

An Approach for Extracting and Predicting Instance-specific Attribute Values from E-commerce Sites for Used Products

Hettiarachchige Dona Nidhana Harshika^{*}, Kihaya Sugiura[†],
Naoki Yamada[‡], Masahiro Nishi[§], Naoki Fukuta[¶]

Abstract

To obtain valuable products, reviews, rating, and other important product information given by sellers are important issues for consumers for their purchasing decisions. However, in the used-products purchasing scenario via e-commerce sites, consumers may consider much more attributes about the products than that for purchasing new products. This is due to the need for understanding instance-specific conditions before optimize their purchasing decision. Thus, the available descriptions for a used product may differ in each other, and it may drop some important information to make a decision for consumers. In this paper, we proposed a design and implementation of a system that supports users to investigate instance-specific attribute values by extracting and predicting attributes and values of used items that are selling on e-commerce sites. Our key idea is preparing a system to identify instance-specific attributes as well as their values from the descriptions of items while browsing the e-commerce sites. Also we have implemented floating style user interface to present the extracted and predicted instance-specific attributes. Our system also applies machine learning techniques to predict missing attributes values.

Keywords: E-Commerce, Used Products, Instance-Specific Attributes Values, Prototype System, User Interface

1 Introduction

The role of product recommendation has been more and more important. REEVOO, a social commerce solutions provider, surveyed [1] 1,000 consumers on shopping habits and found that 88% percent of them sometimes or always consult product recommendations before purchase. Product instance-specific attributes and their values, online reviews, ratings and

^{*} Graduate School of Informatics, Shizuoka University, Japan

[†] Graduate School of Integrated Science and Technology, Shizuoka University, Japan

[‡] Department of Computer Science, Shizuoka University, Japan

[§] Department of Computer Science, Shizuoka University, Japan

[¶] Department Of Informatics, Shizuoka University, Japan

sellers informations are some cornerstones of consumer decision making. Therefore there are many approaches on products recommendation. (e.g., [16]).

Although product recommendation is considered important to consumers, many recommendation methods are facing for data scarcity challenges[1]. For example, most of the POI recommendation [2] (Recommendation methods of point of interest) methods [3, 4, 5, 6, 7, 8] mentioned about the issues on data scarcity issues and presented various approaches to overcome them by utilizing implicit feedback of facts, for POI recommendations.

The number of consumers who are purchasing used products via online shopping sites would be increasing because of the increase of share online shopping. Of course every manufactures are realizing their product's new versions to face to their competitive market. Therefore a lot of products could become second handed products.

Compared to the cases to recommend brand new products, used products recommendation is more likely to suffer data scarcity issue since making recommendations for used products are depending on their instance-specific values. If we consider about a specific brand of new product, its product instances (actual physical items) have almost identical or very similar specification attribute values, (e.g., weights, size of screen, etc). Therefore, the same product could be purchased from two or more buyers so that we could utilize the other buyers' behaviors for brand-new products recommendation. However, if we consider about a specific used product, an item has its own conditions or special attributes to distinguish an item from others (e.g., used by a very popular person, etc.).

In this paper, we propose an idea, design, and a prototype implementation of a system to dynamically retrieve missing data for used product information descriptions when a user is browsing e-commerce sites to purchase those items. The contributions of this paper are shown as follows.

- We identify the problem of browsing data on used products, specially on instance-specific attribute values.
- To solve the problem, we give a design and a prototype of a new system that can dynamically analyze item descriptions of used products to extract important attributes for the item as well as extract or predict the attributes values.

2 Related Works

Product recommendations are one of the most important sources in e-commerce. Data scarcity is a vast challenging area for not only in product recommendation also in almost all recommendation systems. There are several surveys for the works to overcome, data scarcity problem in several research areas.

Geo Cong et al. [2] proposed a ranking based factorization method, Rank-GeoFM, for POI recommendation. In the proposed model, they learned the factorization by fitting the users' ranking for POIs, which alleviates the data scarcity problem.

Adham Beykikhoshk et al.[9] identified and explained the inherent difficulty associated with topic modeling on tweets-insufficient information content. They have proposed a framework which uses tweets not as end point data mining sources but rather as intermediaries used to discover much richer associated content.

There are approaches on new or unpopular products which are suffering from lack of recommendation specifications such as product reviews, as well. Park et al. [1] proposed an automatic relevant review text retrieval for products having no reviews. They have proposed

several methods to solve above mentioned issue, including summarization-based method such as MEAD and probabilistic method such as Specification Generation model.

Zhou and Chaovalit [10] performed sentiment classification on reviews using domain ontology database, which may be regarded as product specifications. Their approach aims to enhance polarity mining with ontology by providing detailed topic-specific information. Bhattattacharya et al. [11] employed IMDb's structured data to categorize documents, and Yu et al. [12] proposed to organize the aspects of a specific product into a hierarchy. Product reviews and specifications were jointly modeled using topic models by Duan et al. [13] to generate augmented specifications with useful information.

