Research Paper

DOI: 10.3969/j.issn.1673-5188.2018.01.009
http://kns.cnki.net/kcms/detail/34.1294.TN.20180214.1416.002.html, published online February 14, 2018

Behavior Targeting Based on Hierarchical Taxonomy Aggregation for Heterogeneous Online Shopping Applications

ZHANG Lifeng, ZHANG Chunhong, HU Zheng, and TANG Xiaosheng
(Beijing University of Posts and Telecommunications, Beijing 100876, China)

Abstract

Behavior targeting (BT) based on individual web-browsing history has become more valuable in precision marketing for many companies through capturing users' interest and preference. It is common in practice that the behavior data collected from different online shopping applications are inconsistent since they are labelled by different item taxonomy, where the same behavior could have different representations and therefore analysis confusion arises. To address this issue, we propose a semantic similarity based strategy to transform the heterogeneous behavior extracted from deep packet inspection (DPI) data of a telecommunication operator into a unique standard one. The Word Mover's Distance algorithm is exploited to evaluate the semantic similarity of the distributed representations of two web-browsing histories. Moreover, the architecture of the behavior targeting platform on Hadoop is implemented, which is capable of processing data with size of PB level every day.

Keywords

BT; online shopping application; DPI; Word Mover's Distance; hierarchical taxonomy

1 Introduction

In the era of mobile Internet, ubiquitous network provides users with convenient service through mobile phones. This directly leads to large amounts of behavior data transmitted in the network pipeline. The behavior data is the uniform resource locator (URL) that a user views. It can be mined to learn user's interest and preference, further, to identify potential buyers from the large amounts of Internet users. With the development of deep packet inspection (DPI) technology, most operators and Internet service providers (ISPs) use it to extract users' behavior data and phone information (e.g. URLs, user agents, and phone numbers) for data mining [1], [2]. Furthermore, with the wide spread use of mobile phones, more and more people tend to choose online shopping on the phone due to its convenience, infinite choice, and lower price [3]–[5]. Compared with the offline shopping, online shopping behavior data can be collected through the DPI technology continuously, so that users' behavior can be consistently mined and then tagged with a semantic label to explain it.

In behavior targeting (BT) methodology, individual web-browsing behaviors are used to identify users' interest and preference. Further, advertising can take advantage of BT to achieve precise marketing. Therefore, it is of great significance to online advertising and it has been studied for applications. Natasha Singer has mined data tastes of music in Pandora [6]. WU Zenghong et al. have studied users' interest in map service based on browse behavior [7]. ZHU Qiushan et al. analyzed users' interest in video recommendation [8]. Although some studies have been conducted on online shopping platforms [9], [10], BT has not been studied on online shopping due to the heterogeneity of online stores and different hierarchical taxonomies between online stores.

For users' behavior on online shopping platforms, one of the key technical issues in BT is the problem of how to generate labels based on URL with accuracy, comprehensiveness, consistency and semantic, which mainly represents the behavior of a user, Therefore, we have to extract the information of items (products) that a user is browsing. Online shopping stores use stock-keeping units (SKU) as a unique ID to represent a unique item [11], and a SKU is transmitted in the URL when the user is browsing it. However, the SKU is just a code and it does not have semantic meaning. So there is no help on explaining users' behavior. However, the item represented by a particular SKU has a hierarchical taxonomy label on the website of online store. This hierarchical taxonomy label is a comprehensive and accurate description of the item. More importantly, the hierarchical taxonomy label that implies users' interest and preference can describe users' online shopping behavior, which is the target of BT.

BT on the shopping platform based on DPI has the following challenges. First, SKU transmits in the URL through URL parameters. The parameters are made of a key and value separated by an equals sign (=) and joined by an ampersand (&). We identify the SKU by the key from the URL. Due to the heterogeneity of online stores, the key is different between them, which makes it hard to extract SKU from different online stores. Secondly, heterogonous online stores have different taxonomy systems, which leads to the situation where one item has inconsistent representations but similar semantic labels, and may easily cause confusion. Therefore, these challenges must be con-

quered to construct accurate, comprehensive, consistent and semantic labels on users' behavior.

The main contribution of our work is the development of an extensive methodology for attaching a semantic label to users' online shopping behavior and implement this methodology on Hadoop platform. Our methodology addresses all the above challenges. First, we adopt the Word Mover's Distance (WMD) algorithm to handle inconsistent hierarchical taxonomy labels due to the different taxonomy of heterogonous online shopping applications. Second, our work extracts the item ID from URL according to the rules of regular expression. We analyze the key of SKU in the URL and find it delivered in several forms in every single online shopping application. A bunch of rules are then summarized for the online shopping applications we studied. Third, we collect the hierarchical taxonomy labels corresponding its SKUs through the web crawler. Finally, we design and implement a platform to achieve our purpose.

Our intention of this work is to develop an extensive methodology for BT on users' online shopping behavior, and detect interest-based targeting. By this we hope to provide operators and ISP with BT, which supports further data mining, such as user profiles and precision marketing [12]. The rest of the paper is organized as follows. In Section 2, we introduce the methodology of our data analysis from input to output. The implementation of the methodology based on Hadoop is presented in Section 3. Section 4 then gives the conclusion and future research directions.

2 Methodology

In this section, we introduce the methodology of attaching a hierarchical and semantic label to users' online shopping behavior based on DPI data. As for the data source, we introduce two kinds of data we used in our analysis. Then in the label processing part, we adopt the WMD algorithm to aggregate the labels with same semantic meaning but different representations. In the DPI processing part, the rules of regular expression are made to extract the item ID from URL and query the final label according to the item ID. **Fig. 1** shows the overview of the data analysis model.

