Pricing the Nearly Known - When Semantic Similarity is Just not Enough

Gilad Fuchs*, Pavel Petrov*, Ido Ben-Shaul, Matan Mandelbrod, Oded Zinman, Dmitry Basin, Vadim Arshavsky*

{gfuchs,ppetrov,ibenshaul,mmandelbrod,ozinman,dbasin,varshavsky}@ebay.com

eBay Research, Israel

ABSTRACT

Helping sellers price their listings is an important and challenging task at E-commerce marketplaces, as the information provided by sellers is often partially structured and lacking. To help the seller gain trust in the recommended price, a collection of supporting similar listings are retrieved and provided along with their prices. We address the problem of retrieval-based price recommendation using a novel approach, which enables a trade-off adjustment between semantic similarity and price accuracy. Balancing the two required since, based on our study, retrieval of semantically similar listings does not guarantee pricing accuracy. In contrast, a price-accuracy driven approach may produce less semantically supporting listings. We also suggest a third method - training a Multi-Task network which learns in parallel both semantic similarity and a pricing-based objective. Framing the solution as a Multi-Task network unfolds the ability to control the balance between explainability and accuracy, thus providing a powerful tool to precisely tailor the correct pricing solution to different real world business use cases.

CCS CONCEPTS

• **Applied computing → Electronic commerce**.

KEYWORDS

Transformers, Sentence Similarity, E-commerce

1 INTRODUCTION

Price recommendation commonly exists in various E-commerce marketplaces listing creation forms and is aimed to help sellers price their listings correctly, reduce the time needed to perform market research and increase the chances of conversion. Retrieval-based

---

*Both authors contributed equally to this research.

price recommendation is based on aggregating (e.g. averaging) the prices of a set of similar listings. As introduced in [5], the main challenge in retrieval-based price recommendation stems from the fact that many listed items do not have internationally recognized product identifiers (such as GTIN) associated with them. This means the listed items are often defined using the information provided by the seller during the listing creation. Such information is semi-structured, and not standardized - a given listing may be titled differently by different sellers, and the set of associated attributes may be partially provided. Thus, the basic challenge a retrieval-based pricing method faces is the identification of a set of similar listings for a given target listing to be priced. As the listing title often contains the most concise and relevant information, our work is based solely on titles rather than additional elements associated with the listing.

The intent-driven similarity approach in [5] is based on the realization that users' queries can be used to assign weights to each of the title's token embeddings. The drawback of the query-based weighing is the fact that some titles may include tokens which are highly significant for pricing, yet are rarely used in buyer queries. Examples of such tokens are the memory size when searching for a laptop (e.g. '128 GB'), or the quantity in multi-quantity listings (e.g. 'Pack of 4 cards'). We present two approaches to improve the price recommendation, and then a combined method which enables tuning the trade-offs between the two. The first approach follows the attitude in [5] in the sense that it adheres to semantic textual similarity between the embedding vectors of listing titles. However, rather than using the set of user queries to assign weights to each title token, it uses a Siamese dual encoder to compute titles embeddings. In this scheme, similar listings are ones who were co-clicked by a user in the same search session. The second approach (dubbed Title2Price) relinquishes the explicit establishment of semantic similarity, and instead computes the direct function assigning prices to titles. It uses embeddings from a BERT model [3], fine-tuned to compute prices directly. This approach achieves better performance than the first one in terms of price accuracy metrics. However, the resulting title embeddings, having been constructed in a price-oriented manner, may turn out to have lower semantic similarity to a given seed listing than the ones computed by the Siamese-based approach. In some use cases, this may incur lower trust of the recommended price by the sellers.

Thus we have two pricing models - the first achieves high semantic similarity on account of lower pricing accuracy, and the second improves pricing accuracy on account of the semantic similarity. To reconcile the trade-off between the objectives, we've established a third, combined model, which enables tuning this trade-off to achieve target business goals. The combination of the two models
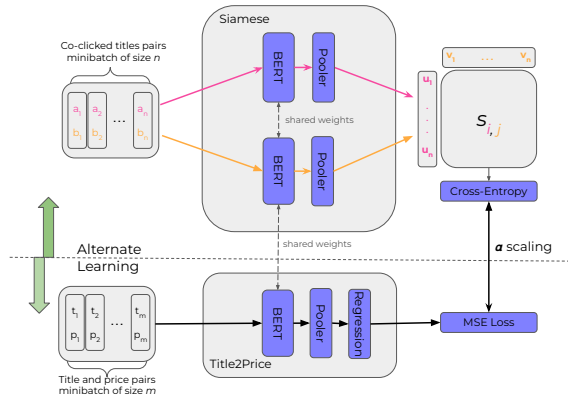
Figure 1: The Multi-Task model architecture.

is implemented as a multi-task network, which optimizes semantic similarity and price accuracy in parallel.