Although product specifications have been provided in many e-commerce web sites, there are only a limited or no number of studies that specially utilized descriptions of specifications for used product data analysis. Not only from a consumer's perspective but also from a used product seller's perspective, lack of used products specification data is a most challenging area for used product analysis.

However, to the best of our knowledge, no existing system has been presented to automatically extract and predict attribute values on product descriptions for used products. In this paper, we proposed a design and implementation of a system that supports such automated extractions and predictions of instance-specific attribute values from the given descriptions of the items in e-commerce sites while browsing them.

3 Our Approach

If we compare the existing systems specially using for the purpose of product recommendation in e-commerce web sites, all most all recommendation systems are depending on, products rating, customer rating, purchasing history of the items, sellers rating and so on. Figure 1, shows Amazon.com ratings page prompts the customer to rate items which have recently purchased. These ratings are used as input to a recommendation engine on Amazon.com web site, to help their customer to find other items that customer is likely to like. Customers are asked to invest efforts in rating, in exchange for which they get more useful recommendations on their page. J. Ben al. [20], have done an investigation on existing e-commerce recommendation applications specially on Amazon.com and e-bay etc. On their work they did not mention about the used products recommendation system. As we have described on Introduction section even for the specific product its conditions are different from item to item. Because of this differences could be difficulties on utilizing purchasing history for used items, since actually there is no identical item in a used item market. Therefore above mentioned rating also could have difficulties to be applied for the used products recommendation systems. Without considering the reliability of such customer ratings, we could have these serious issues when we apply such a method to used product markets.

To help users, make the best buying decision efficiently, the user may have read and understand the information which are provided by seller's, as well as the seller's information provided by the e-commerce sites such as ratings.

As we described in introduction, data scarcity issue on purchasing on an e-commerce site, it used products is more complicated, compared to purchasing of a brand new product, since all the used products may have their instance-specific attributes and their values.

We checked some of details about used products which advertised on Amazon.com web site[18], on 29th March 2016. We have identified that the specification of attribute values (details provide by seller about the product) are really different for each product.



Figure 1: Customers are asked to invest efforts in rating, in exchange for which they get more Useful Recommendations on a E-commerce Web Site

For example, the following are the information description which were provided by the seller about two different used products which advertised on Amazon.com on 31st of March 2015.

The description about the one, for “BLACK+DECKER EK700 9-Inch Electric Carving Knife, White” is shown in Figure 2.

Used - Like New

The electric carving knife is in excellent condition. It works perfectly with the blades. Blades are super clean and may have not been used. Looks like picture. Free delivery confirmation. Fast shipping. I am happy to answer any questions. Thanks!

Figure 2: Example Description of used carving knife on <http://www.amazon.com>

The description about another used product “APPLE iPhone 6 GOLD” is shown in Figure 3.

Used - Acceptable

iPhone 6 gold unlocked 16gb phone only 5/10 dings and scratches very rough cosmetic condition but 100% functional

Figure 3: Example Description of used iPhone on <http://www.amazon.com>

By analyzing some used products examples on amazon site, as above mentioned examples, and those product descriptions seems significant for used products purchasing scenario via e-commerce sites. Since in each other characteristic of those descriptions are very different. This issue is left even when we assume the given product descriptions are correct. As an example scenario, we choose purchasing used apple iPhone, which having a very good second hand market all around the world in all most all e-commerce web sites. Figure 4 shows of the structure of process has done in our system.

Since our system runs in between the users and the e-commerce site, the system itself has little amount of used products purchasing history, since it may not have sufficient data to model the users unless users already brought lots of items via our system. Therefore we did not direct recommendation functions or methods on our system.

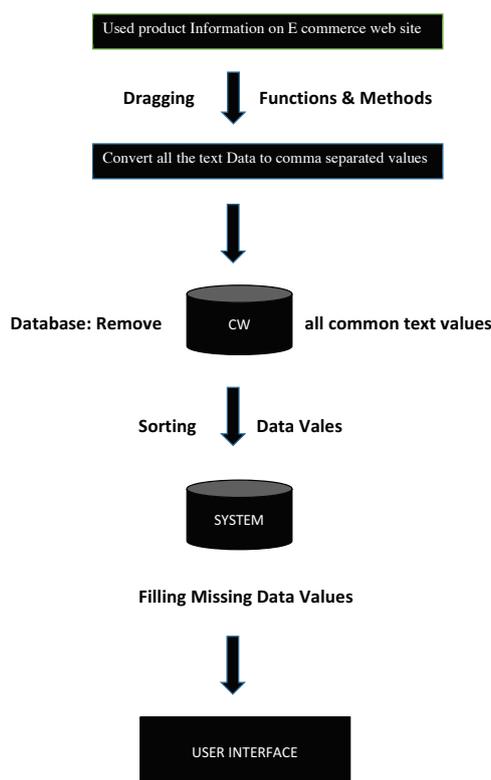


Figure 4: Overview of the Processing Stage on Our System

3.1 Extracting Instance-specific Attributes

We have manually extracted data about used apple iPhone descriptions which provided by seller on Amazon.com web site[19], advertised on 31st March 2015, which explain in Figure 5. About the third item on Figure 5, we extracted those information as a text document as shown in Figure 6. These extracted data included all the words which seller used to describe its specification values.