2.1 Data Source

2.1.1 DPI Data

In our analysis, DPI data was provided by one of the largest operators in China. It contains more than tens of millions anonymized mobile phone data records in a period of two months in 2016. The data fields we used in our methodology are presented in **Table 1**, and other 33 data fields are omitted.

In the Table 1, the international mobile subscriber identity (IMSI) is the unique identifier tied with unique users. Moreover, it is encrypted by Message-Digest Algorithm (MD5) for privacy and security concerns. The URL represents the content that users are browsing, and only URL from online shopping applications is retained after being filtered by the domain name. Our intention is to extract the SKU, which is contained in the URL that users request from the massive DPI data. ST and ET are used to mark the behavior time of online shopping. To understand SKU easily, we denote ID as item ID in the following context.

2.1.2 Web Crawler

The item ID consists of a string of numbers or characters. It is an identifier of item and it does not have any semantic meaning. Hence it cannot help us target users' online shopping behavior. Therefore, we have to retrieve the hierarchical taxono-

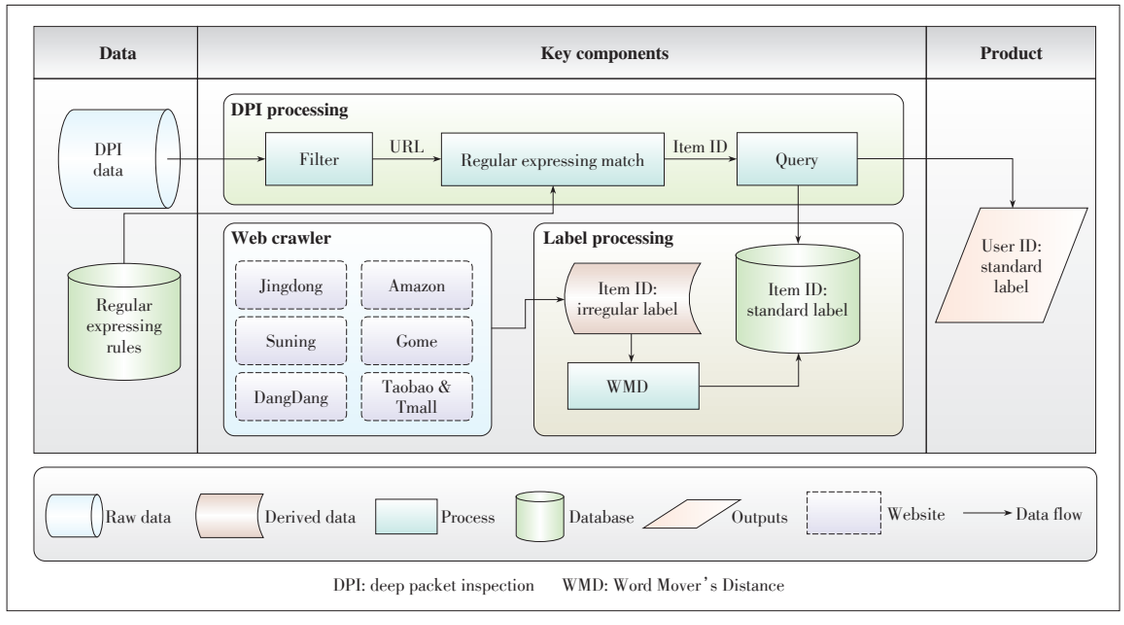


Figure 1.▶ The overview of data analysis model.

▼Table 1. The formats of DPI data

Serial number	Data field
1	IMSI
2	Start time
3	End time
4	URL
5	Domain name

IMSI: international mobile subscriber identity URL: uniform resource locator

my label through the web crawler. The hierarchical taxonomy label represents what users are browsing, and implies users' interest and preference.

According to iResearch's report, titled "Online Shopping Industry Monitoring Report in China 2016 [13]", the top six online shopping stores accounted for more than 80% of the online shopping market in China, and these top six stores are JingDong, Gome, Suning, Dangdang, Amazon, and Taobao & Tmall. We are focusing on retrieve hierarchical taxonomy labels from the above six online stores. In an online store, each product corresponds to a unique item ID while this item ID corresponds to a hierarchical taxonomy label. For example, an item from Jingdong is 133980, and its hierarchical taxonomy label is 'Men's Clothing → Bottoms → Pants'. As a result, when we extract an item ID from the URL, the hierarchical taxonomy label corresponding to this item ID can be attached to the behavior this time.

2.2 Label Processing

Because the above six online stores have different taxonomy systems, it can lead to the situation where one item has inconsistent representations but similar semantic labels and cause confusion. For example, in **Table 2** that shows the labels of iPhone 7 in four online stores, the strings before iPhone 7 represent the hierarchical taxonomy labels.

As Table 2 shows, these labels have the same meaning but different representations of hierarchical taxonomy, and the shortcoming of the original taxonomy is obvious. First, the labels in Suning and Gome are in a reversed form, i.e., Mobile Communication → Mobile Phones and Mobile Phones → Mobile Communication. Second, the label in JingDong repeats "Mobile Phones" which shows redundancy. Third, the corresponding level of taxonomy has different category grain. Last but not least, all these labels are used to describe iPhone 7, but the labels are different, which leads to confusion easily. Considering these drawbacks of original taxonomy, we normalize these labels to a unified meaning for easy understanding and this is useful for further data analysis.

