Efficient Feature Selection for Product Labeling over Unstructured Data

Zeki YETGİN
Computer Engineering Department,
Mersin University
Mersin, Turkey

Abdullah ELEWI
Computer Engineering Department,
Mersin University
Mersin, Turkey

Furkan GÖZÜKARA
Computer Engineering Department,
Çukurova University
Mersin, Turkey

***Abstract*—The paper introduces a novel feature selection algorithm for labeling identical products collected from online web resources. Product labeling is important for clustering similar or same products. Products blindly crawled over the web sources, such as online sellers, have unstructured data due to having features expressed in different representations and formats. Such data result in feature vectors whose representation is unknown and non-uniform in length. Thus, product labeling, as a challenging problem, needs efficient selection of features that best describe the products. In this paper, an efficient feature selection algorithm is proposed for product labeling problem. Hierarchical clustering is used with the state of the art similarity metrics to assess the performance of the proposed algorithm. The results show that the proposed algorithm increases the performance of product labeling significantly. Furthermore, the method can be applied to any clustering algorithm that works on unstructured data.**

***Keywords*—Product labeling; product clustering; feature selection; similarity metrics; hierarchical clustering**

I. Introduction

With recent developments in web technologies, online shopping sites are changing to powerful product search engines to integrate various attractive services for sellers and customers such as product recommendation and comparison systems [1]-[3]. These web platforms can get data from online marketplaces, unify them, and provide e-commerce services using this data for their customers as in online comparison shopping engines [2]-[3].

One general concern in these search platforms is the product labeling problem for clustering identical products [4], generally referred to as record linkage [2]. Usually, customers want to compare the same products from different sellers, e.g. to see their prices. The product labeling requires assigning labels to those products that have identical features. However, one product is commonly described in different ways by different online web sources. Moreover, in order to describe the structure of the product information, each web source should have its own schema. Currently, ontology mapping approaches [5]-[6] are used to unify the product information from various resources. Ontology mapping is the schema matching approach in order for the web sources to learn their structure description of product information. However, a separate ontology should be developed manually for each web source, and importantly the ontology mapping may not provide perfect data collection. That is, the collected product data may still include unstructured or incomplete features. To address the problem of unstructured or imperfect data, new decision methods, apart from ontology matching approaches, are required in product clustering and labeling.

Formally, product labeling can be considered as a clustering problem to group the identical products into the same category using some similarity metrics. Each product is described by a feature vector and the similarity metrics define the degree of similarity between any pair of feature vectors. If these vectors contain only descriptive and relevant features that contribute much to its identification, the performance of clustering identical products will be improved significantly. However, with unstructured vectors where no vector metadata is available, selecting most descriptive and important features becomes crucial and challenging for product labeling. Thus, product labeling requires efficient feature selection methods to cope with unstructured nature of product data. In this paper, a web crawler is implemented to blindly collect products' features in many categories. Then product labeling is demonstrated using hierarchical clustering algorithms of various types with the proposed feature selection method applied.

The paper is organized as follows: Related works are investigated in the second section. The third section provides the system model and the proposed algorithm. The fourth section demonstrates the experimental results of the proposed methods. Finally, conclusion and future directions are given.

II. Related Work

Most works related to product clustering usually focus on analysis of customer behaviors [7] to cluster recommended products of interest [1] or analysis of product reviews [8]-[9] to study human opinions about the products features. These works usually use sentiment analyses or opinion mining [9]-[16] where human subjects are involved for assessment of the product features. For example, the authors in [15] cluster the similar features and try to find the correlation between the human opinions and set of features of the products. Commonly, feature selections of the products are also studied with respect to opinion mining or human behaviors. In [9], [12]-[14], information extraction systems are introduced, which extracts fine features with respect to associated opinions. However, none of the works in the literature studied feature selection for product labeling for unstructured web crawled over product dataset.

In literature, the problem of categorizing identical products, referred to here as product labeling, is expressed using different terminologies such as record linkage, entity resolution, duplicate detection [17], clustering of identical products [4], and product normalization [2], [18]. To the best of our knowledge, only few works [2], [4], [19] addressed the record linkage problem for ecommerce products. In [2] the record linkage problem is addressed by using supervised-learning of a similarity function, which is costly and not practical due to continuous need of training. Also, in [4] clustering algorithm is used to label identical ecommerce products where new similarity and performance metrics for clustering of identical products are proposed. Moreover in [19], an incremental hierarchical clustering system for record linkage in ecommerce domain is proposed. Although there are many works related to record linkage, almost none of them consider product labeling or product identification for web crawled products taking feature selection into account.

