The WDC Gold Standards for Product Feature Extraction and Product Matching

Petar Petrovski, Anna Primpeli, Robert Meusel, and Christian Bizer

Data and Web Science Group, University of Mannheim, Germany
{petar,anna,robert,chris}@informatik.uni-mannheim.de

Abstract. Finding out which e-shops offer a specific product is a central challenge for building integrated product catalogs and comparison shopping portals. Determining whether two offers refer to the same product involves extracting a set of features (product attributes) from the web pages containing the offers and comparing these features using a matching function. The existing gold standards for product matching have two shortcomings: (i) they only contain offers from a small number of e-shops and thus do not properly cover the heterogeneity that is found on the Web. (ii) they only provide a small number of generic product attributes and therefore cannot be used to evaluate whether detailed product attributes have been correctly extracted from textual product descriptions. To overcome these shortcomings, we have created two public gold standards: The WDC Product Feature Extraction Gold Standard consists of over 500 product web pages originating from 32 different websites on which we have annotated all product attributes (338 distinct attributes) which appear in product titles, product descriptions, as well as tables and lists. The WDC Product Matching Gold Standard consists of over 75 000 correspondences between 150 products (mobile phones, TVs, and headphones) in a central catalog and offers for these products on the 32 web sites. To verify that the gold standards are challenging enough, we ran several baseline feature extraction and matching methods, resulting in F-score values in the range 0.39 to 0.67. In addition to the gold standards, we also provide a corpus consisting of 13 million product pages from the same websites which might be useful as background knowledge for training feature extraction and matching methods.

Key words: e-commerce, product feature extraction, product matching

1 Introduction

The Web has made it easier for organizations to reach out to their customers, eliminating barriers of geographical location, and leading to a steady growth of e-commerce sales.[1] Beside of e-shops run by individual vendors, comparison shopping portals which aggregate offers from multiple vendors play a central

[1] Retail e-commerce sales worldwide from 2014 to 2019 - http://www.statista.com/statistics/379046/worldwide-retail-e-commerce-sales/

role in e-commerce. The central challenge for building comparison shopping portals is to determine with high precision which e-shops offer a specific product. Determining whether two offers refer to the same product involves extracting a set of features (product attributes) from the web pages containing the offers and comparing these features using a matching function. The extraction of detailed product features from the HTML pages is challenging, as a single feature may appear in various surface forms in headlines, the product name, and free-text product descriptions. Product matching is difficult as most e-shops do not publish product identifiers, such as *global trade item number* (GTIN) or ISBN numbers, but heterogeneous product descriptions have different levels of detail [8].

To evaluate and compare product matching methods a comprehensive gold standard is needed. The most widely known public gold standard for product matching was introduced by Köpcke et al. [3]. However, this gold standard has two shortcomings: First, the gold standard only contains offers from four sources (Amazon.com, GoogleProducts, Abt.com and Buy.com) and thus only partly covers the heterogeneity of product descriptions on the Web. Moreover the gold standard contains only four attributes: product title, description, manufacturer and price; with more detailed product attributes (such as screen size or amount of memory) being part of free-text product titles and descriptions. These attributes need to be extracted from the free-text before they can be exploited by sophisticated matching methods. A more recent gold standard for product matching was introduced by Ristoski and Mika [14]. Their gold standard contains offers from a large number of websites which employ Microdata and schema.org markup. Their gold standard thus overcomes the shortcoming that data is gathered only from a small number of e-shops. However, their gold standard provides only two textual product attributes (name and description) and can thus not be used to evaluate feature extraction methods.

In order to overcome both above mentioned shortcomings, this paper presents two publicly accessible gold standard datasets and a product data corpus which can be used to train and evaluate product feature extraction and product matching methods:

Gold Standard for Product Feature Extraction containing over 500 annotated product web pages. On each web page, we manually annotated all product features which appear within: (i) the name of the product marked up with Microdata, (ii) description of the product marked up with Microdata, (iii) specification tables, and (iv) specification lists.