2 RELATED WORK

Automatic listing pricing problems arise in contexts such as E-commerce, accommodation, assets value estimation and more. The task of listing pricing has been previously studied by several related works. In the Kaggle Mercary challenge [1] the approach taken is based on feature engineering followed by a regression model. A similar method has been applied in [11]. In our settings, explicit feature extraction is replaced by feeding the listing titles through a BERT encoder [3] and using pooling to generate title embeddings. Due to the variability and loose structure of titles provided by sellers, such approach has proved more robust and of higher performance than explicit feature engineering.

More advanced approaches to E-commerce pricing consist of converting textual, visual and structured inputs associated with the query listing to a vector representation. This is followed by a linear regression, Neural Network or other relevant scheme using the embeddings as input features. Such an approach is presented in [6] whereby a price suggestion system for online second-hand listings is based on their uploaded images and text descriptions. In [4] the model consist a combination of LSTM and CNN, for processing textual features and visual features. While combination of textual and image data is a valid comprehensive approach, our experience indicates that information which is relevant for pricing is typically mostly specified in the listings titles. A similar realization is concluded in [9], where images and their associated text are used to train pricing models. Based on these results we've limited our inputs to titles.

The prominent added value of our work stems from applying multi-task learning so as to balance semantic similarity and price accuracy in dense retrieval settings. To the best of our knowledge, no previous works have identified and tried to tune the trade-off between the two.

3 METHODS

Siamese model. In this work we follow [12] and train Siamese dual encoder [2] based on BERT encoder. The training data is based

on a eBay's search engine logs. For each query, we consider the search results (listings) that were clicked by the user. For a given query, if there are k "co-clicked" listings in the search results, they are all labeled as similar and aggregated into a pool of similar listings. To encourage the co-clicked listings to be similar, we filter search queries that are short and nonspecific by removing queries with less than 6 tokens. The number of tokens chosen is based on business prior knowledge of average meaningful tokens per title. Using the listings pool, we further sample pairs of similar listings for each query: either undersampling $\lfloor 2\sqrt{k} \rfloor$ pairs for the training set or two pairs for test/validation sets. We apply the main BERT encoderr and the subsequent pooling layer to each title, thus producing pairs of embeddings. Since we only have positive examples, we use the "Multiple Negatives" loss function, as described in [7]. Using this method, negative samples are obtained from the non-positive samples found in the same batch.

Title2Price model. The Title2Price model is a BERT model with a regression layer, which is fine-tuned using sold listings with an MSE loss function. The listing title serves as the input, whereas the response variable is the sold price of the listing transaction. The prices were transformed using $\log(1+p)$ to achieve two goals. First, the mapping results in modeling the ratio between the predicted and real prices, which is more suitable than the absolute difference and provides better performance (data not shown). Second, it solves the issue of potentially predicting non-positive prices. Once the model has been fine-tuned, we present two ways of making predictions: 1. The output of the final regression head. 2. Using the model to produce title embedding and make a prediction based on its nearest neighbors' prices. The advantage of the second approach is explainability - as opposed to the "black box predictions" of the first approach.

Multi-task model. The Multi-Task architecture combines both Title2Price and Siamese dual encoder. In this case, both models share the BERT and the Pooling layers. We take an approach similar to the one taken in [10], where at each step, a task is chosen stochastically based on varying weights. The training process for k different tasks is as follows. The task sampling depends on the task weights $\omega_1, \ldots, \omega_k$ and mini-batch sizes for each task $n_1, \ldots, n_k$. At each iteration, the j-th task is selected with probability proportional to $\lceil \omega_j / n_j \rceil$.

In our specific setting we have two tasks: $\mathcal{T}_{\text{Title2Price}}, \mathcal{T}_{\text{Siamese}}$ with weights $\omega_1 = 1.0, \omega_2 = 0.5$ accordingly, to balance the difference in data size of both tasks. The cost of the Siamese model is multiplied by a normalizing factor $\alpha \in \mathbb{R}_+$ such that $\mathcal{L}_{\text{MT}}$ is composed of $\mathcal{L}_{\text{Title2Price}}$ and $\alpha \times \mathcal{L}_{\text{Siamese}}$, depending on the task chosen at this iteration. When $\alpha = 0$, the model is identical to the Title2Price model (essentially skipping the Siamese iterations). However, when α is large, the effect of the Siamese iterations are dominant, and the model is optimized similar to the plain Siamese model. An overview of the Multi-Task model architecture is shown in figure 1.