3.2 Extracting Attribute Terms

In order to separate each terms that may explain characteristics text document, we converted above text document as comma separated text values for analyzing purposes as in Figure 7.

Algorithm 1 shows the basic process of extracting attributes and their values from Web pages. As written in line 1 and 2 of the Algorithm 1, the text descriptions of the items in the target pages are extracted. Then, as written in line 3 and of the Algorithm 1, the text is tokenized and split into a set of words shown in Figure 7. We have sorted the extracted term data in order to duplicates data values. By using a common word database we have removed all commonly used words which will have low weight of information gain about explaining used iPhones. In this example case, we have applied 30 commonly used stop words as this common word database¹. By doing this step we are able to identify significant attributes

¹To minimize the effect of this common word database to the experiments, we have prepared 30 commonly used stop words by ourselves. This can be improved by using a statistical approach such as IDF (inverse-



Figure 5: Structure of Items, Information Given by Sellers on Amazon.com

GSM Unlocked (Will work with AT&T, T-Mobile, etc). Will NOT work with CDMA carriers such as Verizon, Sprint, Virgin Mobile, etc. Tested and fully functional. Includes generic charger and USB cable. 30 day warranty. Unlocked cell phones are compatible with GSM carries like AT&T and T--Mobile as well as with GSM SIM cards (e.g. H20 with CDMA carriers and select prepaid carriers). Unlocked cell phones will not work like Sprint, Verizon, Boot or Virgin.

Figure 6: Description data

related to this specific used iPhones. In this case, we have extracted total 310 attributes. They included some functional attributes such as ‘GSM’(a synonym of Global System for Mobile Communications, which is a standard of protocols for mobile cellular networks) or ‘CDMA’(a synonym of Code-Division Multiple Access), as well as non-functional attributes such as ‘clean’. However, due to the limitation of the approach, we cannot extract the degree of such non-functional attributes such as ‘how clean the display is’. This is one of limitations in our approach.

Then we have arranged all text data, and sorted those data to gathered equal words on equal column. And turning those words to attributes values as used iPhone products. As long as used the attributes directly obtained from products descriptions provided by sellers, they are different for each item instance, and some of most significant attribute’s values are missing in each rows. Here we mentioned those missing values as “UNKNOWN” on each missing value positions, as follows.

document–frequency) based word ranking.

GSM, Unlocked, (,Will, work, with ,AT&T, T-Mobile, etc,), Will, NOT,
work, with, CDMA, carriers, such, as,.....

Figure 7: Tokenised Text Description

GSM, Unlocked, AT&T, T-Mobile,CDMA,Tested,...
Functional,Included,warranty

Figure 8: Extracted Attributes

GSM	UNLOCKED	CDMA
YES	UNKNOWN	NO
NO	YES	NO
UNKNOWN	YES	UNKNOWN
YES	NO	YES
YES	UNKNOWN	UNKNOWN
YES	YES	YES
NO	YES	NO
UNKNOWN	UNKNOWN	UNKNOWN

Algorithm 1 Generate List of Attributes

Require: *URL*

- 1: *HTMLData* \leftarrow *getHTMLData(URL)*;
 - 2: *details of UsedItem* \leftarrow *extracted Details(HTMLData)*;
 - 3: *list of keywords* \leftarrow *parse(details of UsedItem)*;
 - 4: *attribute values(YES/NO/UNKNOWN)* \leftarrow *set Attributes(List of keywords)*;
 - 5: *list of keywords and attribute values* \leftarrow *remove not Significant Attributes(attributes)*;
 - 6: *generate CSV Data(attribute values,List of keywords)*;
-

3.3 Predicting Missing Attribute’s Values

After we gathered all the data on used iPhone on the e-commerce web site, identified a significant issue we have to solve in our system. The issue is that our data containing a lot of missing values on it. Therefore when we consider about the data obtained from the specific used products given by sellers, those data are suffering a kind of data scarcity issues.

By using machine learning techniques, the missing values of an attribute could be predicted, by an intuitive idea of using the given attributes that are most similar to the attribute whose values are missing. Initially, we have applied KNN algorithm, which is fundamental and simplest classification technique for predicting missing values in machine learning techniques.

We have evaluated our initial performance on KNN-based prediction by using WEKA. Table 1, shows the results of the initial performance of KNN-based prediction. Since the ratio of wrong estimation is necessary to be improved, we have modified the KNN algorithm to reduce the wrongly estimating percentage of the result. We have used, the similarity of two item’s descriptions determined using a distant function, and we are able to predict those missing values considering attributes data values.

