PMap: Ensemble Pre-training Models for Product Matching

Natthawut Kertkeidkachorn[1] and Ryutaro Ichise[2,1]

[1] National Institute of Advanced Industrial Science and Technology,
Tokyo 135-0064, Japan
[2] National Institute of Informatics, Tokyo 101-8430, Japan
n.kertkeidkachorn@aist.go.jp, ichise@nii.ac.jp

Abstract. Mining the Web of HTML-embedded Product Data (MWPD) Challenge aims to benchmark methods dealing with two e-commerce data integration tasks: 1) Product Matching and 2) Product Classification. In this paper, we present the design of our system, namely PMap, for the MWPD Challenge on the Product Matching task. PMap aggregates the results of the various state of the art pretraining models to resolve the identical products. Results on MWHPD show that PMap outperforms the baseline and obtains the promising performance for the product matching task. The code and the system's outputs are available.[3]

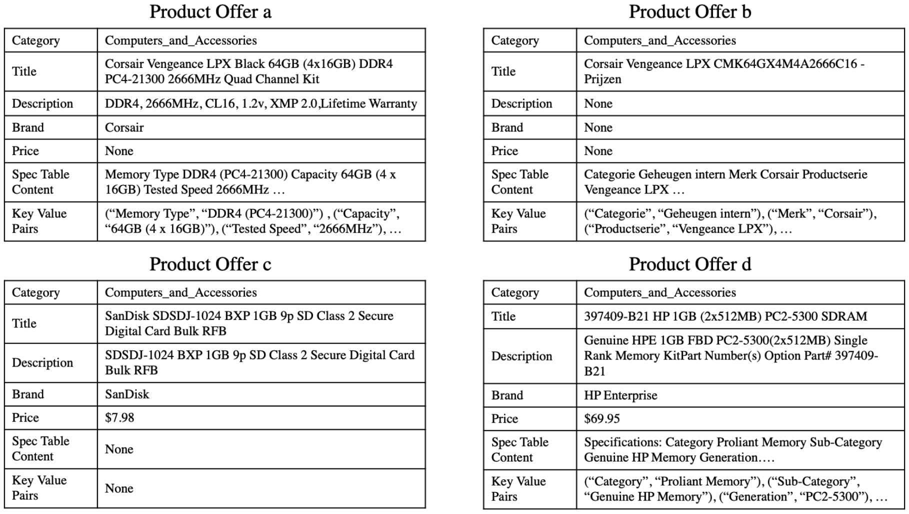
1 Introduction

Due to the growth of online shops in the e-commerce domain, semantic annotation plays a key role in enhancing the accessibility and visibility of products. Annotating the products with the semantic markup language helps a search engine to retrieve the product as a user's expectation. However, annotated products suffer from inconsistent and heterogeneous problems from cross-sector e-commerce vendors. As a result, it even leads to a situation where the product's information is conflicted. Furthermore, without a clear benchmark, it is hard to judge the progress of the methods in this field. To address these challenges, Mining the Web of HTML-embedded Product Data (MWPD) challenge[4] is introduced. The goal of the MWPD challenge is to provide the benchmark for the methods dealing with two fundamental tasks in e-commerce data integration: 1) Product Matching and 2) Product Classification.

In this study, we focus on the Product Matching task. Product Matching is to match the same products from different websites that refer to the same real-world product. To deal with the Product Matching task, we introduce the ensemble pre-train models, namely PMap. PMap takes the advantages of contextualized embedding pre-train models together with the aggregating strategy in order to uncover the identical products.

[3] http://github.com/knatthawut/mwpd

[4] https://ir-ischool-uos.github.io/mwpd/



Product Offer a

Category	Computers_and_Accessories
Title	Corsair Vengeance LPX Black 64GB (4x16GB) DDR4 PC4-21300 2666MHz Quad Channel Kit
Description	DDR4, 2666MHz, CL16, 1.2v, XMP 2.0,Lifetime Warranty
Brand	Corsair
Price	None
Spec Table Content	Memory Type DDR4 (PC4-21300) Capacity 64GB (4 x 16GB) Tested Speed 2666MHz …
Key Value Pairs	(“Memory Type”, “DDR4 (PC4-21300)”) , (“Capacity”, “64GB (4 x 16GB)”), (“Tested Speed”, “2666MHz”), …