▼Table 2. Different labels of iPhone 7 from four online stores

Online store	Hierarchical taxonomy
Jingdong	手机→手机通讯→手机 (Mobile Phones → Mobile Communication → Mobile Phones → IPhone 7)
Amazon	电子→手机通讯→手机 (Electronics → Mobile Communication → Mobile Phones → IPhone 7)
Suning	手机&数码→手机通讯→手机 (Phones& Digital → Mobile Communication → Mobile Phones → IPhone 7)
Gome	手机→手机通讯→手机 (Phones → Mobile Communication → Phones → IPhone 7)

We adopt one consistent taxonomy label to represent those similar semantic labels. In this paper, we call the raw hierarchical taxonomy labels crawled from the website as irregular labels. Our intention is to construct a consistent hierarchical taxonomy system based on semantic meaning. The system aggregates those similar semantic irregular labels to a unified one, and maps all these irregular labels to a standard label by standard label system. The basis of mapping irregular labels is the method of semantic similarity, which means we map irregular labels to a standard label with semantic similarity. We will also introduce how to achieve our intention through calculating the similarity between irregular and standard labels.

2.2.1 WMD Based on Word2vec

In our analysis, the irregular label and standard label are both hierarchical and contain several words, because we cannot calculate label semantic similarity directly. However, we creatively consider the label (irregular and standard) as a document, and calculate the label similarity through WMD algorithm, which measures the similarity of two documents based on word2vec embedding. The algorithm was introduced by Matt J. Kusner et al. in 2015 [14]. Before elaborating WMD in detail, we introduce its basis—word embedding.

Word embedding is a language model and a kind of feature learning technique in natural language processing (NLP), where words or phrases from the vocabulary are mapped to vectors. Although one-hot representation and distributed representation can both handle word embedding [14], [15], all the methods essentially use distributed representation in some way. Distributed representation states that words appearing in the same contexts share the semantic meaning. Then semantic similarity between words can be represented by the distance of corresponding vectors

In 2013, T. Mikolov et al. introduced word2vec. It is a particular group of models for learning word embedding from corpus and based on distributed representation [16], [17]. Their model learned a vector representation for each word, using a (shallow) neural network language model. Specifically, they proposed a neural network architecture (Continuous Bag-of-Words model and the Skip-Gram model) which consists of an input layer, a projection layer, and an output layer to predict the nearby words. After training on a large data set, the semantic similarity between words can be represented by the spatial distance of the vectors. This model has the ability to learn rela-

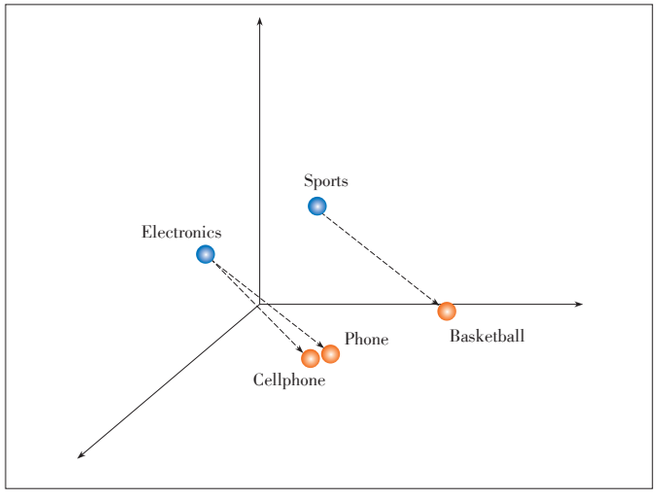
tionships of complex words (**Fig. 2**), which can be explained by the following equations:

$$v(phone) \approx v(cellphone), \tag{1}$$

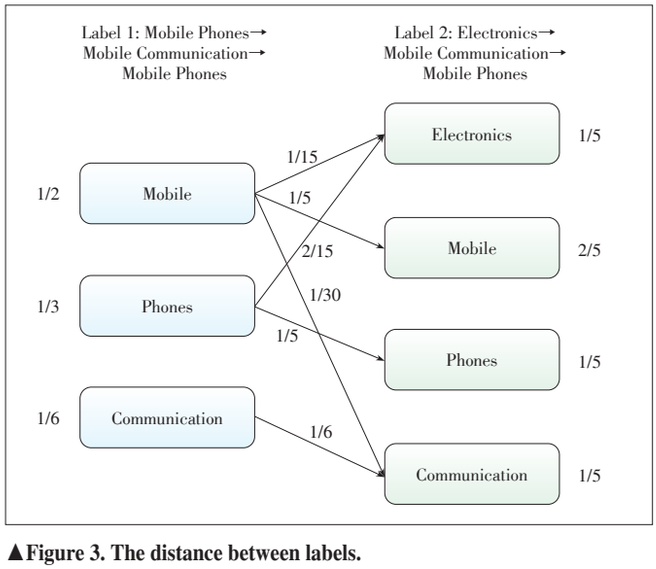
$$(electronics) - v(phone) \approx v(sports) - v(basketball). \tag{2}$$

The learning process of word embedding is unsupervised and it can be computed on the text corpus of interest or be precomputed in advance. Although we prefer word2vec to learn word embedding, other methods of word embedding are also feasible [18]–[20]. In the following introduction of WMD, we assume that a finite vocabulary size of n words is trained by word2vec according to the specific corpus and each word in the vocabulary is represented by a vector.

In **Fig. 3**, Label 1 is Mobile Phones → Mobile Communication → Mobile Phones, and Label 2 is Electronics → Mobile Communication → Mobile Phones. Because we calculate label distance through word distance, word segmentation must be used to split the label. Fig. 3 shows Labels 1 and 2 after word segmentation. The number beside the word is the weight occupied in the corresponding label. We denote the word in Label 1 as the word i and that in Label 2 as the word j. The weight of the word is denoted by $w_{word\,i}$. It represents the frequency of the word divided by the number of total words in this label after word segmentation, because we assume that labels are represented as a normalized bag of word (nBOW) vectors. The distance between the words i and j is denoted as $d(word\,i, word\,j) = \left|\left| word\,i - word\,j \right|\right|_2$. Then we will show how to calculate the label distance by word distance.



