Images Don't Lie: Transferring Deep Visual Semantic Features to Large-Scale Multimodal Learning to Rank

Corey Lynch
Etsy
55 Washington Street
Brooklyn, New York
clynch@etsy.com

Kamelia Aryafar
Etsy
55 Washington Street
Brooklyn, New York
karyafar@etsy.com

Josh Attenberg
Etsy
55 Washington Street
Brooklyn, New York
jattenberg@etsy.com

ABSTRACT

Search is at the heart of modern e-commerce. As a result, the task of ranking search results automatically (learning to rank) is a multi-billion dollar machine learning problem. Traditional models optimize over a few hand-constructed features based on the item's text. In this paper, we introduce a multimodal learning to rank model that combines these traditional features with visual semantic features transferred from a deep convolutional neural network. In a large scale experiment using data from the online marketplace Etsy [1], we verify that moving to a multimodal representation significantly improves ranking quality. We show how image features can capture fine-grained style information not available in a text-only representation. In addition, we show concrete examples of how image information can successfully disentangle pairs of highly different items that are ranked similarly by a text-only model.

Keywords

Learning to rank; Computer vision; Deep learning;

1. INTRODUCTION

Etsy [1] is a global marketplace where people buy and sell unique goods: handmade items, vintage goods, and craft supplies. Users come to Etsy to search for and buy listings other users offer for sale. A *listing* on Etsy consists of an image of the item for sale along with some text describing it. With 35 million listings for sale [2], correctly ranking search results for a user's query is Etsy's most important problem. Currently Etsy treats the ranking problem as an example of supervised learning: learn a query-listing relevance function from data with listing feature representations derived from listings' titles and tags. However, with over 90 million listing images, we wonder: *is there useful untapped information hiding in Etsy's images that isn't well captured by the descriptive text?* If so, how can we integrate this new data modality into Etsy's existing ranking models in a principled way? In this paper we attempt to explore these two major questions on a real world dataset containing over 1.4 million Etsy listings with images. Specifically, we describe a multimodal learning to rank method which integrates both visual and text information. We show experimentally that this new multimodal model significantly improves the quality of search results.

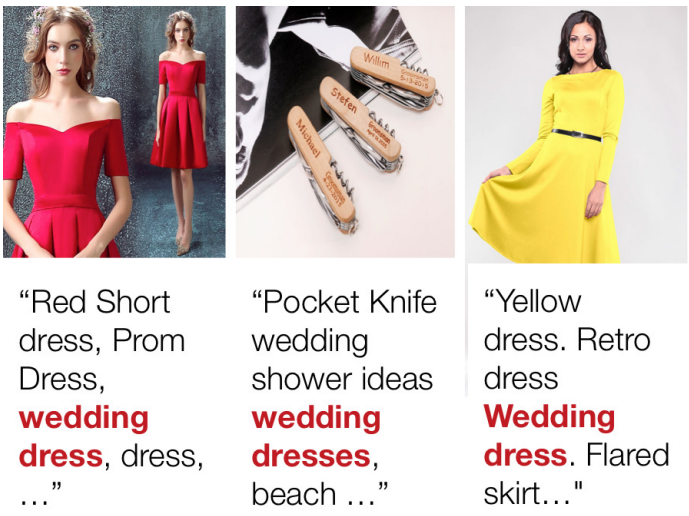

Figure 1: **Irrelevant search results for the query "wedding dress":** Even though it's apparent in the images that these are not wedding dresses, each listing's descriptive title contains the phrase "wedding dress", allowing it to show in search results for the query.

We motivate this paper with a real example of a problem in search ranking. Figure 1 shows listings that appear in the results for the query "wedding dress". Even though it is clear from looking at the images that they aren't wedding dresses, each listing's title contains the term "wedding dress", causing it to appear in the results. This kind of term noise in listing descriptions is pervasive. Sellers write their own descriptions and are motivated to include high traffic query terms to boost the visibility of their listings in search. It is obvious that there is *complementary information* in the listing images that is either unavailable in the text or is even in contrast to the text. Our hypothesis is that if we can mine this complementary high-level visual information, we can incorporate it into our models to improve search ranking as suggested by multimodal embeddings literature [5, 7, 8, 13, 14, 15, 17, 20, 26, 28].

With the goal of learning high-level content directly from the image, deep convolutional neural networks (CNN) [4, 22] are an obvious model choice. CNNs are a powerful class of models in-

[1] www.etsy.com

[2] The statistics reported in this paper are accurate at the time of submission of this paper.