Gold Standard for Product Matching containing over 75 000 correspondences (1 500 positive, and 73 500 negative) between products from a product catalog, containing 150 different products from three different product categories, and products described on web pages.

Product Data Corpus containing over 13 million product-related web pages retrieved from the same web sites. This corpus might be useful as background knowledge for the semi-supervised training of feature extraction and matching methods.

All artefacts presented in this paper as well as the detailed results of the experiments are published as part of the WebDataCommons (WDC) project[2] and can be downloaded from the WDC product data corpus page[3].

The paper is structured as follows: Section 2 describes the selection of the websites and the products that are used for the gold standards. In Section 3 and 4 the creation of the two gold standard datasets is described and statistics about the datasets are presented. Section 5 briefly describes the corpus consisting of product web pages and the way it was crawled from the Web. The baseline approaches and their results based on the corresponding gold standard are presented in the subsequent section. The last section gives an overview of related work.

2 Product Selection

In the following, we describe the process which was applied to select the products used in the gold standards. Namely, we explain how the products from the three different product categories: *headphones*, *mobile phones* and *TVs.* were selected.

Table 3 shows the 32 most frequently visited shopping web sites, based on the ranking provided by *Alexa*[4], which we use for the product selection. We collected first the ten most popular products from the different web sites, for each of the three chosen product categories. We further complemented this list by similar products (based on their name). As example, we found the product *Apple iPhone 6 64GB* to be one of the most popular amongst all shopping web sites. We therefore included also the products *Apple iPhone 6 128GB* as well as *Apple iPhone 5* into our product catalog. Especially for the product matching task, this methodology introduces a certain level of complexity, as the product names only differ by one or two characters. All in all, for each product category we selected 50 different products.

3 Gold standard for Product Feature Extraction

This section describes the process that we used to create the gold standard for product feature extraction from product web pages. First, we explain how the gold standard was curated and then state statistical insights.

Gold Standard Curation We randomly selected 576 web pages, each containing a product description for one of the products selected in Section 2, from the product corpus detailed in Section 5. From the 576 product descriptions, 158 are belonging to the headphones category, 254 to the phones category and 164 to the TVs category.

[2] `http://webdatacommons.org`

[3] `http://webdatacommons.org/productcorpus/`

[4] `http://www.alexa.com/`

From each page we identified four key sources of attributes: As we have already shown in former research [12], information annotated using the markup language Microdata[5] has proven to be a good source of product features. Making use of the open-source library *Any23*[6], we extracted the product `name`, as well as the product `description`, marked up with Microdata with the schema.org properties `schema:name` and `schema:description` from each product web page. Further, as the research presented in [13] has shown promising results extracting features from tables and lists, we used a similar approach to identify specification lists and specification tables on the product pages.

For each extracted source, we label the contained features with an appropriate feature name. As example, if the name of the product is the string *Apple iPhone 6*, we label the sub-string *Apple* as `brand` and *iPhone 6* as `model`. Two independent annotators in parallel annotated the web pages. In case of a conflict, a third annotator solved them.

We also mapped the list of annotated product features to the list of features contained in our product catalog (see Section 4.1). This mapping as well as the gold standard dataset is available on the gold standard web page.

Distribution of Annotated Features In total, we were able to annotate 338 distinct features. Table 1 presents the frequency of properties per category for each of the labeled sources of attributes: Microdata name, Microdata description, specification table and specification list. The percent of frequency distribution is calculated from the total number of products of a product category. The table does not include a comprehensive list of the properties, but selects only those commonly occurring in each of the different annotation tasks. For the title and description we found a lot of `tagline` properties. Tagline was used for properties, which are not product specification related. As an example, when we found the title *amazing iPhone*, the sub-string *amazing* is annotated with the property `tagline`. Moreover, expected properties like `model`, `brand` and `product_type` can be seen amongst the top. For the specification table and specification list a relatively low frequency of properties, with even distribution, can be seen in the three different categories, suggesting a diversity of descriptors used by vendors.