III. System Model

In this section, the proposed feature selection algorithm and its application to hierarchical clustering is described for product labeling problem. Hierarchical clustering is used, as described in [4], to solve the product labeling problem where the feature vectors are formed using the proposed feature selection algorithm. The dataset containing the product features are obtained from [4] where each line represents feature vectors, some samples are shown in Table 1. The proposed method selects important product features and removes the others that don't contribute to identification of the product, which results in final feature vectors.

TABLE. I. Samples of Initial Feature Vectors

'intel' 'core' 'i5' '2300' '80ghz' '6mb' 'vga' '1155p'
'bx80623i52300' 'intel' 'lga1155' 'core' 'i5' '2300' '80ghz' '6mb' 'cache'
'intel' 'core' 'i5' '2300' '80' 'ghz' 'lga1155' 'i?lemci'
'intel' 'ci5' '2300' '80ghz' 'mb' 'vga' '1155p' 'core' 'i5' 'i?lemci'
'intel' 'core' 'i5' '2300'

A. Proposed Feature Selection Algorithm

The proposed feature selection algorithm has 3 phases. In the first phase, it divides the feature space into overlapping clusters where same vectors might be referred to in different clusters. In the second phase, most informative features of the vectors in each cluster are selected and ordered using a weight function which adopts two criteria: 1) Vector length: The vector with smaller length carries more descriptive information than the longer one. That is, the information load per feature is high. 2) Feature position: Features early positioned in the vector are possibly more informative than the late ones. People usually tend to refer to important features at the earliest while describing products. So at the second phase, features in each vector ordered according to their weights and vector lengths are trimmed to cluster average so that features not contributing to the products identification are eliminated. In the final phase the vectors are just trimmed to target-dimension. The key property of the algorithm is that any modification to overlapped vectors can be seen by other clusters and this causes information flow among clusters achieving better identification of informative features.

In order to present the proposed algorithm, let's define the following terminology.

$VS = \{ Vi \mid i = 1..N \}$ represents feature vector space where Vi represents the feature vector of the i^{th} product among N products.

C_i represents i^{th} *cluster* which contains indices of the vectors in VS that are similar to V_i according to *similarity* metric and the *threshold* as system parameters, and formulated in (1).

$$C_i = \{ j \in 1..N \mid \ similarity(V_i, V_j) >= threshold \}$$

$$\text{where } similarity(V_i, V_j) = \frac{|V_i \cap V_j|}{\min(|V_i|, |V_j|)} \tag{1}$$

AD_i represents the average dimension of the vectors in C_i, and formulated in (2).

$$AD_i = \left\lfloor \frac{\sum_{k \in Ci} |V_k|}{|C_i|} \right\rfloor \tag{2}$$

FR(f, C) denotes the frequency of feature f in vectors of cluster C, and formulated at (3).

$$FR(f, C) = \sum_{k \in C} \begin{Bmatrix} 1, & \text{If } f \in V_k \\ 0, & \text{Otherwise} \end{Bmatrix} \tag{3}$$

AL(f, C) denotes the average length of the vectors in cluster C that includes feature f and formulated in (4) and (5). This is used for the first criterion in (7).

$$SL(f, C) = \sum_{k \in C} \begin{Bmatrix} |V_k|, & \text{If } f \in V_k \\ 0 \ , & \text{Otherwise} \end{Bmatrix} \tag{4}$$

$$AL(f, C) = \frac{SL(f, C)}{FR(f, C)} \tag{5}$$

where *SL(f, C)* denotes the summation of the vector lengths in cluster C that includes feature f.