Product Offer b

Category	Computers_and_Accessories
Title	Corsair Vengeance LPX CMK64GX4M4A2666C16 - Prijzen
Description	None
Brand	None
Price	None
Spec Table Content	Categorie Geheugen intern Merk Corsair Productserie Vengeance LPX …
Key Value Pairs	(“Categorie”, “Geheugen intern”), (“Merk”, “Corsair”), (“Productserie”, “Vengeance LPX”), …

Product Offer c

Category	Computers_and_Accessories
Title	SanDisk SDSDJ-1024 BXP 1GB 9p SD Class 2 Secure Digital Card Bulk RFB
Description	SDSDJ-1024 BXP 1GB 9p SD Class 2 Secure Digital Card Bulk RFB
Brand	SanDisk
Price	$7.98
Spec Table Content	None
Key Value Pairs	None

Product Offer d

Category	Computers_and_Accessories
Title	397409-B21 HP 1GB (2x512MB) PC2-5300 SDRAM
Description	Genuine HPE 1GB FBD PC2-5300(2x512MB) Single Rank Memory KitPart Number(s) Option Part# 397409-B21
Brand	HP Enterprise
Price	$69.95
Spec Table Content	Specifications: Category Proliant Memory Sub-Category Genuine HP Memory Generation.…
Key Value Pairs	(“Category”, “Proliant Memory”), (“Sub-Category”, “Genuine HP Memory”), (“Generation”, “PC2-5300”), …

Fig. 1. The samples of the product offers from the MWPD challenge on the product matching tasks [1].

The rest of the paper is organized as follows. We describe the problem setting of the product matching on the MWPD challenge in Section 2. Section 3 reports the design of our approach. In Section 4, the experimental setup details and the experimental results are presented. We then survey the related work in Section 5. In Section 6, we conclude our work.

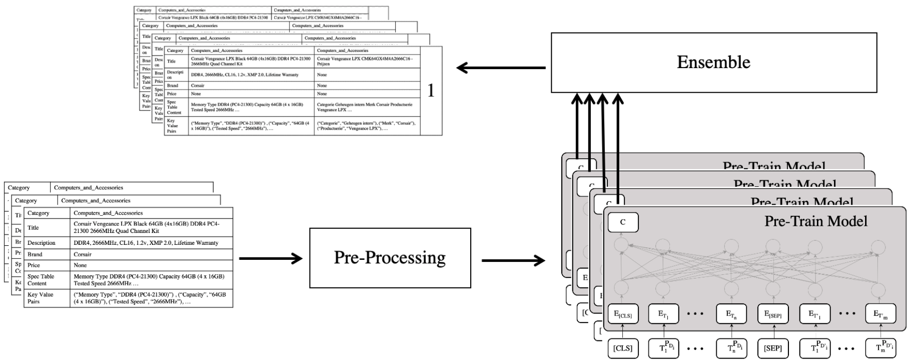
2 Problem Setting

A product offer is a collection of textual attributes that describes the real-world product. Generally, product offers are published as the product descriptions with specification tables, i.e. HTML tables that describe specifications about the offer such as price or brand of the product. The samples of the product offers are presented in Figure 1.

Product Matching in the MWPD challenge is the task to classify whether the given two product offers are identical, i.e. two product offers refer to the same real-world object. We can formulate the Product Matching problem as follows:

Let D and D' be two collections of product offers from different resources. We assume that product offers in D and D' have the same schema, i.e. a product offer is described by the same set of attributes A. Given $D = \{P_{D_1}$, P_{D_2}, P_{D_3}, ..., $P_{D_n}\}$ and $D' = \{P_{D'_1}$, $P_{D'_2}$, $P_{D'_3}$, ..., $P_{D'_n}\}$, where P_{D_i} is the $i-th$ product offer of D and $P_{D'_i}$ is the $i-th$ product offer of D', the objective of the product matching is to model the function $f : (P_{D_i}, P_{D'_i}) \rightarrow \{0, 1\}$. If two products refer to the same object, the function $f(\cdot)$ returns 1, otherwise 0.

