriented metric, we jointly consider precision, recall and F1 score, since recall only presents the ratio of the number of extracted words included in ground-truth short title over the total number of words in ground-truth short title. However, due to the limited display space on mobile phones, the number of words (or characters) of extracted short title itself should be constrained as well. Thus, precision is also measured in our experiments. And $ROUGE\text{–}1_{F1}$ is considered a comprehensive evaluation metric:

$$ROUGE\text{–}1_P = \frac{|S \cap S_{human}|}{|S|},$$
$$ROUGE\text{–}1_R = \frac{|S \cap S_{human}|}{|S_{human}|},$$
$$ROUGE\text{–}1_{F1} = \frac{2 \cdot ROUGE\text{–}1_P \cdot ROUGE\text{–}1_R}{ROUGE\text{–}1_P + ROUGE\text{–}1_R}.$$

Where S_{human} is the manually written short title, and $|S \cap S_{human}|$ is the number of overlapping words appearing in short titles generated by model and humans.

Final results on the test set are shown in Table 2. We can find that our method (Feature-Enriched-Net) outperforms the baselines on both precision and recall. Our method achieves the best 0.725 F1 score, which improves by a relative gain of 4.5%. Pointer-Net would achieve higher precision score for its attentive ability to the whole sentence and select the most important words. Our method considers long-short memories, attention mechanism and other abundant semantic features. That is to say, our model has the ability to extract the most essential words from original long titles and make the short titles more accurate and comprehensive.

We also tune the threshold τ used in BiLSTM-Net and our Feature-Enriched-Net. This threshold indicates how large the predicted likelihood should be so that a word will be included in the final short title. We reported the results in Figure 4, in which we set τ as 0.3, 0.4 and 0.5. From the figures, we can conclude that our model stably performs better than the other.

Table 2: Final results on the test set. We report ROUGE-1 Precision, Recall and corresponding F1. We use the tuned threshold $\tau = 0.4$ (see Figure 4) for BiLSTM-Net and Feature-Enriched-Net. Best ROUGE score in each column is highlighted in boldface.

Models	$R\text{–}1_P$	$R\text{–}1_R$	$R\text{–}1_{F1}$
TextRank	0.430	0.219	0.290
BiLSTM-Net	0.637	0.751	0.689
Pointer-Net	0.648	0.746	0.694
Feature-Enriched-Net	**0.675**	**0.783**	**0.725**

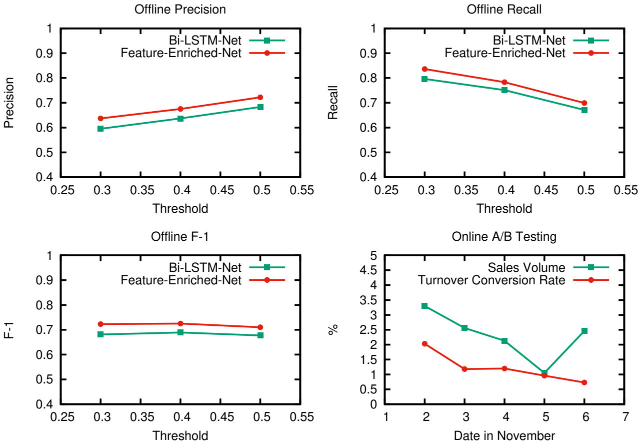


Figure 4: Offline results of Bi-LSTM-Net and Feature-Enriched-Net under different thresholds; Online A/B Testing of sales volume and Turnover Conversion Rate

5.5 Online A/B Testing

This subsection presents the results of online evaluation in the search result page scenario of an E-commerce mobile app with a standard A/B testing configuration. Due to the limited display space on mobile phones, only a fixed number of chars (12 characters in this app) can be shown out and excessive part will be cut off. Therefore, unlike previously mentioned inference approach (with threshold τ in Section 4.4), we regard it as classic Knapsack Problem. Each word x_i in the product title is an item with weight $len(x_i)$ and value $score_i$, where $len(x_i)$ represents the char length of word x_i and $score_i$ represents the predicted likelihood of word by our model. The maximum weight capacity of our knapsack (also known as char length limit) is m. Then the target is:

$$max \sum_{i=1}^{n} score_i z_i$$

$$s.t. \sum_{i=1}^{n} len(x_i) z_i \leq m, z_i \in \{0, 1\},$$

where $z_i = 1$ means x_i should be reserved in the short title. Similar to the standard solution to 0-1 Knapsack Problem, we use a Dynamic Programming (DP) algorithm.