▲**Figure 2. Words relationship after training by word2vec.**



▲**Figure 3. The distance between labels.**

First, each word i in Label 1 is calculated with any word in label 2 in total or in parts, and we use T_{ij} to denote how many word i are involved in distance calculation with the word j. Second, to make total weights of the word involved in the distance calculation, T_{ij} should satisfy the equation $\sum_j T_{ij} = w_{word\,i}$ and $\sum_j T_{ij} = w_{word\,j}$. At last, the distance between two labels can be defined as the minimal cumulative distance of words distance. Naturally, the following linear program provides the minimal cumulative distance of Labels 1 and 2. More details can be found in [14].

$$D = \min \sum_{i,j} T_{ij} d(word\,i, word\,j), T_{ij} \geqslant 0, \tag{3}$$

and (3) is subject to (4) and (5):

$$\sum_j T_{ij} = w_{word\,i} \ \forall i \in \{1,2,\cdots,n\}, \tag{4}$$

$$\sum_i T_{ij} = w_{word\,j} \ \forall j \in \{1,2,\cdots,m\}. \tag{5}$$

According to the WMD algorithm, the distance between two documents can be calculated, but how to construct a standard label system and map the irregular label to a standard one is not mentioned. Then we will introduce the construction of a standard label system and label mapping based on the WMD algorithm.

2.2.2 Standard Label System

Because of the different taxonomy systems, labels crawled from different online stores have different representations for a particular product. Consequently, we put forward a standard label system considering the diversity of online stores. Then we map all irregular labels to standard labels. We find out that the hierarchical taxonomy of Gome is more reasonable and more complete than the other online stores. Therefore, we take the label system of Gome as the base standard label system, and complete it according to labels from other online stores. **Table 3** shows the samples of the standard label system.

▼Table 3. The samples of standard label system

Label ID	Standard label
001001001	手机数码→手机通讯→手机(Phone & Digital → Mobile Communication → Mobile Phone)
001001002	手机数码→手机通讯→对讲机(Phone & Digital → Mobile Communication → Interphone)
001002001	手机数码→手机配件→移动电源(Phone & Digital → Phone Accessories → Mobile Power)
001002002	手机数码→手机配件→蓝牙耳机(Phone & Digital → Phone Accessories → Bluetooth Earphone)
002001001	电脑→电脑整机→笔记本(Computer → Computer machine → Laptop)
002001002	电脑→电脑整机→台式主机(Computer → Computer machine → Desktop)
002002001	电脑 办公设备→打印机(Computer → Office Equipment → Printer)

In the standard label system, each label has a unique ID, which is also hierarchical. The label and its label ID both have three levels, and each three numbers (001-999) in a label ID correspond to a phrase in the label. In the data processing of DPI data, we use the label ID instead of the label to save storage and perform data analysis. We then construct the standard label system based on WMD.

First of all, we deduplicate the raw taxonomy label of Gome and take it as the initial standard label. Second, other labels are merged to this tree according to the WMD algorithm. If the distance between two labels is smaller than the threshold ε, we think these two phrases have the same semantic meaning. That is to say, these two labels can be replaced with each other semantically. The pseudo code of constructing the standard label system algorithm is elaborated in **Algorithm 1**.

Algorithm 1: Construct Standard Label System

```
Input: labels from six online stores.
Output: the standard label system.
Initial parameters:
    deduplicate label taxonomy of Gome as the standard label system, denoted as Standard_System
    deduplicate label taxonomy of other online stores, denoted as Irregular_System
for ire_label in Irregular_System {
    minDistance = INTEGER.MAX_VALUE
    for sta_label in Standard_System {
        tempDistance = WMD(ire_label, sta_label)
        If (tempDistance < minDistance) {
            minDistance = tempDistance
        }
        If(minDistance > ε){
            Standaard_System.add(ire_label)
        }else{
            continue
        }
    }
}
traversing the Standard_System, output the label and encode with its label ID
```

2.2.3 Label Mapping

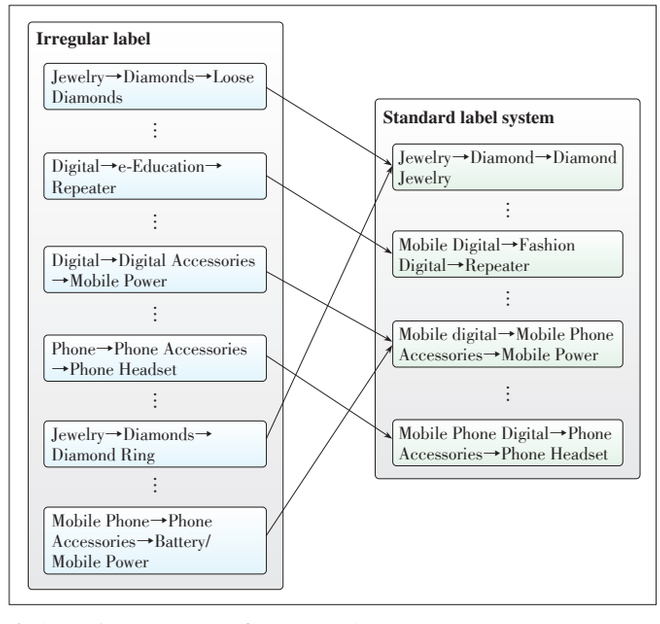
After constructing the standard label system, we map all irregular labels to standard labels for consistency. The purpose of label mapping is to find the label which has the greatest semantic similarity in the standard label system for every irregular label. In other words, we should find a label in the standard label system to satisfy a given irregular label.

$$\min WMD(irregular\ lable, sl)\ sl \in standard\ label\ system. \quad (6)$$

We can easily understand the label mapping from **Fig. 4**. Given an irregular label in the left, we need to find a label from the standard label system in the right to satisfy (6). For an irregular label in a nutshell, we need to find a label in the standard label system to achieve the minimal label distance. At last, every irregular label is mapped to a standard label.