KDD '16, August 13-17, 2016, San Francisco, CA, USA

DOI: http://dx.doi.org/10.1145/2939672.2939728

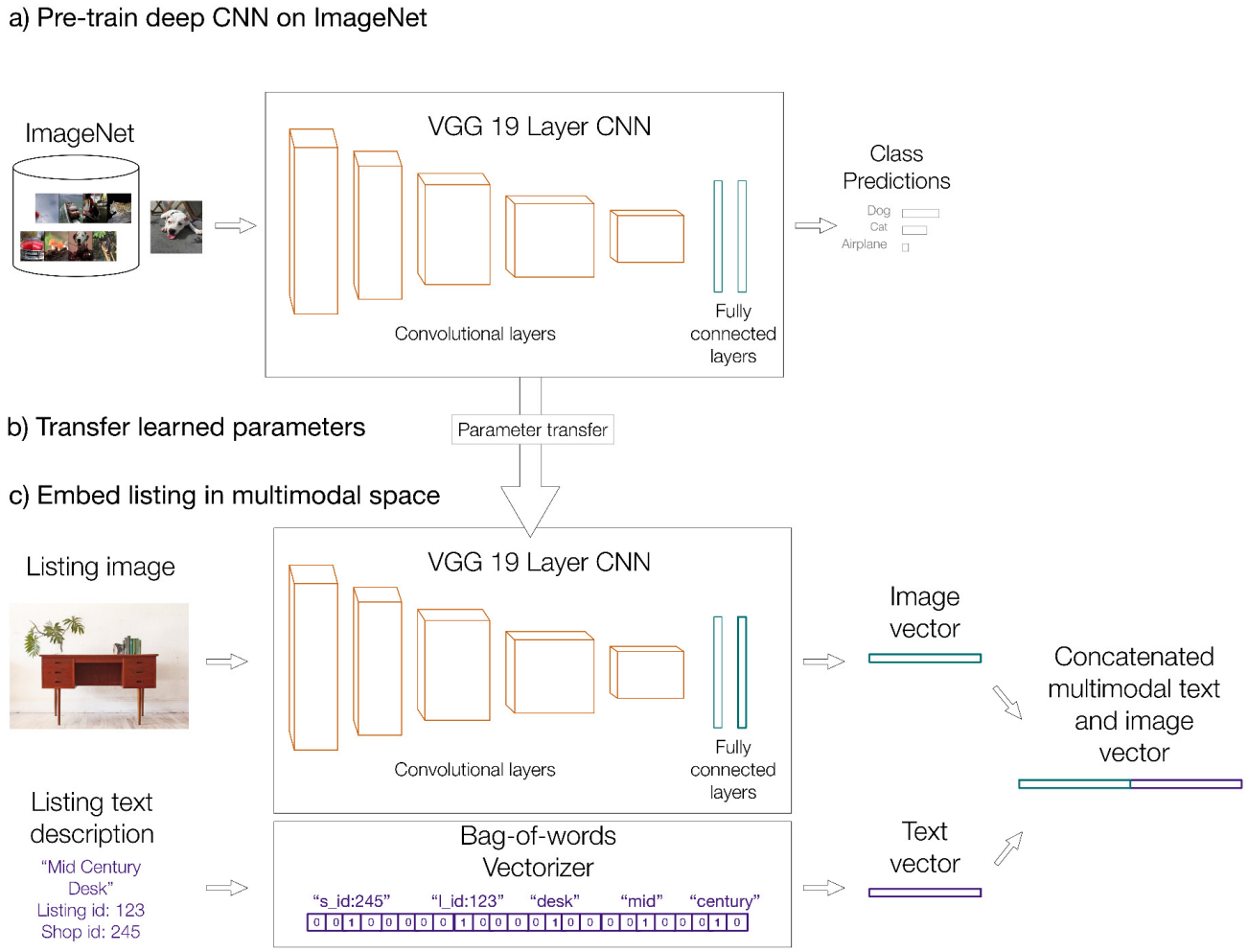

Figure 2: **Transferring parameters of a CNN to the task of multimodal embedding:** In a) we utilize a pre-trained 19 layer VGG-style network that is trained on a large scale object recognition task (ImageNet challenge). In b) we remove the last layer (containing scores for the different object classes) and transfer the parameters of the modified network to our task. In c) we use the modified network as a fixed feature extractor of high-level image content information, taking the last fully connected layer as an image embedding. We simultaneously embed the listing's text in a bag of words space, then concatenate the two embeddings to form a single multimodal descriptor of a listing.

spired by the human visual cortex that rivals human-level performance on difficult perceptual inference tasks such as object recognition [16]. [29] shows that image feature representations learned on large scale object recognition tasks are powerful and highly interpretable. The lower layers of the network learn low-level features like color blobs, lines, corners; middle layers combine lower layers into textures; higher layers combine middle layers into higher-level image content like objects. High-level visual information is made increasingly explicit along the model's processing hierarchy [6]. This high-level description formed in the deep layers of the network is what we are interested in mining as a rival source of information to the listing's text description.