In our online A/B testing evaluation, 3% of the users were randomly selected as testing group (about 3.4 million user views (UV)), in which we substituted the original cut-off long title displayed to users with extracted short titles by our model with DP inference. We claim that after showing the short titles with most important keywords, users have much better idea what the product is about on the “search result” page and thus find the product they want more easily. During the popular Double 11 shopping season, we deployed A/B testing for 5 days (from 2017-11-02 to 2017-11-06) and achieved on average **2.31%** and **1.22%** improvements of sales volume and turnover conversion rate (see Figure 4 for each day). This clearly shows that better short product titles are more user-friendly and hence improve the sales substantially.

Table 3: A real experimental case with 12-chars length limit. Pointer-Net is not included since encoder-decoder architecture can't directly adapt to character length limit

ORIGINAL	xunruo 熏 若 双生 设计师 品牌 泡泡 系列 一 字领 掉 袖 连衣裙 预订 款 (Bookable XunRuo twin designer brand bubble series dress with boat neckline and off sleeves.)
HUMAN	一字领 掉 袖 连衣裙 (Dress with boat neckline and off sleeves.)
Feature-Enriched-Net	泡泡 系列 一字领 掉 袖 连衣裙 (Bubble series dress with boat neckline and off sleeves.)
BiLSTM-Net	熏 若 品牌 掉 袖 连衣裙 预订 (Bookable XunRuo brand dress with off sleeves.)

5.6 Discussions

In Table 3, we show a real case of original long title, along with short title annotated by human beings, predicted by BiLSTM-Net and Feature-Enriched-Net respectively. From the human annotated short title, we find that a proper short title should contain the most important elements of the product, such as category ("dress") and description of properties ("boat neckline and off sleeves"). While some other elements such as brand terms ("XunRuo twin designer brand") or service terms ("open for reservation") should not be kept in the short title. Our Feature-Enriched Net has the ability to generate a satisfying short title, while baseline model tends to miss some essential information.

However, there is still room for improvement. Terms with similar meaning may co-occur in the short title generated by our model, when they all happen to be important terms such as a category. For example, "皮衣" and "皮夹克" both mean "jacket" in a long title, and the model tends to keep both of them. However, only one of them is enough and the space saved can be used to display other useful information to customers. We will explore intra-attention [Paulus *et al.*, 2017] as an extra feature in our future work.

6 Related Work

The task of summarization aims at generating short summaries for long documents or sentences. This research problem has drove lots of attentions in recent years and existing methods can be categorized mainly into two perspectives.

Extractive summarization methods [Erkan and Radev, 2004; McDonald, 2007; Wong *et al.*, 2008] produce summaries by concatenating several sentences or words found directly in the original texts. Several methods have been used to select the summary-worthy sentences, including binary classifiers [Kupiec *et al.*, 1995], Markov models [Conroy and O'leary, 2001], graphic model [Erkan and Radev, 2004; Mihalcea, 2005] and integer linear programming (ILP) [Woodsend and Lapata, 2010]. Compared to traditional methods which heavily rely on human-engineered features, neural network based approaches [Kågebäck *et al.*, 2014; Cheng and Lapata, 2016; Nallapati *et al.*, 2017; Narayan *et al.*, 2017] have gained popularity rapidly. The general idea of these methods is to regard the extractive summary as a sequence classification problem and adopt RNN like networks. Besides, attention-based framework can perform better by attending to the whole document when extracting a word (or sentence) [Cheng and Lapata, 2016]. Our work is based on extractive framework as well. Besides a deep attentional RNN network, we also employ explicit semantic features such as TF-IDF score and NER tags, making our model more informative.

On the other hand, *abstractive summarization* methods [Chen *et al.*, 2016; Nallapati *et al.*, 2016; See *et al.*, 2017], which have the ability to generate text beyond the original input text, in most cases, can produce more coherent and concise summaries. These approaches are mostly centered on the attention mechanism and augmented with recurrent decoders [Chopra *et al.*, 2016], Abstract Meaning Representations [Takase *et al.*,], hierarchical networks [Nallapati *et al.*, 2016] and pointer network [See *et al.*, 2017]. However, It is not necessary to use abstractive methods in short title extraction problem in this paper as the input is product title, which already contains the required informative words.

Our work can be comfortably set in the area of *short text summarization*. This line of research is in fact essentially *sentence compression* working on short text inputs such as tweets, microblogs or single sentence. Recent advances typically contribute to improving seq-to-seq learning, or attentional RNN encoder-decoder structures [Chopra *et al.*, 2016; Nallapati *et al.*, 2016]. While these methods are mostly abstractive, we use an extractive framework, combining with deep attentional RNN network and explicit semantic features, due to different scenarios of summarization problem. Besides, while existing summarization systems focus on news articles or other documents, we are the first to do summarization in a mobile E-commerce setting.