2.3 DPI Processing

In this part, our intention is to extract the item ID from URL following a bunch of rules of regular expression (Regex), a string matching algorithm. The algorithm is designed for "find" or "find and replace" operations on strings and it is perfectly suitable for our needs. But it still has two problems. First, URL contains multiple information in the form of parameters and we have to recognize which part involves item ID. Sec-



▲Figure 4. The example of label mapping.

ond, we have to consider six online shopping stores, which increases the difficulty because the keys of item ids are different between heterogonous online stores. Next we will introduce our procedure of processing DPI data and solutions to the above problems.

First of all, we filter out the data of the six online shopping applications from DPI data through the domain name. Second, we extract the item ID from the URL through Regex match. At last, we query the corresponding label according to the item ID from the database. But how to get the Regex? We summarize it manually from the large raw DPI data for particular online store, and then we introduce our methodology to summarize the Regex in an example of Jingdong.

In the beginning, we filter out a large number of URLs from Jingdong. Then we identify the key of item ID from tens of parameters of URL. In this process, we find out there are several forms of the key even in one online shopping application. Finally, the item ID with the form of key-value is transformed to Regex manually. **Table 4** shows the Regex matching item IDs in Jingdong. All these Regex has been tested by real URL that users have browsed.

Through the way above, the Regex of other five online shopping applications can also be summarized. **Table 5** shows the number of Regex of these applications.

▼**Table 4. The Regex of item IDs in Jingdong**

Serial number	Regex
1	ware[iI]d|sku(=|%3D)(\d+)
2	order.*ware[iI]d(=|%3D)(\d+)
3	jd\.com/(product/)?(\d+)\.html
4	productIds=(\d+)
5	orderComment/(\d+)
6	item\.jd\.com/(\d+)

▼**Table 5. The number of Regex of the six applications**

Application	Number
Jingdong	6
Suning	5
Amazon	4
Gome	4
Dangdang	2
Taobao & Tmall	4

3 Implementation Based on Hadoop

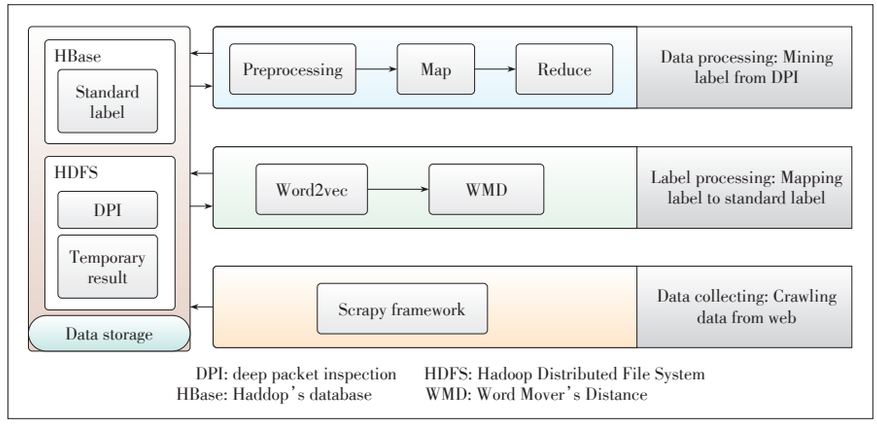
The data scale of telecom network has reached to PB level for each day, so a big data platform with high reliability and high effectiveness is extremely important for operators. We design a big data platform based on Hadoop to achieve our goal through the above methodology. In a nutshell, users' behavior can be consistently mined and attached to a label on this platform.

The architecture of the platform consists of four modules (**Fig. 5**). They are data storage, data collecting, label processing and data processing.

3.1 Data Storage

In the era of big data, it is impossible to store extremely massive amounts of data in a single machine with varieties of demands. Therefore, alternative technologies have been investigated in order to solve this issue. Hadoop Distributed File System (HDFS) works as a part of a Hadoop cluster or as a stand-alone universal distributed file system. It is widely used as a storage system in industry area because of its high stability and scalability. In our data storage module, we also adopt HDFS as our storage system for DPI data. Haddop's database (HBase) is a non-relational database and HDFS is served as its physical storage. However, their functions are different in our system.

1) HDFS. In our system, HDFS mainly stores two kinds of files. One is the raw DPI data from telecom operators, which generates every day in a directory named by date and province. The other is temporary result, it contains the raw item labels crawled from the web, the standard labels after label processing and the final results of data processing module, that is, the user ID, timestamps and standard label connected by comma.
2) HBase. We adopt HBase to store the item ID and its standard label. HBase can support storage and query in the form of key-value pairs compared with HDFS, which just meets our demands for the storage of item ID and standard label and the query by item ID. Second, HBase is a member of Hadoop, which can be well integrated with MapReduce in data processing.



▲**Figure 5. Data analysis architecture.**

Before we adopt HBase as our key-value database, we compared the performance of HBase and Redis [21], which is a key-value database based on in-memory storage. The results show that Redis supports higher concurrent requests, while HBase can also satisfy our concurrent requests. Finally, considering the stability and price, we adopt HBase as our database to store item information.