Title Embeddings. The embedding process is the same for all models: we apply a pooling function to the hidden states from the final layer of BERT. Following [12], we compare two pooling functions:

(1) CLS pooler: returns the hidden state for the CLS token.
(2) MEAN pooler: returns the average of hidden states for all tokens.

The pooler used for the embeddings generation was always the same as used for the model training.

KEN: k, ε-neighbors. We used a modified version of k-nearest neighbors called k, ε-neighbors (KEN). The modified version allows soft thresholding and a larger recall set per query. For a given vector q and a distance metric $d(\cdot, \cdot)$, a set $R_{k,\varepsilon}(q)$ of k, ε-neighbors for q is defined as follows. Let $R_k(q)$ be a set of k-nearest neighbors for q. Then:

$$R_{k,\varepsilon}(q) = \left\{ r : d(q,r) \leq \max_{r' \in R_k(q)} \left[d(q,r') \right] + \varepsilon \right\}$$

In this work we used cosine distance. The nearest neighbors were found by using the Faiss packages [8].

4 TRAINING AND EVALUATION

Datasets. There are three types of datasets:

(1) Seed dataset: title and price for each listing. It includes ~440K listings sold on eBay during a two-week period. This dataset is further split into training (~400K), validation (20K), and test (20K) subsets.
(2) Pool dataset: title and price for each listing. It includes ~15M listings sold on eBay in the 180 days that precede the "seed listings" period.
(3) Co-clicked dataset: titles of listing pairs that were clicked in the same search session. It consists of ~4M co-clicked listing pairs.

For the Pool and Seed datasets, we use the selling price as the response variable. We use the Pool dataset to fine-tune the Title2Price Model and the Co-clicked dataset to fine-tune the Siamese Model. Both these datasets are used to fine-tune the Multi-task Model. The Seed dataset's train and validation sets are used for optimizing the KEN parameters, and the test subset is used to report final metrics.

Metrics. The primary metric for price prediction is Mean Absolute Error (MAE). We assess the explainability of the models by looking at the semantic similarity of a given query to its corresponding nearest neighbors listings. This is done by incorporating attributes assigned to listings, such as Brand, Model, Color, etc. We've introduced a "semantic" metric called Attribute Mismatch Percentage (AMP), which is defined as follows. For each attribute a, and a set of seed listings S that contain a value for this attribute, we take the k nearest neighbors $\mathrm{NN}_k(s)$ for each listing $s \in S$. AMP is then defined as:

$$\mathrm{AMP}(S, a) = 100 \cdot \frac{\sum\limits_{s \in S} \sum\limits_{r \in \mathrm{NN}_k(s)} \mathbb{1}\{a(s) \neq a(r), a(r) \neq \varnothing\}}{\sum\limits_{s \in S} \sum\limits_{r \in \mathrm{NN}_k(s)} \mathbb{1}\{a(r) \neq \varnothing\}}$$

We set $k = 10$ in our experiments under the assumption that users usually do not examine more than 10 top similar listings. As listing attribute information is prevalent across many verticals in E-commerce marketplaces, the metric is transferable to a wide array of categories.

Training. All three model types were trained with both CLS and MEAN pooling. For the Multi-task Model we trained an array of models with gradually increasing values of α. Each model was trained on 8 GPUs, for at most 20 epochs. Adam optimizer with weight decay of 0.01 and a linear scheduler with warm-up (for 500 steps) were used, with an initial rate of $2 \cdot 10^{-5}$. We used a Batch size (per GPU) of 32 for Title2Price and 128 for Siamese (same for the Title2Price and Siamese batches in the Multi-task training).

After each training epoch the following procedure (the same for all models) is performed: (1) Embeddings are produced for both the Seed and Pool datasets. (2) Hyperparameters k and ε for k, ε-neighbors are tuned on the Seed training dataset (see below). (3) Price metrics are calculated for the Seed validation set via the k, ε-neighbors method.

For each listing in the Seed dataset we look for k, ε-neighbors in the Pool dataset. The prediction of the price $\hat{p}(q)$ per seed listing q is given by the median of the neighbors. The tuning of k and ε is done by a grid search, while minimizing the MAE on the validation subset of the Seed dataset. Each model's best checkpoint is selected by the k-nearest neighbors validation MAE results. Finally, the selected checkpoint is used to calculate the metrics on the Seed test dataset.