One consideration to make is that large modern CNNs require large amounts of training data [16, 24]. Even though Etsy's production search system generates millions of training examples per day in aggregate, the amount of examples available to an individual query model can be in the low hundreds, particularly for queries in the long tail. This makes training one deep CNN per query from scratch prone to overfitting. Transfer learning is a popular method for dealing with this problem, with many examples in computer vision [1, 19, 25]. We take the pre-trained 19-layer VGG net [23, 4] as a fixed extractor of general high-level image features. We chose this model due to it's impressive performance on a difficult 1000-way object classification task. Our assumption is that the activations of it's neurons immediately prior to the classification task contain general high-level information that may be useful to our ranking task. Figure 2 gives a high-level overview of our multimodal feature extraction process.

Learning to rank search results has received considerable attention over the past decade [2, 3, 9, 12], and it is at the core of modern information retrieval. A typical setting of learning to rank for search is to: (i) embed documents in some feature space, (ii) learn a ranking function for each query that operates in this feature space over documents. Early approaches optimized over a few

Algorithm 1 Multimodal Embedding of Listings

procedure EMBEDMULTIMODAL($\mathbf{d}_i$)
  $\mathbf{d}_{T_i} \leftarrow$ **BoW**$(text)$
  $\mathbf{d}_{I_i} \leftarrow$ **VGG**$(image)$
  $\mathbf{d}_{MM_i} \leftarrow [\mathbf{d}_{T_i}, \mathbf{d}_{T_i}]$
  return $\mathbf{d}_{MM_i}$

hand-constructed features, e.g. item title, URL, PageRank [12]. More recent approaches optimize over much larger sets of orthogonal features based on query and item text [2]. Our research follows this orthogonal approach and explores the value of a large set of image features. Our baseline listing representation consists of features for a listing's terms, a listing's id, and a listing's shop id. The presence of the latter two features captures historical popularity information at the listing and shop level respectively for the query.

To our knowledge, this is the first investigation of the value of transferring deep visual semantic features to the problem of learning to rank for search. The results of a large-scale learning to rank experiment on Etsy data confirms that moving from a text-only representation to a multimodal representation significantly improves search ranking. We visualize how the multimodal representation provides complementary style information to the ranking models. We also show concrete examples of how pairs of highly different listings ranked similarly by a text model get disentangled with the addition of image information. We feel this, along with significant quantitative improvements in offline ranking metrics demonstrates the value of the image modality.

This paper is organized as follows: In Section 2 we describe our multimodal ranking framework. Section 2.1 gives a detailed explanation of how we obtain multimodal embeddings for each listing. Section 2.2 gives a brief introduction to learning to rank and describes how these embeddings are incorporated into a pairwise learning to rank model. Section 3 describes a large scale experiment where we compare multimodal models to text models. Finally in Section 4, we discuss how moving to a multimodal representation affects ranking with qualitative examples and visualizations.

2. METHODOLOGY

Here we describe how we extend our existing learning to rank models with image information. We first explain how we embed listings for the learning to rank task in both single modality and multimodal settings. Then we explain how listings embedded in both modalities are incorporated into a learning to rank framework.

2.1 Multimodal Listing Embedding

Each Etsy listing contains text information such as a descriptive title and tags, as well as an image of the item for sale. To measure the value of including image information, we embed listings in a multimodal space (consisting of high-level text and image information), then compare the multimodal representation to the baseline single modality model. Let $\mathbf{d}_i$ denote each listing document. We then use $\mathbf{d}_{T_i} \in \mathbb{R}^{|T|}$ and $\mathbf{d}_{I_i} \in \mathbb{R}^{|I|}$ to denote the text and image representation of $\mathbf{d}_i$ respectively. $|T|$ denotes the dimensionality of the traditional feature space which is a sparse representation of term occurrences in each $\mathbf{d}_i$. $|I|$ denotes the dimensionality of the image feature space which in contrast to text is a dense feature space. The goal of multimodal embedding is then to represent a listing through text and image modalities in a single vector, i.e. the $\mathbf{d}_{MM_i} \in \mathbb{R}^{|M|}$ where $|M|$ is the dimensionality of the final embedding.

Given a listing document $\mathbf{d}_i$ consisting of the title and tag words, a numerical listing id, a numerical shop id, and a listing image,