In our implementation, the format of item information and samples are shown in **Table 6**. The column family: qualifier and timestamp can be set by a default value, while Rowkey and cell value must store the corresponding item ID and its standard label. Furthermore, six tables need to be created to store the information of six different online stores.

3.2 Data Collection

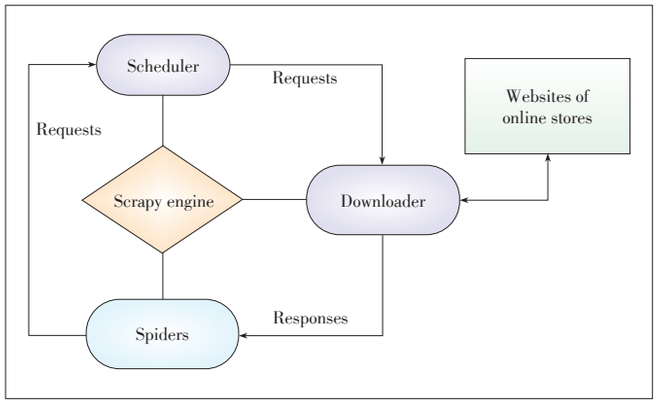
This module is used to collect data from websites, in other way, crawl the item ID and irregular labels from the online store. We implement this function to extract the data from websites based on an open source and collaborative framework named Scrapy [22]. **Fig. 6** shows the framework of Scrapy.

This framework is responsible for crawling item information from the website without considering the scheduler and downloader because they are implemented by the framework. All we need to do is to implement the spider module in the Fig. 6, and focus on the design of our spider. The design of our spider takes hierarchical labels into consideration, because three-level hierarchical label corresponds to three-level hierarchical websites. Therefore, we start our web crawler from the starting page, which corresponds to the first level of the hierarchical label. Then we find all secondary page and save it. Furthermore, all the secondary pages are crawled and the last level page of the website is saved. At last, we crawl all last pages and save all hierarchical labels.

We need to pay attention to one problem, that is, the item information is always changing slowly. Therefore, spiders should be launched periodically to re-crawl the item information. The new items browsed by users cannot be attached to a label because we have not crawled this item information. Therefore, we calculate the percentage of the URL from which we can match the item ID but cannot get the label. Once this percentage is greater than 5, we launch our spiders and re-crawl the website. As for new item information, if item ID exists in the last version, we just update the hierarchical label when it changes, and if the item ID does not exist, we add this item information to HBase.

▼**Table 6. The store format of item information and samples**

Rowkey	Column family: qualifier	Timestamp	Cell value
Item ID	"label:standard"	default	Standard label
4005363	"label:standard"	default	(Computer → Computer → Laptop)
3726830	"label:standard"	default	(Phone & Digital → Mobile Communication → Mobile Phone)



▲**Figure 6. The framework of Scrapy.**

3.3 Label Processing

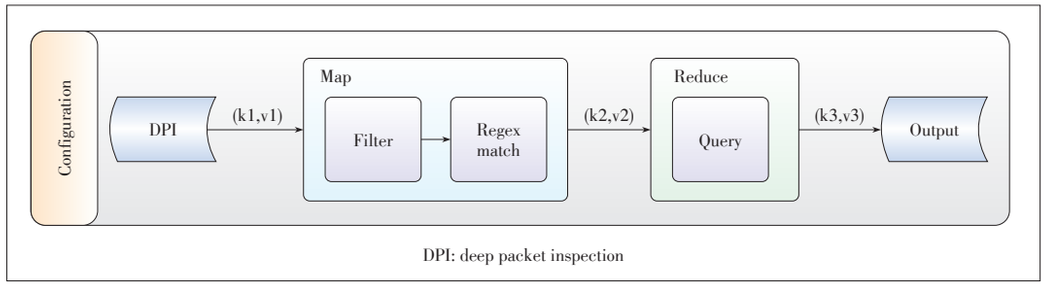
This section focuses on transferring the item ID → the irregular label to the item ID → standard label. The main algorithm is introduced in Section 2.2, and the realization of the label processing is as follows.

1) Construct a vocabulary of word embedding, which is trained by word2vec according to the specific corpus from the web. The word2vec is trained with the corpus from Sogou Labs [23]. It contains various types of content of 130 million original web pages and the amount reaches 5 TB. More importantly, the corpus may directly influence the accuracy of WMD, so a large corpus should be adopted.
2) Construct the standard label system based on the label taxonomy of Gome. In this step, threshold ε should be studied to make the standard label reasonable. We made experiments with different values of ε and found that $\varepsilon = 1.1$ makes the standard label system more reasonable. In this condition, the standard label system has 2153 hierarchical labels, which contains 25 first-level phrases, 296 second-level phrases and 1620 third-level phrases.
3) Map the item ID → the irregular label to the item ID → standard label according to the Algorithm 1. After mapping the irregular label, every item ID corresponds to a unique standard label.
4) Store the item ID → the standard label to HBase in corresponding tables. In this step, we create six tables with the same names of the corresponding online store. The item ID → standard label is written into the corresponding table in the format mentioned in Section 3.1.

3.4 Data Processing

In this part, we will focus on the processing of massive amounts of DPI data based on Hadoop MapReduce, which is the most popular open source implementation of the MapReduce framework proposed by Google [24]. The feature of Hadoop MapReduce is high fault tolerance and scalability. It is easy to program and perfectly suitable for our demands [25],

[26]. There are two stages named map and reduce in a Hadoop MapReduce job, we only have to define the map and reduce to finish our job. The input and output formats of map and reduce are shown in **Fig. 7**. The formats are denoted as a set of key-value pairs (key, value). The procedure of DPI data processing is as follows.