For example, in Figure 1, the product offer a and the product offer c are from D and the product offer b and the product offer d are from D'. The pairs



Category	Computers_and_Accessories	Computers_and_Accessories
Title	Corsair Vengeance LPX Black 64GB (4x16GB) DDR4 PC4-21300 2666MHz Quad Channel Kit	Corsair Vengeance LPX CMK64GX4M4A2666C16 - Prijzen
Description	DDR4, 2666MHz, CL16, 1.2v, XMP 2.0, Lifetime Warranty	None
Brand	Corsair	None
Price	None	None
Spec Table Content	Memory Type DDR4 (PC4-21300) Capacity 64GB (4 x 16GB) Tested Speed 2666MHz …	Categorie Geheugen intern Merk Corsair Productserie Vengeance LPX …
Key Value Pairs	("Memory Type", "DDR4 (PC4-21300)") , ("Capacity", "64GB (4 x 16GB)"), ("Tested Speed", "2666MHz"), …	("Categorie", "Geheugen intern"), ("Merk", "Corsair"), ("Productserie", "Vengeance LPX"), …

Category	Computers_and_Accessories
Title	Corsair Vengeance LPX Black 64GB (4x16GB) DDR4 PC4-21300 2666MHz Quad Channel Kit
Description	DDR4, 2666MHz, CL16, 1.2v, XMP 2.0, Lifetime Warranty
Brand	Corsair
Price	None
Spec Table Content	Memory Type DDR4 (PC4-21300) Capacity 64GB (4 x 16GB) Tested Speed 2666MHz …
Key Value Pairs	("Memory Type", "DDR4 (PC4-21300)") , ("Capacity", "64GB (4 x 16GB)"), ("Tested Speed", "2666MHz"), …

Fig. 2. The design pipeline of PMap

of product offers (a, b) and (c, d) are given. The pair of product offers (a, b) is the match pair ($f : (a, b) \rightarrow 1$), while the pair of product offers (c, d) is the non-match pair ($f : (c, d) \rightarrow 0$).

3 Approach

We design our system (PMap) as the 3-steps pipeline. As shown in Figure 2, our pipeline consists of 1) Pre-processing, 2) Fine-tuning Pre-train Models, and 3) Ensemble Models. The details of each step are as follows.

3.1 Pre-processing

In the MWPD challenge, WDC Product Data Corpus[5] is used as the dataset. It is derived from the Web Data Commons[6] extracted by using schema.org annotations from the Common Crawl[7]. Although some cleaning pre-processing steps are taken into account on the dataset [6], we found that it is still necessary to further pre-process the dataset due to the character encoding and symbol in the data. To pre-process the dataset, we remove symbols and non-alphabet characters by using a simple regular expression.

3.2 Fine-tuning Pre-train Models

Fine-tuning Pre-train Models is the core step of PMap. In this section, we explain the pre-train models and how to fine-tune them.

[5] http://webdatacommons.org/largescaleproductcorpus/v2/index.html
[6] http://webdatacommons.org/structureddata/
[7] https://commoncrawl.org

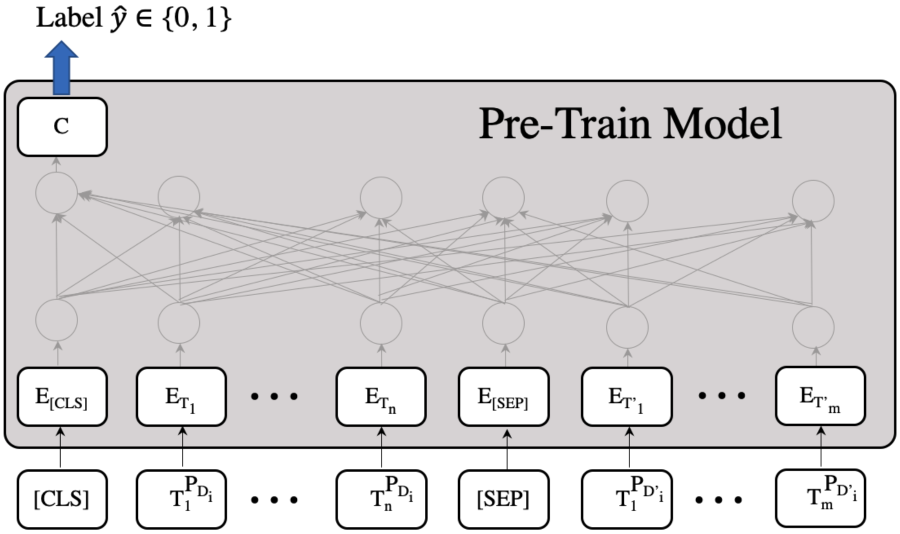


Fig. 3. Illustration of the fine-tuning pre-train models for the product matching task.