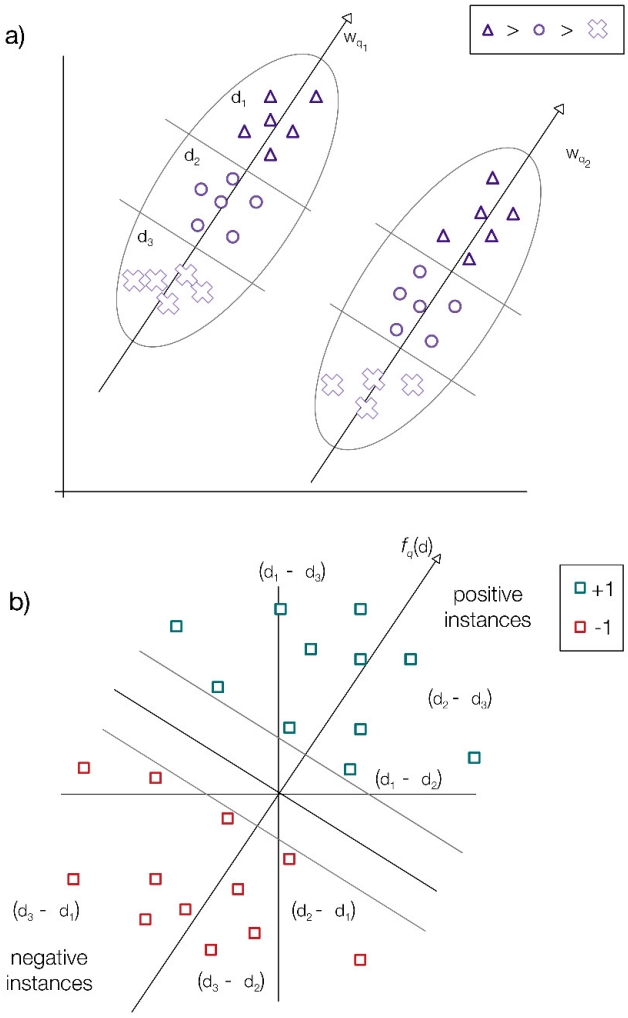

Figure 3: **Transformation to Pairwise Classification [9]:** a) Shows a synthetic example of the ranking problem. There are two groups of documents (associated with different queries) embedded in some feature space. Documents within each group have different relevance grades: $relevance(\mathbf{d}_1) > relevance(\mathbf{d}_2) > relevance(\mathbf{d}_3)$. The weight vector w corresponds to a linear ranking function $f(\mathbf{d}) = \langle w, \mathbf{d} \rangle$ which can score and rank documents. Ranking documents with this function is equivalent to projecting the documents onto the vector and sorting documents according to the projections. A good ranking function is one where documents in $\mathbf{d}_1$ are ranked higher than documents in $\mathbf{d}_3$, and so on. Documents belonging to different query groups are incomparable. b) shows how the ranking problem in (a) can be transformed into a pairwise classification task: separate well-ordered and non-well ordered pairs. In this task, well-ordered pairs are represented as the vector difference between more relevant and less relevant document vectors, e.g., $\mathbf{d}_1 - \mathbf{d}_2$, $\mathbf{d}_1 - \mathbf{d}_3$, and $\mathbf{d}_2 - \mathbf{d}_3$. Non-well ordered-pairs are represented as the vector difference between less relevant and more relevant document vectors, e.g., $\mathbf{d}_3 - \mathbf{d}_1$, $\mathbf{d}_2 - \mathbf{d}_1$, and $\mathbf{d}_3 - \mathbf{d}_2$. We label well-ordered instances $+1$, non-well-ordered -1, and train a linear SVM, $f_q(\mathbf{d})$, which separates the new feature vectors. The weight vector w of the SVM classifier corresponds to the ranking function w in (a).

Algorithm 2 Generate Pairwise Classification Instances

procedure GETPAIRWISEINSTANCES($\{\langle \mathbf{d}_i^+, \mathbf{d}_i^- \rangle\}$)
  $L \leftarrow \{\}$
  for $i = 1 \ldots |P|$ **do** ▷ $|P|$ labeled tuples
    $\mathbf{d}_{MM_i}^+ \leftarrow$ **EmbedMultimodal**$(\mathbf{d}_i^+)$
    $\mathbf{d}_{MM_i}^- \leftarrow$ **EmbedMultimodal**$(\mathbf{d}_i^-)$
    Draw r uniformly at random from $[0, 1)$
    if $r > 0.5$ **then**
      $x_i \leftarrow \mathbf{d}_{MM_i}^+ - \mathbf{d}_{MM_i}^-$
      $y_i \leftarrow +1$
    else
      $x_i \leftarrow \mathbf{d}_{MM_i}^- - \mathbf{d}_{MM_i}^+$
      $y_i \leftarrow -1$
  $L = L.append(\langle x_i, y_i \rangle)$
  return L ▷ The list of classification instances.

we obtain a multimodal feature vector $\mathbf{d}_{MM_i}$ by embedding traditional terms in a text vector space, embedding the image in an image vector space, then concatenating the two vectors. Algorithm 1 describes this multimodal embedding. The text-based features are based on the set of title and tag unigram and bigrams, the listing id, and the shop id for each $\mathbf{d}_i$. These baseline features are then represented in a bag-of-words (BoW) [27] space as $\mathbf{d}_{T_i} \in \mathbb{R}^{|T|}$. $|T| = |D| + |L| + |S|$, where $|D|$ is the size of the dictionary, $|L|$ is the cardinality of the set of listings, and $|S|$ is the cardinality of the set of shops. Each element in $\mathbf{d}_{T_i}$ is 1 if $\mathbf{d}_i$ contains the term, and is 0 otherwise.