Table 1: Results of Initial Performance of KNN Algorithm Via WEKA

	correctly estimated	wrongly estimated
Number of Instances	427	68

In Algorithm 2, we have briefly described which we have used to filling “UNKNOWN” values on this data set, that is a simplified version of k-nearest neighbour. We have implemented above mentioned Algorithm 2 as a two-loop function algorithm. Because we have considered two main possibilities of behaviors of our data set in this scenario. Line 1 to line 10 we described the first possibility and line 11 to line 18, have described the second possibility which we were considered on this scenario. As input: Can use extracted and sorted used product’s attributes values from its condition statement which describe on e-commerce web site via Algorithm 1. As output: Predicted “UNKNOWN ” attributes values for in each item’s attribute which containing value of “UNKNOWN”. Line 1, we have mentioned items to used items, which were published on e-commerce web sites. In our experiment and for case study we have used, used Apple iPhone which published on Amazon.com website. And raw headers are describing the items and number of items, as shown example in Figure 9.

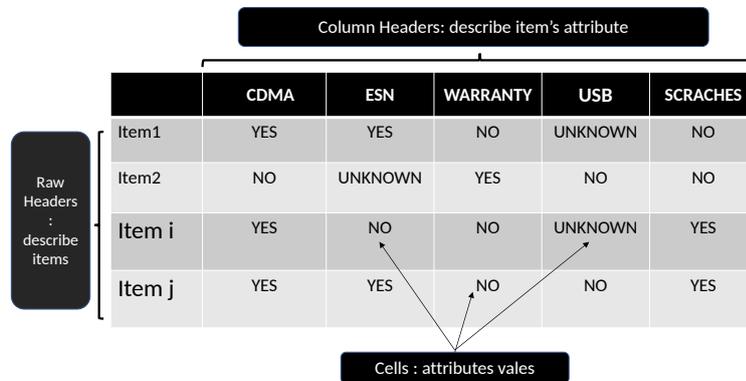


Figure 9: Structure of the Data Set, which Extracted via Algorithm 1

Line 2, In our extracted and formatted data set, if any item included the attribute value “UNKNOWN ” in it’s raw, first Algorithm 2 find the item i which containing the unknown attribute values. Line 3-4, Have mentioned attribute name “USB ” as Column header value for item i’s “UNKNOWN ” value, as shown in Figure 10. Line 5, Algorithm finding the item j which have k maximum number of same attributes value pairs with item i, shown as a example Figure 10: k = 3.

Then filling item i 's “UNKNOWN ” attribute value paring with item j’s attribute USB’s value. Shown example Figure 10:“UNKNOWN ” =NO. Then we have described other considered possibility scenario from line 11 to line 19 men-

	CDMA	ESN	WARRANTY	USB	SCRACHES
Item1	YES	YES	NO	UNKNOWN	NO
Item2	NO	UNKNOWN	YES	NO	NO
Item i	YES ↓	NO	NO ↓	UNKNOWN ↕	YES ↓
Item j	YES	YES	NO	NO	YES

Attribute pairs

Figure 10: Structure of the Attributes Value Pairs on the Data Set

tioned in Algorithm 2, if algorithm did not find any equal attribute value pairs, then filling the UNKNOWN attribute value with most frequently used value according to the attribute name.

3.4 User Interface

We have designed our system user interface which user can simply find the needs on used iPhone.

As shown on Figure 11, users can view only descriptions, which have been supplied by the sellers, about the used iPhone. If the user is interested about a significant characteristic about a used iPhone, then the user may search and see information about other phones, descriptions, which as well. By using our system, if a user seeks about some significant characteristics on the phone, the user just input some keyword on our interface, as shown in Figure 12.

If the seller provided lots of unnecessary details about a used product then using our interface user can simply find the significant characteristics when looking for phones, without reading every words on the description.

The user can see the predicted missing significant specifications which the user is looking for the product as well.

For an example scenario, Figure 13 shown the original view and a zoomed view of current existing view of products condition statement area of a e-commerce web site which we extracted from www. amazon.com web site. Then if a user want to search the attribute key word “CDMA” then user has to search on “FIND MORE” button. Then it will displayed our proposed user interface as shown in Figure 14. The user has to entered the attribute keyword “CDMA” in the appeared user interface, as shown in Figure 15. Then the result will be predicted via our proposed and implemented system and displayed as shown in Figure 17. If its a predicted value results shows as “MAYBE YES”, as shown in figure 16 or “MAYBE NO”. If it is a extracted value (which mean that attribute value already included in the existing product condition statement), result will shows as “YES” or “NO”. And also if that key word not exists or unable to predict, then the results shows as “NOT

Algorithm 2 Filling “UNKNOWN” attribute values of Used Items’s conditions statements

Require: *Extracted Used Products Attribute Values* from it’s Conditions statements

Ensure: *Predicted “UNKNOWN” attribute values* in each item

```

1: for all  $i \in \text{items}$  do
2:   Finding  $i$  which include row cell containing “UNKNOWN” attributes value
3:   if  $i$  included “UNKNOWN” value then
4:     for all unknown attribute  $a$  in  $i$  do
5:       if found item  $j$  that have  $k$  or more same attribute value pairs then
6:         Filling missing attribute value by  $j$ ’s attributes values
7:       end if
8:     end for
9:   end if
10: end for
11: for all  $i \in \text{items}$  do
12:   if  $i$  included unknown value then
13:     for all unknown attribute value pair in  $i$  do
14:       if there is the single most frequent value for the attribute then
15:         Fill the missing attribute value with the most frequently used value
16:       end if
17:     end for
18:   end if
19: end for

```

FOUND”.