7 Conclusion

To the best of our knowledge, this is the first piece of work that focuses on extractive summarization for E-commerce product title. We propose a deep and wide model, combining attentional RNN framework with rich semantic features such as TF-IDF scores and NER tags. Our model outperforms several popular summarization models, and achieves ROUGE-1 F1 score of 0.725. Besides, the result of online A/B testing shows substantial benefits of our model in real online shopping scenario. Possible future works include handling similar terms that appear in the short titles generated by our model.

References

[Bahdanau *et al.*, 2014] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[Chen *et al.*, 2016] Qian Chen, Xiao-Dan Zhu, Zhen-Hua Ling, Si Wei, and Hui Jiang. Distraction-based neural networks for modeling document. In *IJCAI*, pages 2754–2760, 2016.

[Cheng and Lapata, 2016] Jianpeng Cheng and Mirella Lapata. Neural summarization by extracting sentences and words. *arXiv preprint arXiv:1603.07252*, 2016.

[Cheng *et al.*, 2016] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. Wide & deep learning for recommender systems. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*, pages 7–10. ACM, 2016.

[Chopra *et al.*, 2016] Sumit Chopra, Michael Auli, and Alexander M Rush. Abstractive sentence summarization with attentive recurrent neural networks. In *NAACL-HLT*, pages 93–98, 2016.

[Conroy and O'leary, 2001] John M Conroy and Dianne P O'leary. Text summarization via hidden markov models. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 406–407. ACM, 2001.

[Erkan and Radev, 2004] Günes Erkan and Dragomir R Radev. Lexpagerank: Prestige in multi-document text summarization. In *EMNLP*, volume 4, pages 365–371, 2004.

[Huang *et al.*, 2015] Zhiheng Huang, Wei Xu, and Kai Yu. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*, 2015.

[Kågebäck *et al.*, 2014] Mikael Kågebäck, Olof Mogren, Nina Tahmasebi, and Devdatt Dubhashi. Extractive summarization using continuous vector space models. In *Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC)@ EACL*, pages 31–39, 2014.

[Kupiec *et al.*, 1995] Julian Kupiec, Jan Pedersen, and Francine Chen. A trainable document summarizer. In *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 68–73. ACM, 1995.

[Lin and Hovy, 2003] Chin-Yew Lin and Eduard Hovy. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *NAACL-HLT-Volume 1*, pages 71–78. Association for Computational Linguistics, 2003.

[Luong *et al.*, 2015] Minh-Thang Luong, Hieu Pham, and Christopher D Manning. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*, 2015.

[McDonald, 2007] Ryan McDonald. A study of global inference algorithms in multi-document summarization. *Advances in Information Retrieval*, pages 557–564, 2007.

[Mihalcea and Tarau, 2004] Rada Mihalcea and Paul Tarau. Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, 2004.

[Mihalcea, 2005] Rada Mihalcea. Language independent extractive summarization. In *Proceedings of the ACL 2005 on Interactive poster and demonstration sessions*, pages 49–52. Association for Computational Linguistics, 2005.

[Mikolov *et al.*, 2013a] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[Mikolov *et al.*, 2013b] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[Nallapati *et al.*, 2016] Ramesh Nallapati, Bowen Zhou, Caglar Gulcehre, Bing Xiang, et al. Abstractive text summarization using sequence-to-sequence rnns and beyond. *arXiv preprint arXiv:1602.06023*, 2016.

[Nallapati *et al.*, 2017] Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. *hiP (yi= 1— hi, si, d)*, 1:1, 2017.

[Narayan *et al.*, 2017] Shashi Narayan, Nikos Papasarantopoulos, Mirella Lapata, and Shay B Cohen. Neural extractive summarization with side information. *arXiv preprint arXiv:1704.04530*, 2017.

[Paulus *et al.*, 2017] Romain Paulus, Caiming Xiong, and Richard Socher. A deep reinforced model for abstractive summarization. *arXiv preprint arXiv:1705.04304*, 2017.

[See *et al.*, 2017] Abigail See, Peter J Liu, and Christopher D Manning. Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368*, 2017.

[Takase *et al.*,] Sho Takase, Jun Suzuki, Naoaki Okazaki, Tsutomu Hirao, and Masaaki Nagata. Neural headline generation on abstract meaning representation.

[Vinyals *et al.*, 2015] Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. Pointer networks. In *Advances in Neural Information Processing Systems*, pages 2692–2700, 2015.

[Wong *et al.*, 2008] Kam-Fai Wong, Mingli Wu, and Wenjie Li. Extractive summarization using supervised and semi-supervised learning. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 985–992. Association for Computational Linguistics, 2008.

[Woodsend and Lapata, 2010] Kristian Woodsend and Mirella Lapata. Automatic generation of story highlights. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 565–574. Association for Computational Linguistics, 2010.