Each $\mathbf{d}_i$ is also represented in the image space as $\mathbf{d}_{I_i} \in \mathbb{R}^{|I|}$. To obtain $\mathbf{d}_{I_i}$, we adopt a transfer learning approach. Oquab *et al.* [18] shows that the internal layers of a convolutional neural network pre-trained on ImageNet can act as a generic extractor of a mid-level image representation and then re-used on other tasks. In our model, we utilize the VGG-19 network [23] and remove the last fully-connected layer. For every listing $\mathbf{d}_i$, we scale it's image uniformly so that its shortest spatial dimension is 256 pixels, then take the center 224×224 crop. We then pass the cropped image through the modified VGG network to get a 4096 dimensional feature vector [3]. This vector contains the activations fed to the original object classifier. We ℓ_2 normalize the activations, then use the normalized vector as the final 4096 dimensional image representation $\mathbf{d}_{I_i}$. The multimodal representation of the document $\mathbf{d}_i$ can now be obtained simply as $\mathbf{d}_{MM_i} \in \mathbb{R}^{|M|} = [\mathbf{d}_{T_i}, \mathbf{d}_{I_i}]$ where $|M| = |T| + |I|$ is the dimensionality of the final multimodal representation. Figure 2 illustrates this process in details.

2.2 Learning To Rank

E-commerce search is a task that can be described as follows: (i) Query: a user comes to the site and enters a query, q, e.g. "desk". (ii) Retrieval: the search engine finds all listing documents that contain terms from the query in their title or tag descriptions, such as "mid century desk", "red desk lamp", etc. (iii) Ranking: the search engine uses a ranking function to score each listing document, where a higher score expresses that a listing document is more relevant to the query. The search engine sorts listings by that score, and presents the results to the user. The goal of learning to rank [3] is to automatically learn (iii), the ranking function, from historical search log data.

[3]We make our scalable VGG-19 feature extractor available at: https://github.com/coreylynch/vgg-19-feature-extractor

In this paper, we restrict our attention to the *pairwise preference* approach to learning a ranking function [10]. That is, given a set of labeled tuples P, where each tuple contains a query q, a relevant document $\mathbf{d}^+$ and an irrelevant document $\mathbf{d}^-$, we want to learn a function $f_q(\mathbf{d})$ such that $f_q(\mathbf{d}^+) > f_q(\mathbf{d}^-)$. In our case, relevance is binary and determined by the user: a user is said to judge a search result relevant if she purchases the listing, adds it to her cart, or clicks on the listing and dwells on it for longer than 30 seconds.

As shown by Herbrich *et al.* [10], the ranking problem can be transformed into a two-class classification: learn a linear classifier that separates well-ordered pairs from non-well-ordered pairs. This is illustrated in Figure 3. To achieve this, we can transform any implicit relevance judgement pair $(\mathbf{d}^+, \mathbf{d}^-)$ into either a well-ordered or non-well ordered instance. Specifically, suppose for each preference pair $(q, \mathbf{d}^+, \mathbf{d}^-)$ we flip a coin. If heads the preference pair $(q, \mathbf{d}^+, \mathbf{d}^-) \mapsto (\mathbf{d}^+ - \mathbf{d}^-, +1)$ (a well-ordered pair), else $(q, \mathbf{d}^+, \mathbf{d}^-) \mapsto (\mathbf{d}^- - \mathbf{d}^+, -1)$ (a poorly ordered pair).

This results in an evenly balanced pairwise classification dataset for each query q. Algorithm 2 explains the process of generating classification instances for input pairwise preference tuples. The new classification task can now be solved by minimizing the regularized hinge classification loss:

$$\min_w \sum_{i=1}^{m} max([1 - y_i \langle w, x_i \rangle], 0) + \lambda_1 \|w\|^1 + \lambda_2 \|w\|^2$$

via stochastic gradient descent. A well trained pairwise classifier minimizes the number of pairs which are ranked out of order, i.e. the ranking loss. To rank a new set of listing documents for a query, we embed them in the feature space, then use output of the trained classifier to obtain ranking scores for each. This method, also known as RankingSVM, is used extensively in the ranking literature [10, 3, 9].

Following [21], we can obtain large quantities of *implicit* pairwise preference instances cheaply by mining Etsy's search logs[4].

The process for doing so is as follows: A user comes to the site, enters a query, and is presented with a page of results. If she interacts with listing $\mathbf{d}_i$ and ignores the adjacent listing $\mathbf{d}_j$, a reasonable assumption is that she prefers $\mathbf{d}_i$ over $\mathbf{d}_j$ in the context of query q. We call this an implicit relevance judgement, mapping $(q, \mathbf{d}_i, \mathbf{d}_j) \mapsto (q, \mathbf{d}^+, \mathbf{d}^-)$, forming the necessary input for Algorithm 2. Figure 4 illustrates how we move from search logs to multimodal pairwise classification instances.