Pre-train Models, also known pre-trained language representation models, widely gain attention in the NLP community due to their transfer learning ability. Such pre-train models can easily achieve state-of-the-art performances for various NLP standard tasks [8] by simply fine-tuning the models over specific tasks. One of the state-of-the-art pre-train contextual language representation models is BERT [2]. It builds upon a multi-layer bidirectional Transformer encoder, which is based on the self-attention mechanism. During the pre-training representation learning, BERT is trained on large-scale unlabeled general domain corpus from BooksCorpus and English Wikipedia in order to perform the masked language task and the next sentence prediction task. Based on the success of the BERT, various pre-train models have also been introduced such as DistilBert[7] and Roberta[5]. We can build various models for product matching by fine-tuning pre-train models.

Fine-tuning is to optimize the model for the specific task. The architecture for fine-tuning pre-train models for the product matching task is shown in Figure 3. Given the input pair $(P_{D_i}, P_{D'_i})$, the first token of every sequence of input pairs is always a special classification token [CLS]. Following [CLS], the product offer P_{Di_i} is represented as the sequence of tokens containing the title of the product offer $P_{D_i} = \mathrm{T}_1^{P_{D_i}}, \mathrm{T}_2^{P_{D_i}}, \mathrm{T}_3^{P_{D_i}}, ... , \mathrm{T}_n^{P_{D_i}}$, where n is the length for $P_{D'_i}$ of titles after tokenized. Then, [SEP] is put after the sequence representation of P_{D_i}. After [SEP], the product offer $P_{D'_i}$ is represented by the similar way of the product offer P_{D_i} as the sequence of tokens containing the title of the product offer $P_{D'_i} = \mathrm{T}_1^{P_{D'_i}}, \mathrm{T}_2^{P_{D'_i}}, \mathrm{T}_3^{P_{D'_i}}, ... , \mathrm{T}_m^{P_{D'_i}}$, where m is the length of titles for

$P_{D'_i}$. Note that, at first, we aim to treat the product offer as the documents and use the whole details of the product offer as the sequence of tokens. However, the pre-train model allows the sequence of the tokens with the maximum length at 512. To fit the pre-train model within this limitation, we decide to use the only title as a representation of the product offer. As a result, it is still room to investigate the other attributes of product offers as features.

After feeding the input sequences to the pre-train model, the final vector representation C corresponding to [CLS] is used as the representation of the input sequence to pass to the shallow neural network for building the classifier. We compute a cross-entropy loss with the following equations to train the classifier

$$\hat{y} = \sigma(CW^T) \tag{1}$$

$$\mathcal{L} = \sum_{(P_{Di_i}, P_{D'_i})} y \cdot \log(\hat{y}_0) + (1 - y) \cdot \log(\hat{y}_1) \tag{2}$$

, where $\sigma(\cdot)$ is the sigmoid function, W is the classification layer weight of the shallow neural network for fine-tuning ($W \in \mathbb{R}^{2 \times |C|}$), $\hat{y}$ is a 2-dimensional real vector with $\hat{y}_0, \hat{y}_1 \in [0,1]$, $\hat{y}_0 + \hat{y}_1 = 1$ and y is the label for the pair of input ($y \in 0, 1$).

3.3 Ensemble Models

Based on the preliminary results on the validation dataset, we found most of the pre-train models achieved very remarkable performance. However, when we observed and analyzed the result on each sample in the training process, it turned out that each pre-train model could capture different aspects of the data. For example, we found that RoBERTa could capture the typo error, whereas others could not. Due to this signal, PMap combines the results from various pre-train modes to capture various types of aspects of the dataset and make the final prediction with these results.

4 Experiments and Results

In this section, we report the experiments of PMap on the product matching task of the MWPD challenge.

4.1 Experimental Setup

The experimental setup is as follows:

Datasets. The Product Matching dataset is derived from the WDC Product Data Corpus and Gold Standard for Large-Scale Product Matching. The product data corpus contains 16M product offers. In the product matching task, there are 68,461, 1,100, and 1,500 offer pairs for training, validating, and testing respectively.