ML(f, C) denotes the minimum length of the vectors in cluster C that includes feature f and is formulated in (6). This is used for the first criterion in (7).

$$ML(f, C) = \min \{ |V_k| \mid \ k \in C \text{ and } f \in V_k \} \tag{6}$$

WL(f, C) denotes the weight of the feature f in C according to the first criterion (vector length criterion), and is formulated in (7). Once a feature is found in a minimum length vector (*ML*), this should be highly emphasized (most informative state). The more the feature f repeats in cluster C, the more the weight approaches to its most informative value (*ML*). The less the feature f repeats in cluster C, the more the weight approaches to cluster average (*AL*).

$$WL(f,C) = ML(f,C) + \frac{AL(f,C) - ML(f,C)}{FR(f,C)^2} \quad (7)$$

AP(f, C) denotes the average position of the feature f in vectors of cluster *C*, and is formulated in (8). This is used for the second criterion at (9).

$$AP(f,C) = \frac{\sum_{k \in C} pos(f,V_k)}{FR(f,C)} \quad (8)$$

where *pos(f, V)* returns the position (index) of feature *f* in vector *V*, or return *zero* when *f* is absent in *V*.

WP(f, C) denotes the weight of the feature *f* in *C* according to the second criterion (feature position criterion), and is formulated at (9) to give a little bit more significance to the second criterion than the first criterion and also to escape from zero division when they are combined in (10).

$$WP(f,C) = \log_e^2\left(AP(f,C)\right) + 1 \quad (9)$$

Weight(f, C) denotes the overall weight of the feature *f* in *C* according to our two criteria for vector length and feature position, and is formulated in (10).

$$Weight(f,C) = \frac{1}{WP(f,C) * WL(f,C)} \quad (10)$$

The proposed algorithm sorts the vector *V* in each cluster *C* in descending order of *Weight(f, C)* where *f* scans the features of vector *V* and then attempts to reduce the dimension of *V* to cluster average if possible or to target dimension otherwise.

The proposed feature selection algorithm is shown below:

Feature Selection Algorithm

Input: *initial vector space (VS), threshold, target_dimension*

Output: *resulted vector space*

For i=1 to N
  C_i ← create cluster C_i according to (1)
For i=1 to N
  CS←extract all distinct features in vectors of C_i
  AD_i ← compute average dimension of C_i according to (2)
  If AD_i < *target_dimension* Then AD_i =*target_dimension*
  For each feature *f* in *CS*
    Weight(f, C_i)←calculate weight according to (10)
  For each *k* in C_i
    V_k ←sort V_k in descending order of *Weight(f, C_i)*
      where *f* scans features of V_k
    If $|V_k| \geq AD_i$ *Then*
      V_k ←select the first AD_i number of features of V_k
For i=1 to N
  If $|V_i| \geq$ *target_dimension Then*
    V_i ← select the first *target_dimension* number
      of features of V_i

B. Clustering Model and Performance Metrics

In this section, we describe the clustering model for product labeling on which we applied our proposed feature selection algorithm. We have considered hierarchical types of clustering to demonstrate how our feature selection algorithm achieves well in labeling identical products. Hierarchical clustering algorithms use similarity metrics and linkage metrics. The similarity metric determines the degree of similarity between any pair of vectors. This paper considers the similarity metrics recently proposed in [4] for non-uniform feature vectors. So we demonstrate the performance of our algorithm using four similarity metrics proposed in [4], namely minimally-normalized intersection similarity (MNI), globally-normalized locally weighted similarity (GNLW), globally-normalized-indexed similarity (GNI) and globally-normalized globally weighted similarity (GNGW).

Hierarchical clustering algorithm needs also linkage metrics, which use the underlying similarity metrics to measure the similarity among sub clusters to merge them to form a bigger cluster in the hierarchy. With the four similarity metrics mentioned earlier and the five following linkage metrics, different clustering algorithms are considered in the paper. These linkage metrics are single (nearest distance), complete (furthest distance), average (unweighted average distance), weighted (weighted average distance), and median (weighted center of mass distance) linkage clustering. The single method considers the smallest distance between the points in two clusters for the decision of merging whereas the complete method considers the furthest distance between two clusters. The other methods behaves similarly as their names indicate.

We used the performance measurement metrics recently proposed in [4] where three metrics, namely False-Positive (*FP*), False-Negative (*FN*), and Total Error (*TE*) are defined to assess the performance of the product labeling when the original cluster labels are available.

The metric considers a space of product pairs where the labels of pairs that are detected as identical or non-identical by the algorithm are compared with the original true labels that are priori available. The metrics are defined in [4] as follows:

1) False-Negative (FN) indicates the number of the product pairs that are classified as non-identical by the algorithm, although they are actually identical.