3. RESULTS AND DISCUSSION

This section describes a large scale experiment to determine how a multimodal listing representation impacts ranking quality. In Section 3.1, we describe our ranking quality evaluation metric. In Section 3.2 we describe our dataset. Finally we present our findings in Section 3.3.

[4]We obtain labeled pairs for each query using the FairPairs method. A well known and significant problem in collecting training data from search is presentation bias [21]: users click on higher presented results irrespective of query relevance. Ignoring this bias and training on naively collected data can lead to models that just learn the existing global ranking function. The FairPairs method modifies search results in a non-invasive manner that allows us to collect pairs of (preferred listing, ignored listing) that are unaffected by this presentation bias.

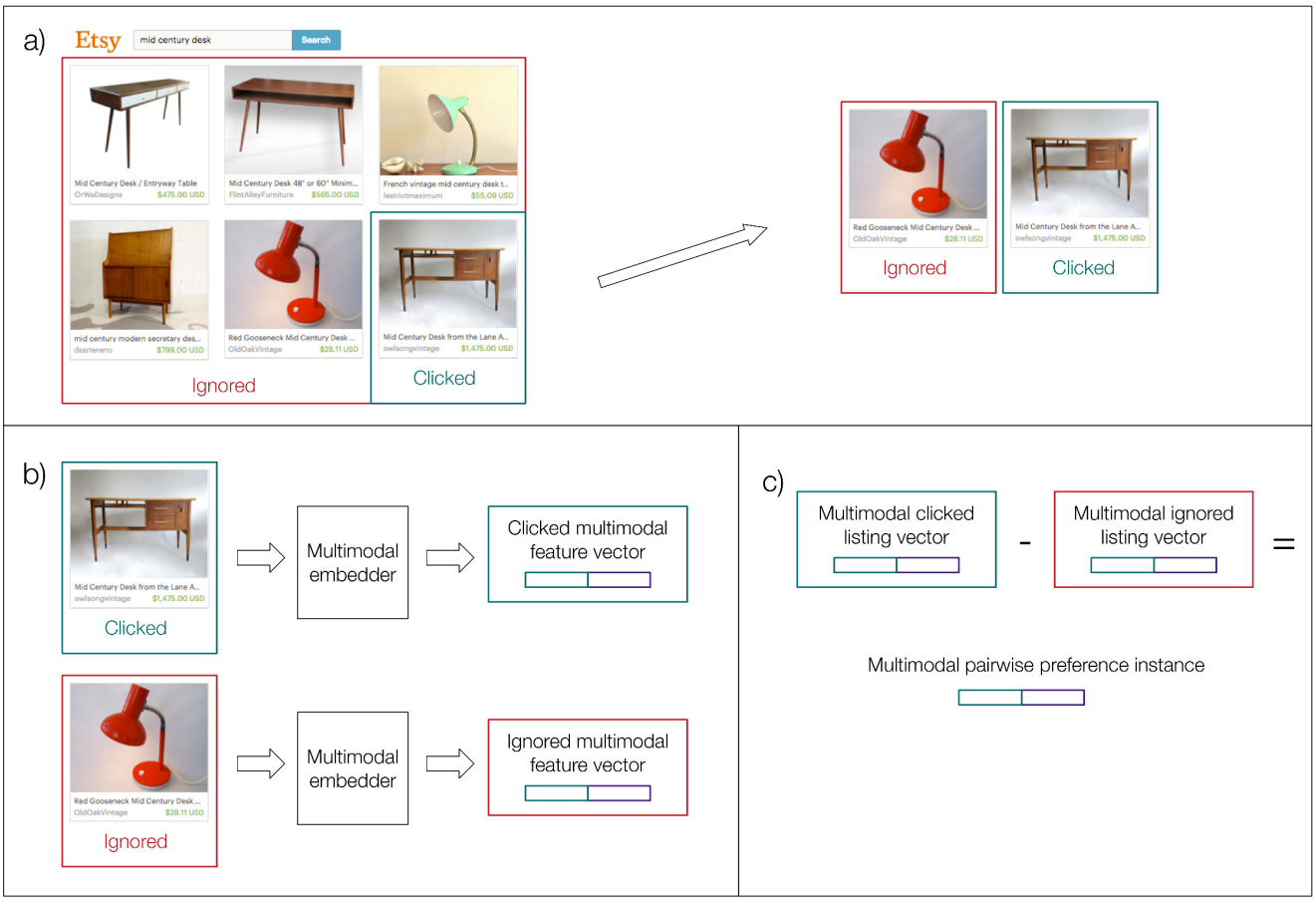

Figure 4: **From search logs to multimodal pairwise classification instances:** a) A user comes to the site, enters the search query **mid century desk**, is presented with some results, clicks one and ignores the others. We take the listing she clicked and an adjacent listing she ignored as an instance of implicit pairwise preference in the context of the query **mid century desk** forming a training triplet $(q, \mathbf{d}^+, \mathbf{d}^-)$ from (**mid century desk**, clicked listing, ignored listing). b) We embed both listings in the pair in multimodal space. c) Map labeled embedded pair to a single pairwise classification instance. We flip a coin. If heads, create a well-ordered pairwise instance (clicked vector - ignored vector) and label it $+1$; if tails, create a non-well-ordered pairwise instance (ignored vector - clicked vector) and label it -1.