5 EXPERIMENTS AND RESULTS

In this work, we focused on one popular category on the eBay marketplace - the "Sport Trading Cards" category. Predicting listing prices with the Title2Price model can be done either directly by its regression layer, or by extracting its embeddings to find similar listings and aggregate their price. In order to compare the price prediction accuracy of both approaches, the MAE between the predicted price and actual transaction sold price was measured during the model fine-tuning. Measuring the MAE for epochs with minimal validation MAE on seed test data shows that k, ε-neighbors is on par with the regular BERT regression for CLS pooling (29.3 vs 29.3 resp.) and outperforms it for MEAN pooling (29.7 vs 31.1 resp.).

Figure 2 shows seed test data MAE for Title2Price, Siamese and Multi-task with different values of the tasks scaling parameter α. Figure 3 shows Attribute Mismatch Percentage (AMP) on the test data for two attributes: Player and Grade. We can see a trade off between the performance of the price prediction and the explainability (semantic similarity of the nearest neighbors). As the scale (α) is increased, there's an ascending trend in MAE and a descending trend in AMP for Player. AMP for Grade, however, exhibits a behavior which is more similar to MAE and tends to grow. The Grade (e.g. PSA Grade) of a Sport Card describes its condition quality. The price of a card is dramatically affected by its grading score. This could explain the different behavior between the Player attribute, compared to the Grade attribute.

Of note, the CLS pooling outperforms the MEAN pooling in terms of MAE (Figure 2) and Grade AMP (Figure 3), while under-performing the MEAN pooling in Player AMP (Figure 3), thereby further demonstrating the pricing-semantic trade-off.

Online evaluation. We evaluate our approach in an online production setting as part of eBay's Price Recommendation service, as described in Section 1. The current production requirements
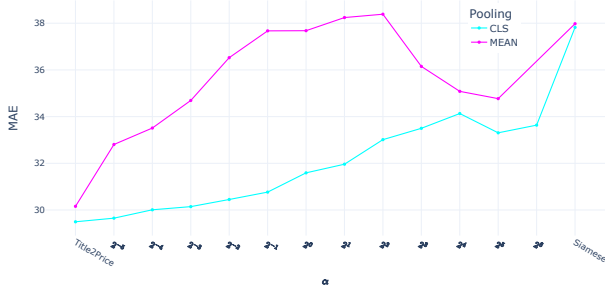
Figure 2: MAE on the seed test data for the Title2Price, the Siamese, and Multi-Task models, on a range of the scale parameter α.
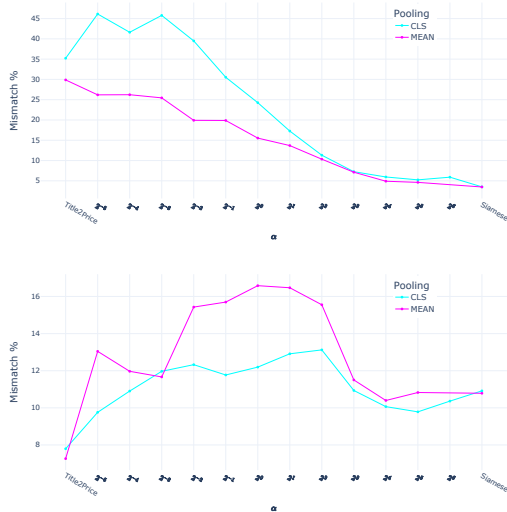

Figure 3: AMP on the seed test data for the Title2Price, Siamese, and Multi-Task models (defined by the scale parameter α) for different attributes. Top: Player, Bottom: Grade.

emphasize the price prediction accuracy (as opposed to semantic similarity), so the Title2Price model was used. Furthermore, the service also has to present similar listings that contributed to the recommendation. Therefore, we use the k, ε-neighbors (KEN) method described in Section 3 to predict the price and retrieve the list of neighbors. In order to encompass the performance requirements of *KNN* search in a production setting, we reduce the dimension of title embeddings from 768 to 256 by adding a linear layer to the original Title2Price model. For the adjusted model, we use MEAN-Pooler as it shows slightly better accuracy for the reduced embeddings.

We deployed the production-adjusted Title2Price model and performed an A/B test to compare it with the current service performance. The test lasted 30 days during the beginning 2022. The new model served 20% of the Price Recommendation service traffic for the "Sport Trading Cards" category during the test.

Table 1: Title2Price vs. production model performance in online A/B test.

	P5	P10	MAE	RMSE
Improvement	8.8%↑	7.5%↑	5.6%↓	29.3%↓

We compared the performance by four main evaluation metrics: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and the adoption rate within a 5% and 10% bi-directional margin (which we define as *P5* and *P10*). These metrics serve as a proxy to the seller's acceptance with the recommendation and correlate with the platform's long-term business success. The Title2Price model demonstrated significant improvement in all four metrics, as can be seen in Table 1.