2) False-Positive (FP) indicates the number of the product pairs that are classified as identical by the algorithm, although they are actually non-identical.

3) Total Error (TE) indicates the total number of decision errors caused by either False-Negatives or False-Positives.

IV. Results and Discussion

A. Datasets

The dataset is acquired as two text files from [4] where one file is for product description (product-file) and the other is for error-free product labels (label-file).

Each line of the product-file describes one product without using any predefined structure (see Table 1) and the corresponding product label is given at the same line of the label-file. The dataset includes one thousand products, selected randomly from one million products, which are blindly crawled from 20 most popular online Turkish sellers (see Table 2). The products are from many different categories including computers, home appliance, smart phones, etc. So each line of the product-file serves as initial feature vector and our proposed feature selection algorithm generates final feature vectors that are input to the hierarchical clustering algorithms.

TABLE. II. SOME ONLINE SELLERS THAT ARE CRAWLED TO COLLECT PRODUCT INFORMATION [4]

Web Site	Number of Products Crawled	Number of Pages Crawled
hepsiburada.com	177.310	313.946
hizlial.com	84.046	166.197
webdenal.com	69.979	121.853
ereyon.com.tr	68.960	92.076
pratikev.com	63.170	69.275
netsiparis.com	40.525	59.294

B. Experimental Results

In this section, the results of the proposed feature selection algorithm are shown and the performance of its success in product labeling problem is demonstrated using the hierarchical clustering algorithms where four similarity metrics (MNI, GNLW, GNI, GNGW) and five linkage metrics (single, complete, average, weighted, median) are used. Exhaustive experiments are conducted and optimum values of performance metrics are provided in tables. All the algorithms in this paper are implemented using MATLAB©. The results of the proposed method are given in Tables 3 to 8 for different target dimensions where the optimum threshold is given at the table title. These tables only provide some example sets for demonstration purposes.

The results in Tables 4, 6 and 8 show that the algorithm selects and orders the informative features successfully in general. For example, the feature ‘st1500dl003’ as an informative word isn’t initially contained in top 3 features in Table 3; however our algorithm succeeds in bringing it to the second position as shown in Table 4. Similarly, in Table 5, the most descriptive features ‘kingston’, ‘16’, ‘gb’, ‘dtig3’ are successfully selected in top 4 features in Table 6. Similar success can be seen in Table 8 where the most 6 informative features are selected. These results are qualitative examples where one must further analyze the selected features.

In order to demonstrate the success of the proposed feature selection algorithm quantitatively we can compare the product labeling performance of the hierarchical clustering algorithm with and without applying the proposed feature selection. The evaluation of each clustering experiment is done by the performance metrics proposed in [4].

TABLE. III. ORIGINAL SET 1

'seagate' 'barracuda' 'green' '5tb' '5900rpm' 'sata' 'gb' 'sn' 'ncq' 'sabit' 'disk' 'st1500dl003'
'seagate' 'barracuda' 'green' 'st1500dl003' '5tb' 'sata' 'sabit' 'disk'
'seagate' '5tb' '6gb' 'barracuda' 'green' 'st1500dl003'
'seagate' 'st1500dl003' '5tb' '5900rpm' '64mb' 'sata3' '6gb' 'barracuda' 'green'

TABLE. IV. RESULTED SET 1 WITH THRESHOLD=0.74, TARGET_DIMENSION = 3

'seagate' 'st1500dl003' '5tb'
'seagate' 'st1500dl003' '5tb'
'seagate' 'st1500dl003' '5tb'
'seagate' 'st1500dl003' '5tb'

TABLE. V. ORIGINAL SET 2

'kingston' '16' 'gb' 'usb' 'memory' 'dtig3' '16gb'
'16' 'gb' 'usb' 'dtig3' 'kingston'
'kingston' 'datatraveler' 'g3' '16' 'gb' 'usb' 'bellek' 'dtig3' '16gbz'
'kingston' 'dtig3' '16gbz' '16gb' 'datatraveler' 'g3' 'usb' 'flash' 'disk'