When a user requests values of some attributes, over web browser, first User Request Handling Server which is implemented on Node.js, accepts the user request as POST request over the HTTP protocol. And JavaScript is used for controlling content written in the html document content on web browser (i.e., AJAX-based technique has been used). This technique has been widely used to implement richer user interfaces on a web browser (e.g., [14]).

To feed the result information to the user’s web browser, the User Request Handling Server sends a request to the Data Handling Server run on Java-VM which contains Algorithm 1 and Algorithm 2 of the extracted information from the original data provided by Amazon.com site, extract the missing attributes values in the product description data, the User Request Handling Server and the Data Handling Server has to communicate over the HTTP protocol to memorize data, extracted from Amazon.com website and make the system scalable, we designed our prototype system on above mentioned architecture.

On the system structure, the User Request Handling Server and the Data Handling Server are independently prepared and run in parallel to avoid imbalance on the workload when some heavy requests have been sent by user. Also this architecture allows the system to effectively cache some result to avoid to consume unwanted computations for duplicate or the same requests. We show the structure of the system in Figure 17. The important key strong points of this design and implementation is that, users can keep experiences on the original site by utilizing floating mode and the necessary processing has been done on background including extraction of data and estimation of missing attribute values.

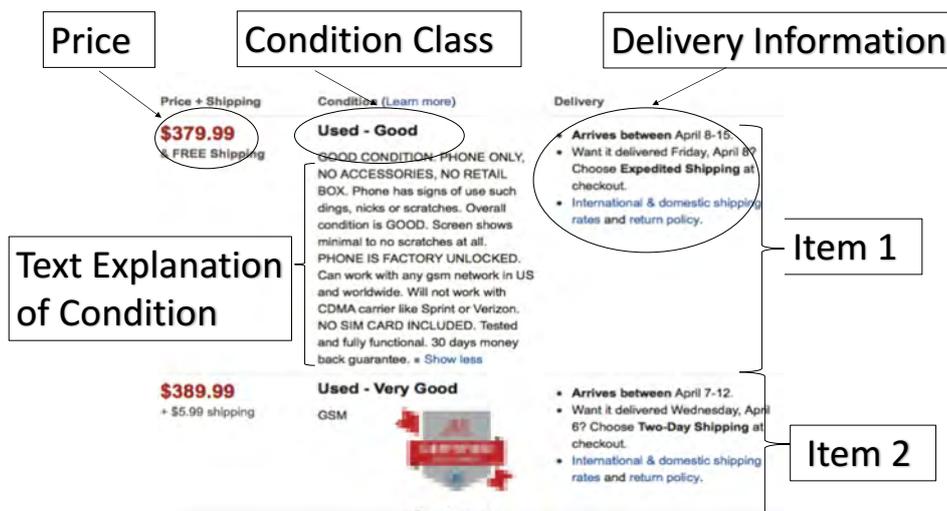


Figure 11: An Example of Original Description Given by Sellers on Amazon.com

Table 2: Accuracy on Missing Value Estimation in the Case Study

	correctly estimated	left unknown	wrongly estimated
Number of Instances	290	203	7

3.5 Preliminary Case Study

We conducted a preliminary case study on the data that have been described in section 3.1. In the case study, we have made a search with the keyword “used iPhone” and total 33 items have been found. Then, we have applied our attribute extraction Algorithm 1 to extract attribute-value pairs from the text descriptions of those items. Here, to simplify this case study, we assume all the known attribute values are boolean, i.e., represented as “Yes”, “No”, “UNKNOWN”. We have merged the extracted data for all 33 items and applied the Algorithm 2. After this process, all “UNKNOWN” attribute values have been obtained their estimated value. However, since we did not have any further information about the sold items in this case study, it was difficult for us to know whether every “UNKNOWN” values are correctly predicted. Therefore, we conduct a preliminary experiment to evaluate the accuracy of our results.

Here, we used the result obtained from the Algorithm 1 in this case study. Then, we randomly selected a known attribute value from the data and replaced that value as : “UNKNOWN ” value. Then, we applied Algorithm 2 to this modified data, and compare the result to the known value which should be the golden-truth. We repeated this process 500 times and the result is shown in TABLE 2. Here, we can see that 58 percent of known values have been correctly estimated and only 1.4 percent were incorrectly estimated by our system. When we compared the results we got more than 14% of wrongly estimated, as shown in Table 1. This shows that the number of wrongly estimated instance has been decreased from 14% to 1.4% by using our modified version of KNN Algorithm 2 in this scenario.