The findings underline that features extracted from the four sources of product web pages contain valuable feature information. The identification of those features with a high precision is essential in order to perform further integration tasks, like the matching of products.

4 Gold standard for Product Matching

In this section, we describe the process which was applied to curate the gold standard for product matching. Further we present valuable statistics about the created gold standard.

[5] `https://www.w3.org/TR/microdata/`

[6] `http://any23.apache.org/`

Table 1: Feature distribution on the labeled web pages, grouped by product category and labeled source

Headphones							
Microdata name		Microdata description		spec. table		spec. list	
prop.	% Freq	prop.	% Freq	prop.	% Freq	prop.	% Freq
product_type	94.30	tagline	89.51	brand	91.97	impedance	36.23
brand	84.18	model	89.51	condition	91.08	frequency_response	34.78
model	81.01	product_type	89.51	mpn	81.26	sensitivity	33.33
tagline	65.82	brand	89.51	product_gtin	47.33	cable_length	33.33
condition	51.27	headphones_form_factor	73.72	model	41.07	package_weight	28.99
color	39.24	color	47.39	additional_features	29.47	headphones_technology	24.64
headphone_use	25.32	headphones_technology	42.12	color	29.47	weight	24.64
headphones_form_factor	24.68	additional_features	36.86	impedance	22.32	color	20.29
compatible_headphone_type	18.99	product_gtin	36.86	headphones_cup_type	19.64	connectivity_technology	17.39
compatible_headphone_brand	18.35	jack_plug	31.59	headphones_form_factor	17.86	max_input_power	17.39

Mobile Phones							
title		description		spec. table		spec. list	
prop.	% Freq	prop.	% Freq	prop.	% Freq	prop.	% Freq
brand	87.80	tagline	52.47	memory	87.30	brand	53.35
phone_type	81.50	brand	49.19	brand	86.20	product_type	45.34
tagline	78.74	phone_type	37.71	phone_type	85.09	modelnum	45.34
memory	73.62	computer_operating_system	26.24	color	79.57	compatible_phones	44.01
color	71.26	display_size	24.60	display_size	71.28	function	42.68
condition	63.78	product_type	22.96	mpn	70.73	retail_package	42.68
product_type	51.57	rear_cam_resolution	21.32	rear_cam_resolution	70.18	material	40.01
phone_carrier	36.61	compatible_phones	19.68	phone_carrier	69.62	memory	30.67
network_technology	19.29	weight	19.68	condition	67.97	package_weight	30.67
display_size	18.11	material	18.04	computer_operating_system	64.65	package_size	29.34