TABLE. VI. RESULTED SET 2 WITH THRESHOLD=0.72, TARGET_DIMENSION = 4

'kingston' '16' 'gb' 'dtig3'
'kingston' '16' 'gb' 'dtig3'
'kingston' '16' 'dtig3' 'usb'
'kingston' 'dtig3' '16gb' 'usb'

TABLE. VII. ORIGINAL SET 3

'samsung' 'intel' 'atom' 'n570' '66ghz' '2gb' '320gb' '10' 'beyaz' 'netbook' 'n150' 'jp0xtr'
'samsung' 'np' 'n150' 'jp0xtr' 'beyaz' 'atom' 'n570' '2gb' '320gb' 'payla??ml?' 'vga' 'gma3150' '10' 'win' 'starter'
'samsung' 'n150' 'jp0xtr' 'atom' 'n570' '66ghz' '2gb' '320gb' '10' 'netbook' 'w7s' 'beyaz'
'samsung' '320gb' 'beyaz' 'n570' 'netbook' '10' '2gb' 'jp0xtr' 'n150'

TABLE. VIII. RESULTED SET 3 WITH THRESHOLD=0.61, TARGET_DIMENSION = 6

'samsung' 'n150' 'jp0xtr' 'n570' '2gb' '320gb'
'samsung' 'n150' 'jp0xtr' 'np' 'n570' '2gb'
'samsung' 'n150' 'jp0xtr' 'n570' '2gb' '320gb'
'samsung' 'n150' 'jp0xtr' 'n570' '2gb' '320gb'

TABLE. IX. PRODUCT LABELING PERFORMANCE WITHOUT THE PROPOSED FEATURE SELECTION (LEGACY METHOD)

Similarity	MNI			GNLW			GNI			GNGW		
Linkage	FN	FP	TE	FN	FP	TE	FN	FP	TE	FN	FP	TE
single	0.29	0.26	**0.28**	0.26	0.09	**0.19**	0.39	0.42	0.40	0.21	0.18	**0.20**
complete	0.58	0.00	0.44	0.57	0.00	0.43	0.35	0.16	0.27	0.54	0.03	0.41
average	0.48	0.11	0.35	0.43	0.20	0.35	0.23	0.20	**0.22**	0.40	0.24	0.33
weighted	0.52	0.05	0.38	0.46	0.13	0.34	0.25	0.22	0.24	0.37	0.13	0.28
median	0.54	0.16	0.43	0.40	0.40	0.40	0.55	0.13	0.43	0.35	0.36	0.35

TABLE. X. PRODUCT LABELING PERFORMANCE WITH THE PROPOSED FEATURE SELECTION WITH TARGET_DIMENSION=3

Similarity	MNI			GNLW			GNI			GNGW		
Linkage	FN	FP	TE	FN	FP	TE	FN	FP	TE	FN	FP	TE
single	0.27	0.02	**0.17**	0.23	0.03	**0.15**	0.13	0.08	**0.10**	0.12	0.10	**0.11**
complete	0.37	0.07	0.26	0.36	0.08	0.26	0.15	0.07	0.11	0.29	0.17	0.23
average	0.31	0.09	0.27	0.33	0.09	0.23	0.11	0.16	0.13	0.20	0.12	0.16
weighted	0.36	0.07	0.26	0.36	0.07	0.26	0.11	0.13	0.12	0.21	0.11	0.17
median	0.33	0.06	0.23	0.32	0.07	0.22	0.09	0.19	0.14	0.27	0.07	0.19

The results are given in Tables 9 and 10 for target dimension = 3 and summarized in Fig. 1 to 3 for target dimension of 3, 4 and 6. Tables 9 and 10 show the performance of our feature selection algorithm in product labeling is almost 50% better than the legacy approach when the best TE of the legacy (0.19) and the best of the proposed method (0.10) are compared. Furthermore, our feature selection algorithm performs better than the legacy one at all linkage and similarity metrics when best TEs of each metric, denoted in bold in Tables 9 and 10, are compared.