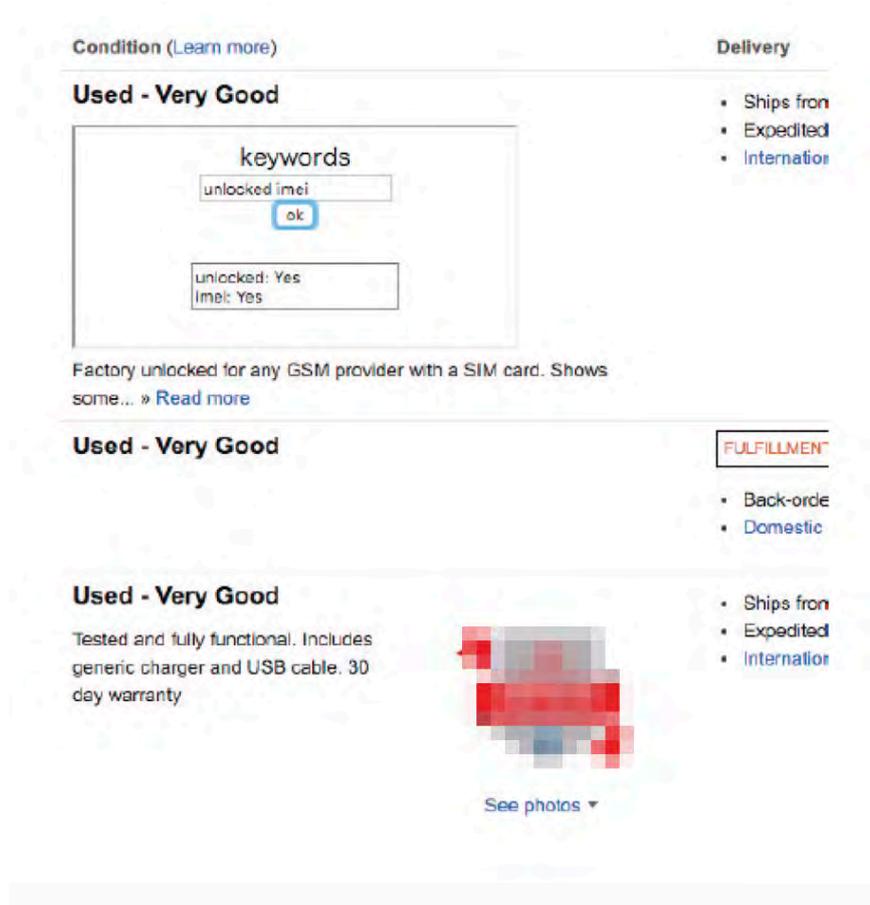


Figure 12: An Example User Interface to lookup Attribute Values on the page

Here, we cannot apply estimation on 40.6 percent of cases. To our best knowledge, this has been caused because of lack of information for that attribute. In this case study, we only extracted text descriptions for 33 items and some attributes were only applicable for only one item in them. In such case, the algorithm cannot find any hints to estimate those values and the values are left “UNKNOWN”. To confirm this issue, we have applied the algorithm 2 in a different condition. The data was prepared for used iPhone and which included total 74 items [21]. In this case, since the situations on estimating the value without any hint were less, the number of “UNKNOWN” has been dramatically decreased from 203 to 1, while the Algorithm 2 keeps its accuracy on the ratio for incorrectly estimated values. This shows that, the algorithm 2 worked well even when the number of items to be used for

Table 3: Accuracy on Missing Value Estimation in the Case Study–2nd Time

	correctly estimated	left unknown	wrongly estimated
Number of Instances	491	1	8

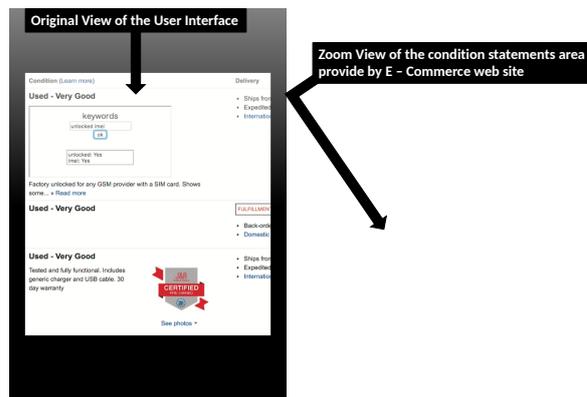


Figure 13: Zoomed View of Condition Statement Area in Amazon.com Web Site.