The online evaluation required creating a scalable infrastructure to execute the model and generate recommendations with low latency in real-time. We deployed a cluster of 48 nodes, each $178GB$ RAM 24 CPU cores with Ubuntu Linux. When a seller creates a new listing, the listing's title is sent to a service node. The node generates the title's embedding using Title2Price Tensorflow model and performs LSH index lookup to fetch a short list of similar listing candidates for recommended price calculation using KEN method described in Section 3.

6 CONCLUSIONS

In this work we propose several models to tackle the task of E-commerce price prediction based on non-structured text. We present two different models, The first, **Title2Price**, introduced to regress the ground truth price and the other, **Siamese**, trained to retrieve semantically similar embeddings. In theory, both tasks should work towards the same goal: embeddings from the Title2Price model would be expected to make semantically similar listings close, as their prices are generally close, and embeddings from the Siamese model should have relatively similar prices. To our surprise, this is not the case. Vanilla Semantic Similarity models lack the ability to differentiate which parts of a title are volatile to the price, and which are non important. For example, two sport cards titles with exactly the same phrasing, but a different grading can have dramatically different prices, while two sport cards of a different year may be of a similar price. On the other hand, the Title2Price model may be very good at predicting the price, but in doing so it might use entirely different cards. Suppose the cards with the titles "AJ Green 2011 Topps Football Rookie Card" and "Nickeil Alexander 2019 Prizm Silver" have very similar prices. The Title2Price model may push the embeddings to be close together, although to a user this may seem very 'wrong', as these cards are not similar. To overcome this difficulty, we present a **Multi-Task** model to balance the two objectives. The models can be trained to support more semantic similarity, or more price accuracy. To test the trade-off between the different models, we present the AMP metrics, on different listing attributes. We show a detailed analysis of the performances using well-known metrics and the ones proposed in this paper.

REFERENCES

[1] Amine Ait Si Ali, Huseyin Seker, Steven Farnie, and John Elliott. 2018. Extensive Data Exploration for Automatic Price Suggestion Using Item Description: Case Study for the Kaggle Mercari Challenge. In *Proceedings of the 2nd International Conference on Advances in Artificial Intelligence, ICAAI 2018, Barcelona, Spain, October 06-08, 2018*. ACM, 41–45. https://doi.org/10.1145/3292448.3292458

[2] J. Bromley, James W. Bentz, L. Bottou, I. Guyon, Y. LeCun, C. Moore, Eduard Säckinger, and R. Shah. 1993. Signature Verification Using A "Siamese" Time Delay Neural Network. In *Int. J. Pattern Recognit. Artif. Intell.*

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv:1810.04805 [cs.CL]

[4] Ahmed Elsayed Fathalla, Ahmad Salah, Kenli Li, Keqin Li, and Piccialli Francesco. 2020. Deep end-to-end learning for price prediction of second-hand items. *Knowledge and Information Systems* (2020), 1 – 28.

[5] Gilad Fuchs, Yoni Acriche, Idan Hasson, and Pavel Petrov. 2020. Intent-Driven Similarity in E-Commerce Listings. In *Proceedings of the 29th ACM International Conference on Information Knowledge Management* (Virtual Event, Ireland) *(CIKM '20)*. Association for Computing Machinery, New York, NY, USA, 2437–2444. https://doi.org/10.1145/3340531.3412715

[6] Liang Han, Zhaozheng Yin, Zhurong Xia, Mingqian Tang, and Rong Jin. 2020. Price Suggestion for Online Second-hand Items with Texts and Images. arXiv:2012.06008 [cs.AI]

[7] Matthew Henderson, Rami Al-Rfou, B. Strope, Yun-Hsuan Sung, László Lukács, Ruiqi Guo, Sanjiv Kumar, Balint Miklos, and R. Kurzweil. 2017. Efficient Natural Language Response Suggestion for Smart Reply. *ArXiv* abs/1705.00652 (2017).

[8] Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2017. Billion-scale similarity search with GPUs. *arXiv preprint arXiv:1702.08734* (2017).

[9] Bin Li and Teng Liu. 2021. An Analysis of Multi-Modal Deep Learning for Art Price Appraisal. *2021 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom)* (2021), 1509–1513.

[10] Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019. Multi-Task Deep Neural Networks for Natural Language Understanding. In *ACL*.

[11] Nabarun Pal, Priya Arora, Dhanasekar Sundararaman, Puneet Kohli, and Sai Sumanth Palakurthy. 2017. How much is my car worth? A methodology for predicting used cars prices using Random Forest. arXiv:1711.06970 [cs.CY]

[12] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *EMNLP/IJCNLP*.