▲**Figure 7. Implementation of Hadoop MapReduce.**

1) Map

The Map process is to transfer the URL to item ID based on regular expression matching, as shown in Fig. 1. We take two modules to achieve this function in this phase, they are Filter and Regex Match.

Filter: The input format is key-value pairs of raw DPI data, which contains a lot of information. The key is offset of current line in the file and the value is raw DPI data. Filter extracts the information we need, including the use ID, URL, timestamp and domain name shown in Table 1, abandoning other information such as user agent, data size, protocol, and IP. At the same time, Filter also extracts the data from the top six online stores according to their domain names and abandon other DPI data. Moreover, we take IMSI encrypted by MD5 for privacy and security concerns as unique ID of users. In a nutshell, the output of Filter is in the form of key-value pairs (encrypted IMSI, URL from top six online stores| timestamp| domain name).

Regex Match: This module handles all URLs after Filter through Regex matching to extract the item ID. It is difficult to follow tens of rules of Regex from the top six online stores to match the item ID. We adopt two strategies to handle this problem. One is matching all rules for one URL and ignoring which application the URL is from. We call this strategy Global Match. The other one is identifying which application the URL is from first, and then matching the URL based on rules from the corresponding application. We call it Partial Match.

We found that Partial Match costs less time than Global Match based on our experiment in the **Table 7**. In the experiment, the number of nodes in our Hadoop cluster is 104, and the time refers to the duration of the Map phase. The final output of the Map phase is key-value pairs (encrypted IMSI, item ID | timestamp | domain name).

2) Reduce

Reduce finishes the last step of our data processing, that is, querying the standard label from the corresponding tables in HBase. In this process, we need to construct six table connections to HBase and then get the standard label by item ID. The item ID from the Map process is queried and then output to the (encrypted IMSI, standard label and timestamp) to a directory.

3) Configuration

Some necessary configuration parameters need to be set first in the job of launching a MapReduce. For example, the input and output format of Map and Reduce, the directory of original input and final output, and the number of Reduce.

3.5 Results

After all the work is done, we achieve the BT on the DPI data, which can attach a Hierarchical label on user behavior. **Table 8** shows the results of our methodology and implementation.

The IMSI is the unique ID tied with unique users, and encrypted by MD5 for privacy and security concerns. The URLs represent users' browsing behavior and the labels are the BT on their behavior. These labels imply users' interest and preference, which will help identify potential buyers from the large amounts of Internet users.

4 Conclusions

We developed an extensive methodology for BT on users' online shopping behavior. The methodology is based on hierar-

▼**Table 7. Experiment results**

DPI data size	Time cost in map	
	Global match	Partial match
2.7 Tb	1 h 13 min	46 min
470 Gb	17 min	10 min

▼**Table 8. The final results of the proposed scheme**

IMSI	URL	Label
898A93AA58976A4945 6222D58420B6B1	http://item.m.jd.com/ product/3368118.html	Online shopping → Appliances → Health appliance
4B0DA828BDF06C1C2 1BC8926456402CA	http://cd.jd.com/img/ channel?callback= jQuery6841693	skuId=10483088139&_= 1503617852363& Online shopping → Home building → Home textile cloth → Sheets
45DCF28D2D947447F3 C9B87E24491131	http://product.dangdang. com/23215376.html	Online shopping → Books → Political/Military
DF6D96FC9B386DABF 2E5C0AADB508BC1	http://item.m.gome.com.cn/ product-A0004771496- pop8003858741.html? cmpid=seo_baidu_kapian	Online shopping → Home appliance → Personal care → Shaver

IMSI: international mobile subscriber identity URL: uniform resource locator

chical and semantic taxonomy aggregation. As a result, we can attach a hierarchical and semantic label to online shopping behavior, which will help identify potential buyers from the large amounts of Internet users and achieve precision marketing. We adopted the WMD algorithm to aggregate similar semantic labels to a unified label, and implemented our methodology on a big data platform. It performed efficiently to mine users' behavior on online shopping applications.

Acknowledgement

The authors would like to thank Beijing University of Posts and Telecommunications, China and China Telecom for cooperation and support for this paper.