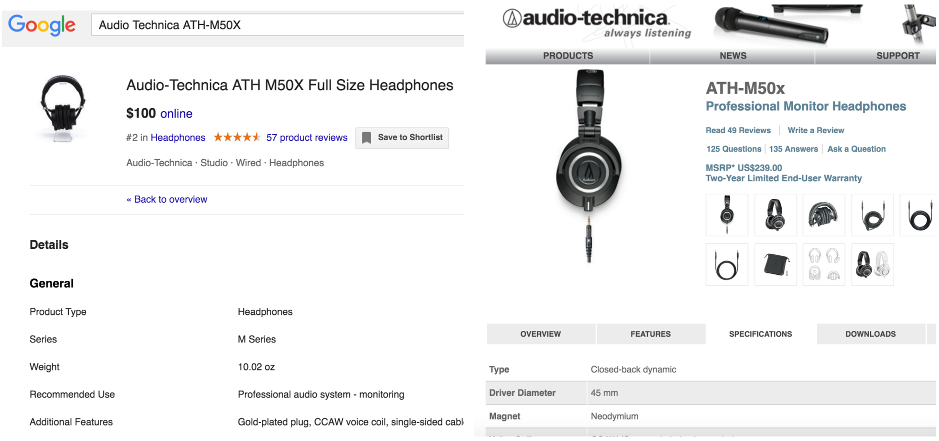
TVs							
title		description		spec. table		spec. list	
prop.	% Freq	prop.	% Freq	prop.	% Freq	prop.	% Freq
product_type	83.77	display_type	76.33	brand	74.83	viewable_size	24.14
tagline	79.87	brand	73.70	display_type	60.41	hdmi_ports	20.69
brand	79.22	tagline	55.28	mpn	59.51	3d_technology	17.24
display_type	70.13	model	44.75	viewable_size	52.29	usb	17.24
model	66.23	video_signal_standard	42.11	condition	49.59	image_aspect_ratio	17.24
viewable_size	48.70	product_type	42.11	display_resolution	36.07	computer_operating_system	17.24
total_size	45.45	smart_capable	34.22	model	34.26	depth	17.24
refresh_rate	20.13	display_resolution	23.69	product_type	29.75	hdmi	17.24
display_resolution	19.48	total_size	13.16	weight	29.75	height	17.24
compatible_tv_type	18.83	curved	10.53	color	29.75	height_with_stand	17.24

In order to generate the gold standard for product matching, we create a product catalog containing the same 150 products described in Section 2. Moreover we make use of the web pages, we crawled based on the names of the products (Section 3).

4.1 Product Catalog

To complement the products with features, for each of the 150 products in our product catalog we obtained product-specific features from the manufacturers' web site or from *Google Shopping*.

Figure 1 shows two example pages, which we used to manually extract the features for our product catalog. Figure 1a depicts a product page on *Google Shopping*. While Figure 1b depicts the manufacturers' web site for the same product.



Audio Technica ATH-M50X

Audio-Technica ATH M50X Full Size Headphones

$100 online

#2 in Headphones ★★★★☆ 57 product reviews

Save to Shortlist

Audio-Technica · Studio · Wired · Headphones

« Back to overview

Details

General

Product Type	Headphones
Series	M Series
Weight	10.02 oz
Recommended Use	Professional audio system - monitoring
Additional Features	Gold-plated plug, CCAW voice coil, single-sided cabl

a) *Features provided by Google Shopping*

ATH-M50x

Professional Monitor Headphones

Read 49 Reviews | Write a Review

125 Questions | 135 Answers | Ask a Question

MSRP* US$239.00

Two-Year Limited End-User Warranty

Type	Closed-back dynamic
Driver Diameter	45 mm
Magnet	Neodymium

b) *Features provided by the Manufacturer*

Fig. 1: Example of web pages from which we extracted data for the catalog

Availability of Product Features In total, 149 different features are identified. We found 38 for products of the category *headphones*, 33 for the category *mobile phones*, and 78 for the category *TVs*.

Table 2 shows the availability of the number identified features for the products for each of the three categories, as well as showing some examples for each identified group. We find that, especially for TVs, 40 of the features are available for at least 50% of the products. For the other products of the other two product categories, we found roughly around 20 features to be available for at least 50% of the products. A description with the complete distribution of properties can be found on our web page.

Table 2: Density of the product features for the products contained in the product catalog and example features

	Number of features				
	10 20	*30*	*40*	*50 60*	*70*
Headphones	**>50% filled**	**20-50% filled**	**<20% filled**	**N \A**	
	form_factor freq_response product_name product_type conn._technology	color magnet_mat microphone cup_type diaphragm	detach_cable foldable max_in_power height width		
Mobile Phones	**>50% filled**		**20-50% filled**	**N \A**	
	processor_type display_resolution display_size height product_name		core_count product_code manufacturer package_height modelnum		
TVs	**>50% filled**			**20-50% filled**	**<20% filled**
	product_name total_size hdmi_ports speakers_qty display_resolution			dlna timer_functions screen_modes pc_interface 3d	memory consumption response_time brightness batteries_inculded