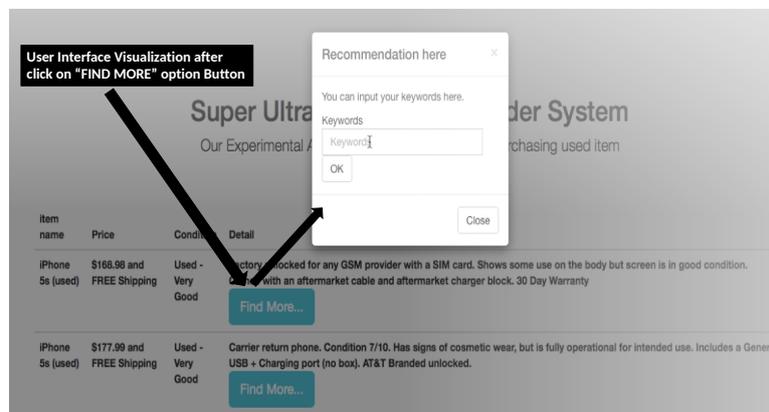


Figure 14: User Interface Displayed on the Web Page

estimation is small (e.g., less than hundred, in this case study).

4 Conclusion

In this paper, we presented an idea and the design of its prototype system that can extract or predict some attributes values of an item scarcity on a online purchasing scenario. Compared to the scenarios on brand new products, used products are suffering from a kind of data scarcity problems due to the difficulty to statically define the necessary attributes to be mentioned, when online purchasing scenarios has been applied for used products.

We proposed a prototype system and methods to fill missing attribute data on used products data even when the appropriate attributes that should be included for the items are unknown and the values were not mentioned in the given text descriptions of the specified item. Also we have implemented a floating style interface to present the extracted and predicted attribute values of the specified item for the user.

We also presented a small case study as a preliminary evaluation of the shown approach. The case study, shows that our prediction approach could be applied even when the case that there are several tens of data. We have examined the preliminary performance of predicting

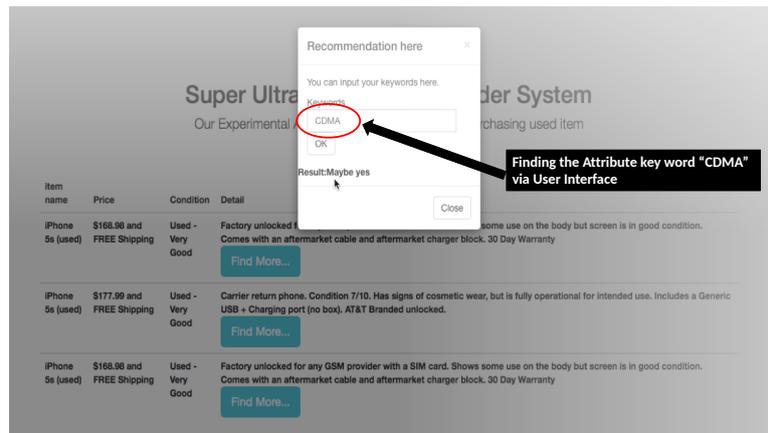


Figure 15: Search an Attribute Keyword Via User Interface

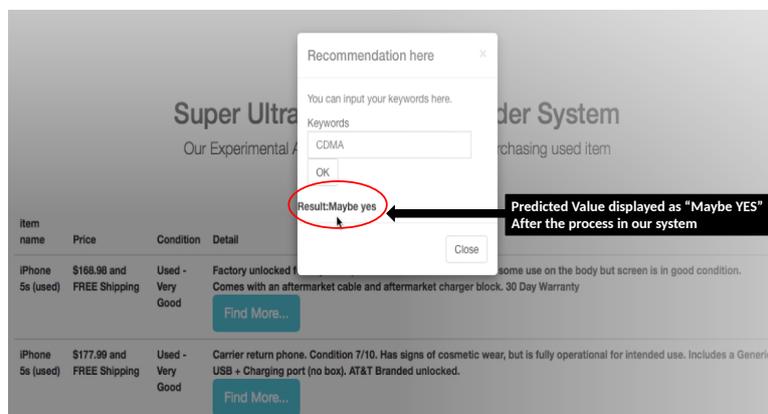


Figure 16: Extracted and Predicted Results via User Interface

missing attribute values when we have golden-truth. In the case study, there is very few case to incorrectly estimate the attribute value while there are still a number of inapplicable cases.

Although there is a possibility to extend our system to have more accurate prediction of missing attributes values, it could affect right design of implementations of user interface. Especially, implementing a better mechanism to explain how the system predicted those missing values is crucial when we deploy more accurate but complicated methods on it. We leave this as our future work.

Acknowledgment

The work was partly supported by Grants-in-Aid for Challenging Exploratory Research 26540162.

