E-fashion Product Discovery via Deep Text Parsing

Uma Sawant
IIT Bombay
Bangalore
uma.sawant@gmail.com

Vijay Gabale
Huew
Bangalore
vijay.gabale@gmail.com

Anand Subramanian
Huew
Bangalore
anandps@me.com

ABSTRACT

Transforming unstructured text into structured form is important for fashion e-commerce platforms that ingest tens of thousands of fashion products every day. While most of the e-commerce product extraction research focuses on extracting a single product from the product title using known keywords, little attention has been paid to *discovering* potentially *multiple* products present in the listing along with their respective relevant attributes, and leveraging the entire title and description text for this purpose. We fill this gap and propose a novel composition of sequence labeling and multi-task learning as an end-to-end trainable deep neural architecture. We systematically evaluate our approach on one of the largest tagged datasets in fashion e-commerce consisting of 25K listings labeled at word-level. Given 23 labels, we discover label-values with F1 score of 92.2%. When applied to 2M listings, we discovered 2.6M fashion items and 9.5M attribute values.

Keywords

Product discovery, Sequence labeling, Multi-task learning

1. PRODUCT DISCOVERY FROM TEXT

Owing to the distributed and unorganized nature of seller onboarding, a large number of product listings on fashion e-commerce sites contain either unstructured text with a title and multi-sentence description, or semi-structured text specifying diverse key-value specifications, a title and possibly a short description. While curating 25K fashion listings from leading Indian fashion marketplaces, we found that more than 35% listings contain information about multiple items along with their respective attributes. For example, *titile–Arcilia frill tops and blouses, description–comes in blue and off white pattern, combine this top with cropped blue jeans and black wedges.* In the absence of an accurate mechanism to comprehensively parse a possibly multi-product listing, many search responses ignore relevant fashion products or fetch irrelevant products. Note that we not only need to identify key items and attributes from the text, but also associate attributes to appropriate items to form meaningful products. For the above example, we wish to discover the following products (1) *label:apparel-item value:top, label:color values blue, off-white*, (2) *label:apparel-item value:jean, label:color value blue, label:style value cropped*, (3) *label:footwear-item wedges,label:color value black.*[1] This challenge is compounded by the occurrence of new keywords (e.g., patterns or brands) unknown to the existing database, necessitating *discovery* of products using partial schema. Moreover, 'Blue' or 'Bodycon' are a few examples of brand names that also share color or shape values. While curating 25K listings, we found that *more than 30%* of values can have at least two labels. Most of the e-commerce-focused prior works [4, 3] consider only title text, may not work on free text descriptions, and do not attempt product discovery especially considering multiple products. On the other hand, most of the prior work on traditional sequence labeling or NER on free text [1, 2, 5] is designed and evaluated on popular POS and NER datasets that have well-formed grammatical english, unlike e-commerce listings.

WWW 2017 Companion, April 3–7, 2017, Perth, Australia.
ACM 978-1-4503-4914-7/17/04.
http://dx.doi.org/10.1145/3041021.3054263

2. NEURAL ARCHITECTURE

Our problem can be formulated as a sequence labeling problem with two distinct classes of labels : **keyword labels** (e.g. red[color] dress[apparel] black[color] shoes[footwear] are[-]) and attribute-product **affiliation labels** (red[P1] dress[P1] black[P2] shoes[P2] are[-]) where P1 is dress and P2 is shoes. Formally, if $\mathcal{X} = \{x_1, \ldots, x_T\}$ is an input text sequence, corresponding keyword labels and attribute-product affiliation labels can be specified as $\mathcal{Y}_a = \{y_1^a, \ldots, y_T^a \mid y_i^a \in \mathcal{L}_a\}$ and $\mathcal{Y}_c = \{y_1^c, \ldots, y_T^c \mid y_i^c \in \mathcal{L}_c\}$ where $\mathcal{L}_a$ and $\mathcal{L}_c$ are the sets of all possible keyword labels and attribute-product affiliation labels resp. Given $\mathcal{X}$, the task of product discovery is to predict $\mathcal{Y}^* = \operatorname{argmax}_{\mathcal{Y}_a, \mathcal{Y}_c}\ p(\mathcal{Y}_a, \mathcal{Y}_c | \mathcal{X})$

We present a deep learning framework designed to capture the nature of fashion product listings as well as label dependencies. For instance, if `black` is marked as color in `[BLUE releases a range of black dresses]`, then the chance of BLUE being tagged as a brand instead of color increases significantly. Knowing there exist multiple color values in the given text increases the chance that those colors are affiliated to different products. Furthermore, fashion grammar and language such as "formal look", 'complements', "bold

[1]Along with the fact that the first product is the dominant product, to be indexed by search engines appropriately.

and bright colors" can influence the probability distribution of products and attributes across the entire listing text.

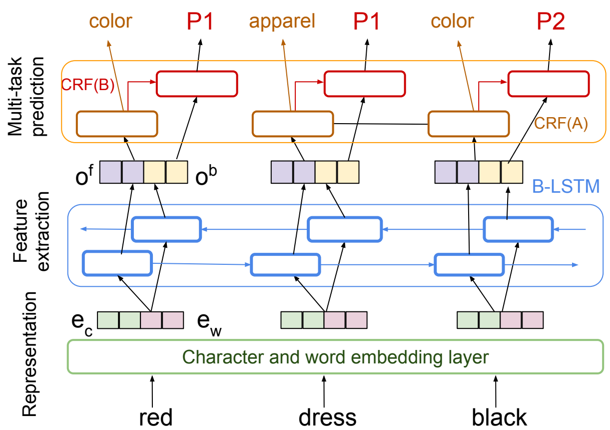

Figure 1: System Architecture.