4.2 Gold Standard Curation

We manually generated 1 500 positive correspondences, 500 for each product category. For each product of the product catalog at least one positive correspondence is included. Additionally, to make the matching task more realistic the annotators also annotate closely related products to the once in the product catalog like: phone cases, TV wall mounts or headphone cables, ear-buds, etc. Furthermore we created additional negative correspondences exploiting transitive closure. As all products in the product catalog are distinct, we can generate for all product descriptions contained in the web pages, where a positive correspondence exist to a product in the catalog, for all other products in the catalog a negative correspondence to this product on the web page.

Using the two approaches we ended up with 73 500 negative correspondences.

Distribution of Correspondences The gold standard for product matching contains 75 000 correspondences, where 1 500 are correct.

Figure 2 depicts the number of positive correspondences which are contained for each product from the three different product categories. Evidently, more than 50% of the products have two or less correspondences. While only a few of the products have between 20 and 25 correspondences.

5 Product Data Corpus

In addition to the two gold standard datasets, we have crawled several million web pages from 32 selected shopping web sites. Although we did not label all these web pages, we provide them for download as background knowledge for the semi-supervised training of feature extraction, product matching. or product categorization methods (Meusel et al. [9]).

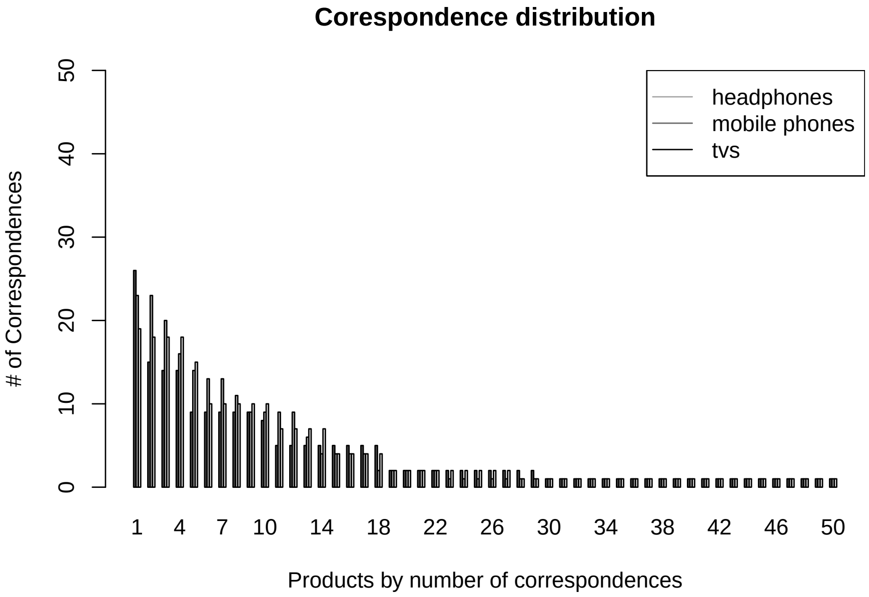


Fig. 2: Distribution of positive correspondences per category

Besides the keyword/product-based web page scraping which were already mentioned in Section 2, we performed a directed crawl for the 32 selected web sites, without the restriction to search for the specific products. We used the python-based, open source crawling framework *scrapy*[7]. We configured the framework to work in a breadth-first-search-fashion way, and restricted the crawling the web pages belonging to the 32 selected web sites. Thereby we discard all other discovered web pages.

The obtained web site-specific crawl corpus contains more than 11.2 million HTML pages. The distribution of number of pages for each of the web sites is listed in Table 3. Although for some web sites the number of gathered pages looks comprehensive, we do not claim this crawl to be complete.