Generally, single linkage performs better than the other linkages. Thus, we analyzed the single linkage further for other dimensions (dimension = 4 and 6) and the results are summarized in Fig. 1 to 3. Figures show that our proposed method improves the success of the product labeling for all dimensions. The success of the product labeling with the proposed feature selection is better for smaller dimensions. That is, the proposed method successfully selects the informative features. As the dimension increases the performance of all methods gets worse due to the fact that resulting feature vectors tend towards original feature vectors. That is, the more features are selected, the more unnecessary details are taken into account. As the dimension increases the GNGW performs better than the other metrics whereas GNI gets worse. For small dimensions GNI and GNGW are preferable. Thus, GNGW provides better performance for the average dimension.

Depending on the problem domain, some linkage or similarity metrics could be preferable. A hierarchical clustering algorithm with a particular linkage and a similarity metric defines the behavior of the algorithm. For product labeling considered here the results show that single linkage is favorable. Similarly, tolerance to decision errors is also dependent on the problem domain. For instance, some problems may have tolerance to the FN errors but not to the FPs. The results show that product labeling with the proposed feature selection method has more tolerance to FPs in general. That is, FNs errors contribute more to the total decision errors than FPs. The only exception is the GNGW where the total errors are mainly caused by both FNs and FPs errors.

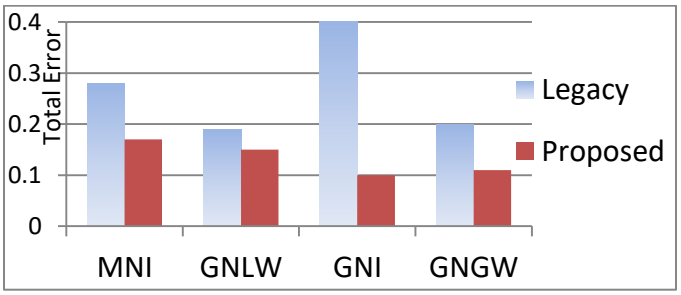

Fig. 1. Performance comparison in terms of TEs for the legacy and the proposed methods with single linkage and target_dimension = 3.

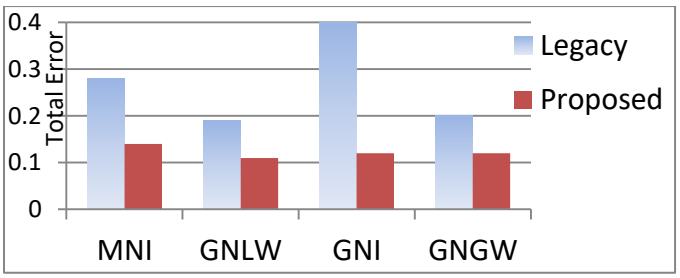

Fig. 2. Performance comparison in terms of TEs for the legacy and the proposed methods with single linkage and target_dimension = 4.

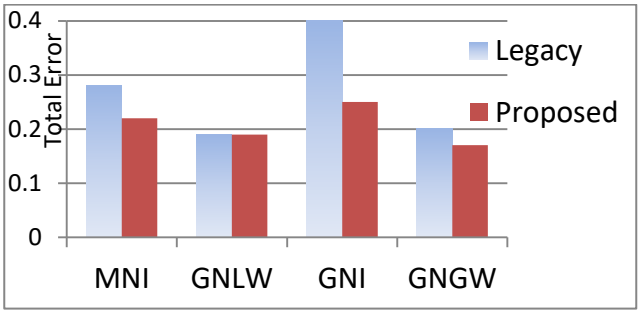

Fig. 3. Performance comparison in terms of TEs for the legacy and the proposed methods with single linkage and target_dimension = 6.

The proposed feature selection algorithm improves the performance of the product labeling by reducing the total error for all possible cases of linkage, similarity and dimension. The proposed algorithm can also be used in product clustering generally to enhance the clustering performance. However, the algorithm is tested on only available dataset due to absence of such datasets in the internet. Further study is still needed to test and improve the algorithm according to its success in new datasets.

V. CONCLUSION

A new feature selection algorithm is introduced for unstructured product data. The performance of the proposed algorithm is demonstrated by applying it into the product labeling problem where our algorithm selects most informative features before product labeling. The proposed algorithm can be used in feature selection phase of any product clustering algorithms. The performance comparison of the proposed algorithm is done by the state of the art performance metrics recently developed for the product labeling problem. The results show that the proposed algorithm provides almost 50% better performance in term of total error when compared to the legacy approach. The proposed algorithm successfully selects the brand names and major descriptive words such as category and model names. However, future works are needed to test the success of the feature selection algorithm on different datasets and improve the algorithm to cope with imperfect nature of data, such as using natural language processing, which is not addressed here.