Table 1. The Result on the Product Matching Task

System	Precision	Recall	F1 (positive pairs only)
Baseline [6]	0.7089	0.7467	0.7273
distilbert-base-uncased*	0.7810	0.7495	0.7649
bert-base-uncased*	0.7848	0.8340	0.8086
bert-large-uncased*	0.7943	0.8493	0.8209
roberta-base*	0.8210	0.8725	0.8459
roberta-large*	**0.8476**	0.8691	0.8582
PMap	0.8204	**0.9048**	**0.8605**

Settings. We select various pre-train models including distilbert-base-uncased, bert-base-uncased, bert-large-uncased, roberta-base, and roberta-large. The pre-train models are available at the huggingface repository[8]. To implement the model as in Figure 3, we employ the implementation of AutoModelForSequenceClassification[9]. We set the hyper-parameters in the fine-tuning process as follows: batch: 8, 16 or 32 (depending on the largest batch that can be loaded to the memory), learning rate: $2e-5$, epochs: 2-4, dropout rate: 0.1. The maximum length of tokens is set at 150 due to the length of the titles in the dataset. During the testing, we select bert-large-uncased, roberta-large, and roberta-base for the ensembling of the results in the pipeline. This selection is based on the observation of the validation dataset.

Baseline. In the product matching task, deepmatcher [6], a state-of-the-art matching method is used as the baseline. Also, we additionally conduct the experiment on each pre-train model for the ablation study of our approach.

Evaluation Metrics. Precision, Recall and F1 score on the positive class (class 1) is calculated.

4.2 Results

Table 1 reports the results of PMap for the product matching task. The best precision is obtained from the Roberta-large model, while PMap gives the best recall. Overall, PMap outperforms the baseline in F1 score and obtains the promising performance for the product matching task.

5 Related Work

Product Matching is a special case of the entity linking, which considers the disambiguation of a real-world entity in the e-commerce domain. There are many

[8] https://huggingface.co/models

[9] https://huggingface.co/transformers/model_doc/auto.html

* We additionally evaluate these results after releasing of the ground truth for the test dataset.

research works related to entity linking [3, 4, 6]. Early works focused on modeling the approaches with rule-based and statistics-based methods [3]. Later, the machine learning-based approach has become a popular approach due to its strong performance [4]. In recent years, the deep learning-based approach is extremely successful in many application domains. Deepmacther[6], one of the deep learning approaches, models the deep neural network and achieves the state of the art for the product matching task. However, we notice that the pre-train models have not been gained much attention in the product matching task yet. The pre-train models (e.g. BERT [2]) achieve remarkable results on many NLP tasks. Therefore, it is worthwhile to explore the pre-train models for the product matching task.

6 Conclusion

In this paper, we report the product matching system, namely PMap. PMap takes the advantages of the pre-train models to build the classifiers and then ensemble the result to make the final prediction. By fine-tuning the pre-train model on the language representation model. we could achieve a better result than the baseline. In the future, we plan to investigate the other details of the product such as description, price, etc. that are left unprocessed and not used in the current system. Also, we plan to validate the results on the various pre-train models because a new model comes out continuously.

References

1. Mining the Web of HTML-embedded Product Data. https://ir-ischool-uos.github.io/mwpd/, accessed: 2020-08-30
2. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of NAACL-HLT. pp. 4171–4186 (2019)
3. Fellegi, I.P., Sunter, A.B.: A theory for record linkage. Journal of the American Statistical Association **64**(328), 1183–1210 (1969)
4. Köpcke, H., Thor, A., Rahm, E.: Evaluation of entity resolution approaches on real-world match problems. Proceedings of the VLDB **3**(1-2), 484–493 (2010)
5. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692 (2019)
6. Mudgal, S., Li, H., Rekatsinas, T., Doan, A., Park, Y., Krishnan, G., Deep, R., Arcaute, E., Raghavendra, V.: Deep learning for entity matching: A design space exploration. In: Proceedings of the 2018 SIGMOD. pp. 19–34 (2018)
7. Sanh, V., Debut, L., Chaumond, J., Wolf, T.: Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. arXiv preprint arXiv:1910.01108 (2019)
8. Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., Bowman, S.: GLUE: A multi-task benchmark and analysis platform for natural language understanding. In: Proceedings of the EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP. pp. 353–355 (2018)