Together with the product-specific crawl corpus we provide over 13 million web pages retrieved from shopping web sites. The pages are provided within WARC files and can be downloaded from our web page.

6 Baselines

In the following, we describe for each of the tasks of product feature extraction and product matching a set of straight-forward experiments and the resulting

[7] https://github.com/scrapy/scrapy

Table 3: Number of pages per web site contained in the product data corpus

Web site	# Pages	Web Site	# Pages
target.com	2,007,121	frontierpc.com	187,184
shop.com	1,754,368	abt.com	115,539
walmart.com	1,711,517	flipkart.com	63,112
selection.alibaba.com	607,410	conns.com	61,158
microcenter.com	459,921	costco.com	54,274
aliexpress.com	431,005	dhgate.com	50,099
ebay.com	413,144	shop.lenovo.com	41,465
macmall.com	401,944	bjs.com	40,930
apple.com	391,539	newegg.com	37,393
bestbuy.com	389,146	microsoftstore.com	24,163
techspot.com	386,273	samsclub.com	22,788
techforless.com	361,234	tomtop.com	13,306
overstock.com	347,846	alibaba.com	7,136
searsoutlet.com	341,924	boostmobile.com	2,487
pcrush.com	292,904	sears.com	659
tesco.com	222,802	membershipwireless.com	457

performance on the gold standard datasets described in Section 3 and Section 4. We performed these experiments in order to verify that the gold standards are challenging enough.

6.1 Product Feature Extraction Baseline

In order to create a baseline for the task of product feature extraction from web pages, we present a straight-forward approach and its results based on the feature extraction gold standard, presented in Section 3. For the evaluation we consider textual information from three different sources as input. The first source is information marked up with Microdata within the HTML page. As second source, we select product specification tables and as the third the specification lists.

Method The approach makes use of the properties which are contained for the different products in the product catalog, described in Section 4. From these property names, we generate a dictionary, which we then apply to all web pages in the gold standard. This means, whenever the name of the feature within the catalog occurs on the web page, we extract this as feature for the product.

Results We applied the dictionary method described above for the tree mentioned sources. The results for the dictionary approach vary for the different parts of the gold standard. However common for all results is underperformance of the method in general. Specifically, the method reaches results in the span of 0.400-0.600 F-score for all parts and all categories, meaning that improvement is needed. More closely, we can find that in general the method provides better recall (0.450-0.600) than precision (0.390-0.570). The reason for the poor performance can be found in the difference of the values coming from the product catalog and the different vendors. For instance, the size of a display in our catalog are inches, however some of the vendors use the metric system for that measure. Category wise, we can conclude that the *headphones* achieves the best results for all input sources, while *mobile phones* and *TVs* have comparable results.

6.2 Product Matching Baselines

In the following we present three different matching approaches and their results based on the product matching gold standard, presented in Section 4.

We use 3 distinct methodologies for feature extraction for the creation of the baselines: (i) bag-of-words (BOW), (ii) dictionary approach and (iii) text embeddings. For all the three approaches we consider textual information from three different sources as input. The first source is the HTML page itself, where we remove all HTML tags which are unrelated to the specification of the product. The second source of features are information marked up with Microdata within the HTML page. As third source, we select product specification tables and lists.

We take into account the textual information of the two input sources and preprocess the text by splitting it on non-alphanumeric characters. We convert all tokens to lower case and remove stopwords. Next, we apply a Porter Stemming Filter[8] to the remaining tokens. Finally, we take n-grams ($n \in 1, 2, 3$) of the resulting tokens.

Using the later described approaches we create vectors from the different input sources and compare them using three different similarities: string matching (sequential overlap of tokens), *Jaccard* similarity and *cosine* similarity based on TF-IDF vectors.

In the following we briefly explain each baseline method and discuss the best. Detailed results for each parameter settings can be found on our website.

Bag-of-Words The bag-of-words model is a simplifying representation where all tokens are used which are created from the preprocessing disregarding word order but keeping the multiplicity.