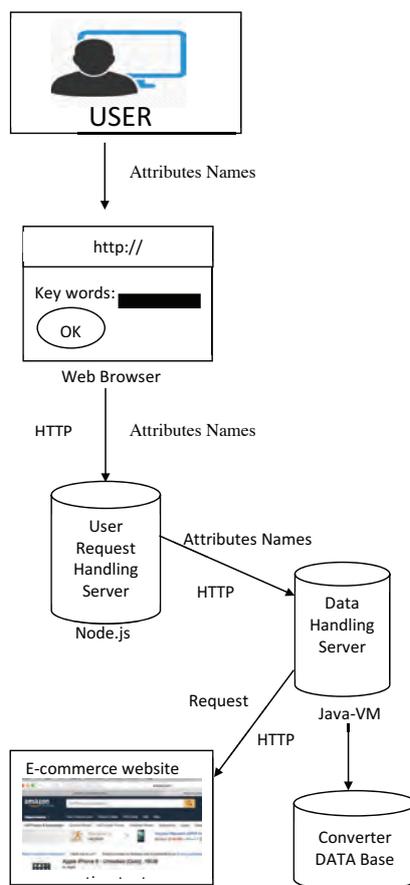


Figure 17: System Structure

References

- [1] D.H Park, H D. Kim, C.X Zhai and L Guo, *Retrieval of relevant Opinion Sentences for new products* , ACM SIGIR, pages 393–402, 2015.
- [2] Xutao.Li, Gao Cong, Xiao.Li, uan.Anh and Shonali Krishnaswamy, *Rank-GeoFM: A Ranking based Geographical Factorization Method for Point of Interest Recommendation* , ACM SIGIR, pages 433–442, 2015.
- [3] M. Ye, P. Yin, W.C. Lee, and D.L. Lee. *Exploiting geographical influence for collaborative point-of-interest recommendation*, In SIGIR, pages 325–334. ACM–2011.
- [4] C. Cheng, H. Yang, I. King, and M.R. Lyu. *Fused matrix factorization with geographical and social influence in location-based social networks*, In AAAI–2012, pages 17–23.
- [5] A. Noulas, S. Scellato, N. Lathia, and C. Mascolo. *A random walk around the city: New venue recommendation in location-based social networks*, In PASSAT, pages 144–153. IEEE–2012.

- [6] J. J. Levandoski, M. Sarwat, A. Eldawy, and M. F. Mokbel, *LARS: A location-aware recommender system*, In ICDE, pages 450–461. IEEE–2012.
- [7] H. Gao, J. Tang, X. Hu, and H. Liu, *Exploring temporal effects for location recommendation on location-based social networks*, In Recsys, pages 93–100. ACM–2013
- [8] B. Liu, Y. Fu, Z. Yao, and H. Xiong, *Learning geographical preferences for point-of-interest recommendation*, In SIGKDD, pages 1043–1051. ACM–2013
- [9] A Beykikhoshk , O Arandjelovic , D Phung and S Venkatesh, *Overcoming Data Scarcity of Twitter: Using Tweets as Bootstrap with Application to Autism-Related Topic Content Analysis* , International conference on advances in Social networks analysis and Mining : IEEE/ACM–2015.
- [10] L. Zhou and P. Chaovalit, *Ontology-supported polarity mining*. *Journal of the American Society for Information Science and technology*, pages 98–110, January 2008.
- [11] I. Bhattacharya, S. Godbole, and S. Joshi, *Structured entity identification and document categorization: two tasks with one joint model*, In Proceedings of ACM KDD–2008, pages 25–33.
- [12] J. Yu, Z. J. Zha, M. Wang, K. Wang, and T. S. Chua, *Domain-assisted product aspect hierarchy generation: towards hierarchical organization of unstructured consumer reviews*, In Proceedings of EMNLP 2011, pages 140–50, 2011.
- [13] D. H. Park, C. Zhai and L. Guo, *Speclda: Modeling product reviews and specifications to generate augmented specifications*, In Proceedings of the 2015 SIAM International Conference on Data Mining. SIAM, 2015, pages 411–418.
- [14] T. Ozono, S. Shiramatsu, and T. Shintani, *A Stable Layered Canvas Mechanism for Collaborative Web Applications*, Proc. 2015 IEEE/WIC/ACM WI2015, pp.101–106, 2015.
- [15] J. Ben Schafer, Joseph A. Konstan and John Riedl, *E-Commerce Recommendation Applications*, GroupLens Research Project–2001–Department of Computer Science and Engineering University of Minnesota Minneapolis, MN 55455 1-612-625-4002, pp.01–24.
- [16] A. Fayazi, K. Lee, J. Caverlee, and A. Squicciarini, *Uncovering Crowdsourced Manipulation of Online Reviews*, Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval–2015, pp. 233–242.
- [17] D. kotzias, M. Denil, N. de Freitas and P. Smyth, *From Group to Individual Labels Using Deep Features*, Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining–2015, pp. 597–606.
- [18] Used Products details on, http://www.amazon.com/gp/offerlisting/B00VHSXBUA/ref=dp_olp_used?ie=UTF8&condition=used.
- [19] Extracted products description data on, http://www.amazon.com/gp/offerlisting/B00VHSXBUA/ref=dp_olp_used?ie=UTF8&condition=used.

- [20] J. Ben Schafer, Joseph A. Konstan and John Riedl, *E-Commerce Recommendation Applications*, GroupLens Research Project Department of Computer Science and Engineering University of Minnesota Minneapolis, 2001-MN 55455 1-612-625-4002, pp.01-24.
- [21] Extracted Product items information on, https://www.amazon.com/gp/offer-listing/B00F3J4B5S/ref=olp_twister_child?ie=UTF8&mv_color_name=2&mv_size_name=0&qid=1500971645&sr=1-2.