3.1 Evaluation metrics

To evaluate the quality of one modality ranking model over another, we measure the model's average *Normalized Discounted Cumulative Gain* (NDCG) [11] on holdout search sessions. NDCG is the standard measure of a model's ranking quality in information retrieval. It lies in the $[0, 1]$ range, where a higher NDCG denotes a better holdout ranking quality. Here we give a brief background on NDCG and why it is a good choice for quantifying ranking quality.

The cumulative gain (CG) of a ranking model's ordering is the sum of relevance scores over the ranked listings. The CG at a particular rank position p is defined as:

$$\mathrm{CG_p} = \sum_{i=1}^{p} rel_i,$$

where rel_i is the *implicit* relevance of the result at position i.

CG on its own isn't a particularly good measure of the quality of a model's ordering: moving a relevant document above an irrelevant document does not change the summed value.

This motivates discounted cumulative gain (DCG) [3] as a ranking measure. DCG is the sum of each listing's relevance score *discounted by the position it was shown*. DCG is therefore higher when more relevant listings are ranked higher in results, and lower when they are ranked lower. The DCG is defined as:

$$\mathrm{DCG_p} = \sum_{i=1}^{p} \frac{2^{rel_i} - 1}{\log_2(i+1)}.$$

Finally, we arrive at Normalized Discounted Cumulative Gain (NDCG) by dividing a model's DCG by the ideal DCG (IDCG) for the session:

$$\mathrm{nDCG_p} = \frac{DCG_p}{IDCG_p}$$

This gives us a number between 0 and 1 for the model's ranking of a holdout search result set, where a higher NDCG approaches the ideal DCG, denoting a better ranking quality.

Our holdout data (both validation and test) are in the form of a list of labeled sessions for each query q. A labeled session is a page of search results presented to a user for the query q, where the user

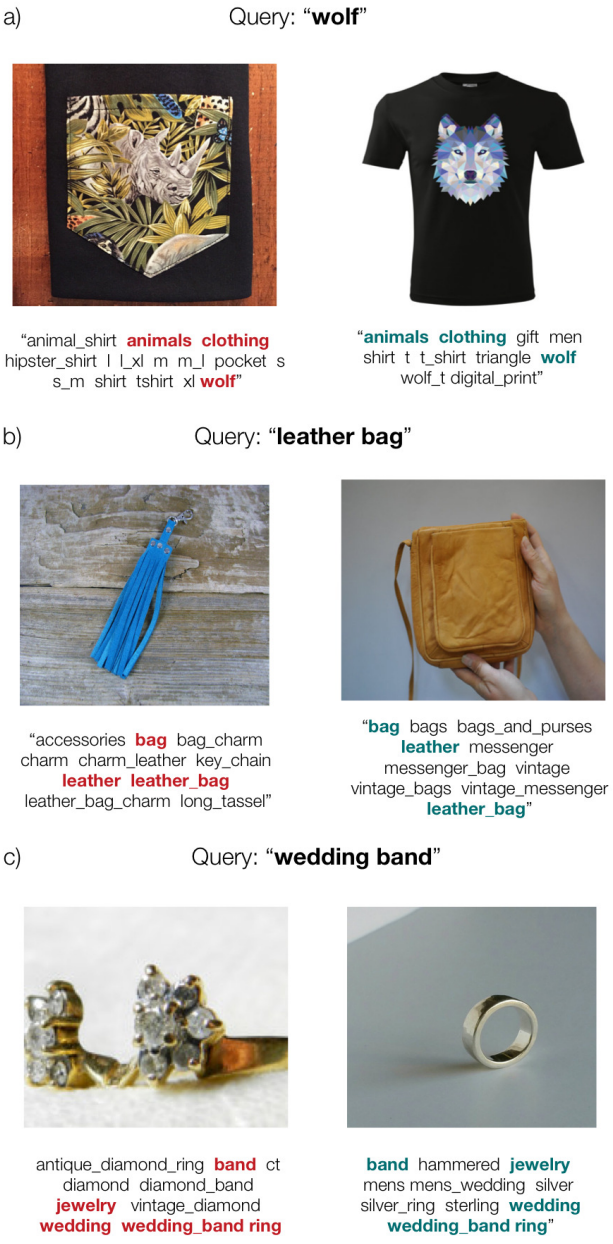

Figure 5: **Image information can help disentangle different listings considered similar by a text model:** Here are three pairs of highly different listings that were ranked similarly by a text model, and far apart by a multimodal model. Title terms that both listings have in common are bolded, showing how text-only models can be confused. In every case, the multimodal model ranks the right listing higher. All three queries benefit from substantial gains in ranking quality (NDCG) by moving to a multimodal representation. In a) for example, we see two listings returned for the query "wolf": the left listing's image shows a picture of a rhino, while the right listing's image shows a t-shirt with a wolf on it. A text-only model ranked these listings only 2 positions apart, likely due to the overlapping words in their titles. A multimodal model, on the other hand, ranked the right listing 394 positions higher than the left. The query "wolf" saw a 3.07% increase in NDCG by moving to a multimodal representation.