With this method we were able to reach 0.588, 0.412, and 0.552 F-score, for headphones, phones and TVs respectively. Generally, the precision and recall show equal performance, with the phones category being the exception. The results indicate that a purely BOW-based approach is not suitable for this task.

Dictionary Similarly, like the feature extraction baseline shown in Section 6.1 we build the dictionary for the known attributes of the product catalog. Conversely, for each known attribute we construct a list of available attribute values. Subsequently, we tag potential values from the labeled set with the attributes from our dictionary.

With this method we were able to reach 0.418, 0.614, and 0.553 F-score, for headphones, phones and TVs respectively. As with the BOW approach, precision and recall have equal performance. Noteworthy is that the results for the dictionary are comparable to the BOW approach. This can be explained by the difference in values used by various web sites (see Section 6.1).

Paragraph2Vec The most prominent neural language model for text embedding on a document level is *paragrph2vec* [5]. Paragraph2vec relies on two algorithms: Distributed Memory (DM) and Distributed Bag-of-Words (DBOW).

[8] http://nlp.stanford.edu/IR-book/html/htmledition/stemming-and-lemmatization-1.html

For the purposes of this experiment we built a DBOW model using 500 latent features. To be able to represent document embeddings paragrapah2vec maps each document to a unique paragraph vector. DBOW ignores the context words in the input and instead forms a classification task given the paragraph vector and randomly selected words from a text window sample.

Paragraph2vec is able to reach 0.675, 0.613, and 0.572 F-score, for headphones, phones and TVs respectively. As expected, text embeddings outperform both BOW and the Dictionary approach.

7 Related Work

This section gives an overview of related research in the areas of product feature extraction and product matching and discusses evaluation datasets for these tasks.

Product Data Corpora The most widely known public gold standard for product matching to this date is introduced in Köpcke et al. [3]. The evaluation datasets are based on data from Amazon-GoogleProducts and Abt-Buy[9]. However, the dataset contains only four attributes: name, description, manufacturer and price. A more recent gold standard is introduced in Ristoski and Mika [14], where the authors provide a dataset marked up in Microdata markup from several web sites. The evaluation dataset was gathered was a subset from the WebDataCommons structured dataset[10] and is gathered from several e-shops. However, the gold standard uses only two textual features: name and description. Besides, crawls from e-commerce web sites have also been published occasionally, like the one used in [6]. Unfortunately the data of such corpora mostly originates from one website, and is therefore not useful for identity resolution or data fusion. Furthermore, the data used by [9] which originated from different web site cannot be directly used for product matching as the authors did not focus on an overlap of products and therefore the usability for identity resolution is unclear.

Feature Extraction Methods One of the most prominent studies for product feature extraction is Nguyen et al. [10]. The authors introduce a pipeline for product feature extraction and schema alignment on product offers from multiple vendors in order to build a product catalog. In [13] the authors use the Bing Crawl to extract features from HTML table and list specifications and showcase their system with a product matching use case. In order to identify HTML tables and lists on product web pages, they use several consecutive classification approaches, which we also use in order to identify the location of tables and lists on the web page. Again the used dataset is not publicly available, although the authors provide (some) of their results to the public. For the purposes of this study we have reimplemented the methodology for extracting feature-value pairs

[9] `http://dbs.uni-leipzig.de/en/research/projects/object_matching/fever/benchmark_datasets_for_entity_resolution`

[10] `http://webdatacommons.org/structureddata/index.html`

used in this study and we reached 0.724 F-score for tables and 0.583 F-score for lists.

In our previous works [11, 12], we showed the usability of product-related Microdata annotations for product feature extraction. In particular the works underline that it is possible learning product-category-specific regular expressions to extract features particular from titles and descriptions of the products.