References

[1] H. Asghari, M. van Eeten, J. M. Bauer, and M. Mueller, "Deep packet inspection: effects of regulation on its deployment by internet providers," in *41st Research Conference on Communication, Information and Internet Policy*, Arlington, USA, 2013. doi: 10.2139/ssrn.2242463.
[2] R. Antonello, S. Fernandes, C. Kamienski, et al., "Deep packet inspection tools and techniques in commodity platforms: challenges and trends," *Journal of Network and Computer Applications*, vol. 35, no. 6, pp. 1863–1878, Nov. 2012. doi: 10.1016/j.jnca.2012.07.010.
[3] K. Sha, "Trends and issues related to online shopping market in China," in *IEEE 6th International Conference on Information Management, Innovation Management and Industrial Engineering*, Xi'an, China, 2013, pp. 183–187. doi: 10.1109/iciii.2013.6703114.
[4] 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. doi: 10.1109/3468.852436.
[5] J. H. Wu, L. Peng, Q. Li, et al., "Falling in love with online shopping carnival on singles´ day in China: an uses and gratifications perspective," in *IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*, Okayama, Japan, 2016, pp. 1–6. doi: 10.1109/icis.2016.7550801.
[6] J. Chen and J. Stallaert , "An economic analysis of online advertising using behavioral targeting," *MIS Quarterly*, vol. 38, no. 2, pp. 429–449, 2014. doi: 10.2139/ssrn.1787608.
[7] W. Zenghong, C. Yufen, and Z. Jun, "Personalized map service user interest acquisition based on browse behavior," in *IEEE International Conference on Control Engineering and Communication Technology (ICCECT)*, Liaoning, China, 2012, pp. 916–919. doi: 10.1109/iccect.2012.225.
[8] Q. Zhu, M. L. Shyu, and H. Wang, "Video topic: modeling user interests for content-based video recommendation," *International Journal of Multimedia Data Engineering and Management (IJMDEM)*, vol. 5, no. 4, pp. 1–21, 2014. doi: 10.4018/ijmdem.2014100101.
[9] D. I. Maditinos and K. Theodoridis, "Satisfaction determinants in the Greek online shopping context," *Information Technology & People*, vol. 23, no. 4, pp. 312–329, 2010. doi: 10.1108/09593841011087789.
[10] R. Olbrich and C. Holsing, "Modeling consumer purchasing behavior in social shopping communities with clickstream data," *International Journal of Electronic Commerce*, vol. 16, no. 2, pp. 15–40, 2011. doi: 10.2753/jec1086-4415160202.
[11] M. Pazzani and D. Billsus, "Learning and revising user profiles: the identification of interesting web sites," *Machine Learning*, vol. 27, no. 3, pp. 313–331, 1997. doi: 10.1023/A:100736990.
[12] G. E. Hinton, J. L. McClelland, and D. E. Rumelhart, "Distributed representations," in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1*. Cambridge, USA: MIT Press, 1984, pp. 77–109.
[13] IResearch. (2017, May). *Online shopping industry monitoring report in China 2016* [Online]. Available: http://wreport.iresearch.cn/uploadfiles/reports/636228578101640793.pdf
[14] M. J. Kusner, Y. Sun, N. I. Kolkin, and K. Q. Weinberger, "From word embeddings to document distances," in *32nd International Conference on Machine Learning*, Lille, France, 2015, pp. 957–966.
[15] W. Y. Zou, R. Socher, D. Cer, and C. D. Manning, "Bilingual word embeddings for phrase-based machine translation," in *Conference on Empirical Methods in Natural Language Processing*, Seattle, USA, 2013, pp. 1393–1398.
[16] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *26th International Conference on Neural Information Processing Systems*, Lake Tahoe, USA, 2013: 3111–3119.
[17] T. Mikolov, K. Chen, G. Corrado, and J. Dean. (2013, Sept. 7). *Efficient estimation of word representations in vector space* [Online]. Available: arxiv.org/abs/1301.3781
[18] R. Collobert and J. Weston, "A unified architecture for natural language processing: deep neural networks with multitask learning," in ACM 25th International Conference on Machine Learning, Helsinki, Finland, 2008, pp. 160–167. doi: 10.1145/1390156.1390177.
[19] A. Mnih and G. E. Hinton, "A scalable hierarchical distributed language model," in *Neural Information Processing Systems (NIPS 2008)*, Vancouver and Whistler, Canada, 2008, pp. 1081–1088.
[20] J. Turian, L. Ratinov, and Y. Bengio, "Word representations: a simple and general method for semi-supervised learning," in *48th Annual Meeting of the Association for Computational Linguistics*, Uppsala, Sweden, 2010, pp. 384–394.
[21] Redis. (2017, May). *Redis* [Online]. Available: https://redis.io
[22] Scrapy. (2017, May). *Scrapy* [Online]. Available: https://scrapy.org
[23] Sogou. (2017, May). *SogouT* [Online]. Available: http://www.sogou.com/labs/resource/t.php
[24] J. Dean and S. Ghemawat, "MapReduce: simplified data processing on large clusters," *Communications of the ACM*, vol. 51, no. 1, pp. 107–113, 2008. doi: 10.1145/1327452.1327492.
[25] A. B. Patel, M. Birla, and U. Nair, "Addressing big data problem using Hadoop and Map Reduce," in *IEEE Nirma University International Conference on Engineering (NUiCONE)*, Ahmedabad, India, 2012. doi: 10.1109/nuicone.2012.6493198.
[26] J. Dittrich and J. A. Quiané-Ruiz, "Efficient big data processing in Hadoop MapReduce," *Proceedings of the VLDB Endowment*, vol. 5, no. 12, pp. 2014–2015, Aug. 2012. doi: 10.14778/2367502.2367562.

Manuscript received: 2017-08-05

Biographies

ZHANG Lifeng (zhanglifeng@bupt.edu.cn) is a postgraduate student at Beijing University of Posts and Telecommunications, China. His research interests include data mining and massively parallel processing of data.

ZHANG Chunhong (zhangch@bupt.edu.cn) is a lecture of School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, China. She received her Ph.D. degree in computer science, M.Eng. degree in information technology, B.Eng. degree in telecommunication engineering in 1993, 1996 and 2013 respectively. She was a visiting scholar at Illinois Institute of Technology, USA in 2015. Her research interests include data mining, natural language processing, and ubiquitous computing.

HU Zheng (huzheng@bupt.edu.cn) received his Ph.D. degree from Beijing University of Posts and Telecommunications, China in 2008. He is working in the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications. His current research interests lie in the user behavior modeling and analysis in mobile internet and social networks. He has published more than 30 papers and been granted more than 10 patents in related area.

TANG Xiaosheng (txs@bupt.edu.cn) received his Ph.D. degree from Beijing University of Posts and Telecommunications, China. He is working in the Beijing University of Posts and Telecommunications. His current research interests include user behavior modeling and analysis in mobile internet and social networks. He has published more than 20 papers and been granted more than 10 patents in related areas.