Figure 1 shows our end-to-end architecture partially inspired by the work in [2]. To capture capitalization, phrasing, function words of the input text (wherever present), we carefully form the representation layer using: (a) a word embedding e_w from pre-trained embedding table and (b) a character-derived word representation e_c using [2]. The next layer is a bi-directional LSTM used for extracting features and long-range word dependencies. Finally, given the dependency of our two set of labels, we propose joint training of our two tasks through a **multi-task framework**. To capture the conditional dependency of affiliation labels on keyword labels, we supply the output distribution of attribute prediction model as additional features to the input of product-attribute assignment prediction model. Thus, the final layer performs the prediction of keyword labels and product-attribute affiliation labels using *cascaded CRFs*.[2]

3. EVALUATION

We consider 24990 fashion listings that are tagged at word level by two fashion curators. We use 12.7K listings for training, 4.9K for validation and 7.2K for testing. The datasets contain 1276K, 293K and 563K tokens and 8.3K, 1.7K and 3.7K vocabulary respectively. We have total 23 labels and 5311 unique label-values. 1563 out of 5311 values have at least one conflicting label and about 110K occurrences of label-values. About 20% of the label values for each of the labels are never seen by the trained models. More than 63% of listings have at least 4 attribute values associated with at least one fashion item. The median number of sentences for a listing is 11.

In our neural architecture, we consider CNN window size of 4 with 28 filters, and LSTM state size of 240 with dropout rate of 0.5. The other learning parameters are set as per [2]. Note that we do not preprocess data and do not hand-craft any features. We compare our approach with prior work in [5] (prior-A) that proposes RNNs for multi-task learning with two separate CRFs for two different tasks, and another prior work in [1] (prior-B) that proposes a word2vec and CRF based approach to tag attributes in titles. We choose the prescribed hyper-parameter values for both the prior works. We pre-train neural networks for all approaches on 1.5M listings for the task of predicting next word in the text. We evaluate all the approaches using four metrics: weighted precision, weighted recall, weighted F1 and weighted accuracy. We explore three different tasks: brand-label-disambiguation (restricted to brands as they are the most ambiguous keyword labels), keyword label prediction only on out-of-vocabulary labels (20% of all keyword labels) and product discovery, which includes both keyword label prediction and attribute-product affiliation label prediction. Results in Table 1 demonstrate that our cascaded-CRF based deep network is able to capture label dependencies and context across multiple sentences, and it outperforms disjoint-CRF based prior work in semantic disambiguation. Our model maintains its performance on out-of-vocabulary labels and is able to disambiguate brand names that often have ambiguity with color, shape or item names.

[2]Cascaded CRFs approach performed significantly better in our experiments as compared two disjoint CRFs as well as a single CRF in the joint label space.

criteria	approach	(acc, f1)	(prec, recall)
product discovery	prior-A	(81%, 80.84%)	(81.4%, 80.3%)
	prior-B	(75.4%, 74.84%)	(73.4%, 74.2%)
	our-work	(92.1%, 92.2%)	(92.7%, 91.8%)
out-of-vocab prediction	prior-A	(75.9%, 76.59%)	(75.9%, 77.3%)
	our-work	(92.2%, 92.29%)	(92.5%, 92.1%)
brand-label disambiguate	prior-A	(71.2%, 70.59%)	(70.9%, 70.3%)
	our-work	(85.2%, 85.29%)	(85.5%, 85.1%)

Table 1: Comparison of multi-task neural architectures.

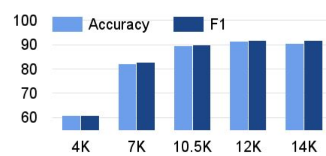

Figure 2: Sufficiency of labeled listings.

Furthermore, once the network is pretrained on 1.5M listings, it starts discovering products with high accuracy by utilizing only an order of 10K labeled listings (Fig 2). We also crawled additional 2M fashion e-commerce listings to evaluate our approach in the wild. Our trained model discovered and tagged more than 2.6M fashion products with more than 9.5M fashion attributes. We believe that our approach is generic and can be utilized to discover multiple items and respective attributes from an unstructured text in other domains such as drug discovery in health-care.

4. REFERENCES

[1] M. Joshi, E. Hart, M. Vogel, and J.-D. Ruvini. Distributed word representations improve ner for e-commerce. In *Proceedings of NAACL-HLT*, 2015.

[2] X. Ma and E. Hovy. End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *Proceedings of 54th ACL, 2016*, Germany.

[3] K. Mauge, K. Rohanimanesh, and J.-D. Ruvini. Structuring e-commerce inventory. In *Proceedings of the 50th ACL, 2012.*

[4] K. Pallika, W. Michael, and P. Adam. Attribute extraction from noisy text using character-based sequence tagging models. In *Machine Learning for eCommerce workshop, NIPS*, 2015.

[5] N. Peng and M. Dredze. Multi-task multi-domain representation learning for sequence tagging. *CoRR*, abs/1608.02689, 2016.