The work by Ristoski and Mika [14] uses the Yahoo product data ads to train *conditional random fields* for extracting product features from the titles as well as the descriptions product offers that were annotatated using the Microdata syntax. A similar work that employs conditional random fields for chunking product offer titles is [7].

Product Matching Methods Recent approaches by [2] match unstructured product offers retrieved from web pages to structured product specification using data found in the Microsoft Bing Product catalog. A work focusing on the exploitation of product specific identifiers, like the *manufacturer part number* (MPN) or the GTIN for product matching is presented in [4]. In [1] the authors introduce a novel approach for product matching by enriching product titles with essential missing tokens and calculate the importance score computation that takes context into account.

All those works make use of proprietary data for the task of product matching, which on the one hand side makes it hard to validate their results. On the other hand side it is also not possible to compare results of different approaches, as the heavily depend on the used data.

All of the artifacts and results from this paper are available for download at `http://webdatacommons.org/productcorpus/`.

References

1. V. Gopalakrishnan, S. P. Iyengar, A. Madaan, R. Rastogi, and S. Sengamedu. Matching product titles using web-based enrichment. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, CIKM '12, pages 605–614, New York, NY, USA, 2012. ACM.
2. A. Kannan, I. E. Givoni, R. Agrawal, and A. Fuxman. Matching unstructured product offers to structured product specifications. In *17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 404–412, 2011.
3. H. Köpcke, A. Thor, and E. Rahm. Evaluation of entity resolution approaches on real-world match problems. *Proceedings of the VLDB Endowment*, 3(1-2):484–493, 2010.
4. H. Köpcke, A. Thor, S. Thomas, and E. Rahm. Tailoring entity resolution for matching product offers. In *Proceedings of the 15th International Conference on Extending Database Technology*, pages 545–550. ACM, 2012.
5. Q. V. Le and T. Mikolov. Distributed representations of sentences and documents. *arXiv preprint arXiv:1405.4053*, 2014.

6. J. McAuley, C. Targett, Q. Shi, and A. van den Hengel. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 43–52. ACM, 2015.
7. G. Melli. Shallow semantic parsing of product offering titles (for better automatic hyperlink insertion). In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 1670–1678, New York, NY, USA, 2014. ACM.
8. R. Meusel, P. Petrovski, and C. Bizer. *The Semantic Web – ISWC 2014: 13th International Semantic Web Conference, Riva del Garda, Italy, October 19-23, 2014. Proceedings, Part I*, chapter The WebDataCommons Microdata, RDFa and Microformat Dataset Series, pages 277–292. Springer International Publishing, Cham, 2014.
9. R. Meusel, A. Primpeli, C. Meilicke, H. Paulheim, and C. Bizer. Exploiting microdata annotations to consistently categorize product offers at web scale. In *Proceedings of the 16th International Conference on Electronic Commerce and Web Technologies (EC-Web2015/T2)*, Valencia, Spain, 2015.
10. H. Nguyen, A. Fuxman, S. Paparizos, J. Freire, and R. Agrawal. Synthesizing products for online catalogs. *Proceedings of the VLDB Endowment*, 4(7):409–418, 2011.
11. P. Petrovski, V. Bryl, and C. Bizer. Integrating product data from websites offering microdata markup. In *Proceedings of the companion publication of the 23rd international conference on World wide web companion*, pages 1299–1304. International World Wide Web Conferences Steering Committee, 2014.
12. P. Petrovski, V. Bryl, and C. Bizer. Learning regular expressions for the extraction of product attributes from e-commerce microdata. 2014.
13. D. Qiu, L. Barbosa, X. L. Dong, Y. Shen, and D. Srivastava. Dexter: large-scale discovery and extraction of product specifications on the web. *Proceedings of the VLDB Endowment*, 8(13):2194–2205, 2015.
14. P. Ristoski and P. Mika. Enriching product ads with metadata from html annotations. In *Proceedings of the 13th Extended Semantic Web Conference. (To Appear)*, 2015.