REFERENCES

[1] L. S. Chen, F. H. Hsu, M. C. Chen, and Y. C. Hsu, “Developing recommender systems with the consideration of product profitability for sellers,” Information Sciences 2008; 178: 1032–1048.

[2] M. Bilenkoil, S. Basil, and M. Sahami. "Adaptive product normalization: Using online learning for record linkage in comparison shopping" Fifth IEEE International Conference on Data Mining, IEEE, 2005.

[3] R. B. Doorenbos, O. Etzioni, and D. S. Weld. "A scalable comparison-shopping agent for the world-wide web." Proceedings of the first international conference on Autonomous agents. ACM, 1997.

[4] Z, Yetgin, and F. Gözükara "New metrics for clustering of identical products over imperfect data." Turkish Journal of Electrical Engineering & Computer Sciences 23.4 (2015): 1195-1208.

[5] S. Park, W. Kim, S. Lee, and S. Bang, “Product matching through ontology mapping in comparison shopping”, In: Proceedings of IIWAS; 4–6 December 2006; Yogyakarta, Indonesia: ACS. pp. 39–49.

[6] M. Walther, N. Jackel, D. Schuster, and A. Schill, “Enabling product comparisons on unstructured information using ontology matching”, Advances in Intelligent and Soft Computing 2011; 86: 183–193.

[7] B. Galitsky, and J. L. Rosa, “Concept-based learning of human behavior for customer relationship management”, Information Sciences 2011; 181: 2016–2035.

[8] H. Almagrabi, A. Malibari, and J. McNaught. "A Survey of Quality Prediction of Product Reviews", International Journal of Advanced Computer Science & Applications (IJACSA) 1.6 (2015): 49-58.

[9] H. Jeong, D. Shin, and J. Choi, “Ferom: feature extraction and refinement for opinion mining”, ETRI Journal 2011; 33: 720–730.

[10] Y. Cao, P. Zhang, and A. Xiong. "Sentiment analysis based on expanded aspect and polarity-ambiguous word lexicon." International Journal of Advanced Computer Science and Applications (IJACSA) 6.2 (2015): 97-103.

[11] M. Al-Ayyoub, A. Nuseir, G.Kanaan, and R. Al-Shalabi,"Hierar-chical classifiers for multi-way sentiment analysis of arabic reviews." International Journal of Advanced Computer Science and Applications (IJACSA) 7.2 (2016): 531-539.

[12] D. Marcu, and A. Popescu, “Extracting product features and opinions from reviews”. In: Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing; October 2005; Stroudsburg, USA: ACL. pp. 339–346.

[13] Z. Shu, J. Wenjie,X. Yingju, M. Yao, and Y, Hao,“Morpheme-based product features categorization in chinese reviews mining”, In: Proceedings of the 6th International Conference on Advanced Information Management and Service; December 2010; Seoul, China: IEEE. pp. 324–329.

[14] G. Somprasertsri, and P. Lalitrojwong, “A maximum entropy model for product feature extraction in online customer reviews”. In: Proceedings of IEEE Conference on Cybernetics and Intelligent Systems; July 2008; Las Vegas, USA: IEEE. pp. 575–580.

[15] Z. Zhongwu, L. Bing, X. Hua, and J. Peifa, “Clustering product features for opinion mining”. In: Proceedings of the 4th ACM International Conference on Web Search and Data Mining; February 2011; Hong Kong, China: ACM. pp. 347–354.

[16] S. S. Sadidpour, et al. "Context-Sensitive Opinion Mining using Polarity Patterns International Journal of Advanced Computer Science and Applications (IJACSA) 7.9 (2016): 145-150.

[17] P. Christen, Data matching: concepts and techniques for record linkage, entity resolution, and duplicate detection. Springer Science & Business Media, 2012.

[18] S. Basu, M. Bilenko, and M. Sahami. "Method and system to produce and train composite similarity functions for product normalization." U.S. Patent No. 7,702,631. 20 Apr. 2010.

[19] F. Gözükara, and S. A. Özel, “An Incremental Hierarchical Clustering System for Record Linkage in Ecommerce Domain”, unpublished.