has provided implicit relevance judgements.[5] We evaluate a trained query model f_q by computing its average NDCG over all labeled holdout sessions for the query q. To achieve this, we first compute each session's NDCG as follows: i) embed all the session's listing documents in the model space, ii) score and rank embedded listings with the model, and iii) compute the labeled session NDCG. Then we average over session NDCG to get average query NDCG. Finally, to compare the ranking ability of models based in one modality vs. another, we report the average modality NDCG across query NDCGs.

3.2 Dataset

For our experiment, we select a random 2 week period in our search logs to obtain training, validation, and test data. We mine query preference pairs from the first week as *training data*. We mine labeled holdout sessions from the following week, splitting it evenly into *validation* and *test* sessions. This results in 8.82 million training preference pairs, 1.9 million validation sessions, and 1.9 million test sessions. Across training and test there are roughly 1.4 million unique listing images.

We tune model learning rates, λ_1 and λ_2 regularization strengths on our validation data. We also allow queries to select the best modality based on validation NDCG from the set of multiple modalities : *text-only*, *image-only*, and *multimodal* prior to test. Of the 1394 total queries we built models for, 51.4% saw an increase in validation NDCG by utilizing the multimodal representation over the baseline representation. We present results in Section 3.3.

3.3 Results

Here we present and discuss the results of incorporating image information into our ranking models. Table 1 summarizes the results of our experiment for queries that moved from a text-only representation to a multimodal one. For each modality, we present the average lift in NDCG over our baseline text-only models. We find that consistent with our expectations, switching from a text-only ranking modality to one that uses image information as well (MM) yields a statistically significant **1.7% gain in average ranking quality**. We find it interesting that a purely image-based representation, while strictly worse than both other modalities, only underperforms the baseline representation by 2.2%.

Table 1: Lift in Average NDCG, relative to baseline (%), on sample dataset is compared across various modalities. * indicates the change over baseline method is statistically significant according to Wilcoxon signed rank test at the significance level of 0.0001.

Modality	Text	Image	MM
Relative lift in NDCG	+0.0%	−2.2%*	**+1.7%***

We can explore how incorporating image information changes the ranking by looking at a continuum visualization of how the two modality models rank listings for a query: The top row shows the top 90th percentile of ranked listings, the bottom row shows the 50th percentile of ranked listings, and every row in between is a continuum. Figure 6 illustrates a continuum visualization for the query bar "bar necklace". We observe that by moving from a text-only to a multimodal representation, the top ranked listings in each

[5]We collect implicit relevance judgments in the test period the same as in training: documents are labeled 0.0 if ignored and 1.0 if the user purchased the listing, added the listing to their cart, or clicked on the listing and dwelled for more than 30 seconds.

Figure 6: **Visualizing ranking changes by incorporating image information:** Here we visualize how moving from a text-only representation to a multimodal one impacted the ranking for the query “bar necklace”. Each row shows the top ranked listings for the query by percentile, from 90^{th} percentile on the top to 50^{th} percentile on the bottom. We observe that the images in each row show more visual consistency under a multimodal ranking than a text-only ranking. That, along with the fact that the query saw a 6.62% increase in ranking quality (NDCG) by moving to a multimodal representation, suggests that the multimodal representation captures relevant visual style information not available in the text.

band show greater visual consistency. This suggests that there is *relevant style information* captured in the image representation that is not available, or not well described in the text representation. Incorporating this complementary side information leads to a $\mathbf{6.62\%}$ **increase** in offline NDCG.

We can also see concrete examples of the image representation providing valuable complementary information to the ranking function. Figure 5 shows three examples of listings that were ranked similarly under a text model and very differently under a multimodal model. For example, Figure 5 (a) shows two listings that match the query “wolf". It is apparent by looking at the two listing images that only the right listing is relevant to the query: the left shows a picture of a rhino; the right is a t-shirt with a wolf on it. The text-only model ranked these two listings only two positions apart. In contrast, the multimodal model ranked the relevant right-hand listing 394 positions higher. We can see by looking at the bolded terms in common how the text model could be confused: as points embedded in text space, these listings are essentially on top of each other. It is clear that the images contain enough information to differentiate them: one contains a wolf, one contains a rhino. By embedding listings in a space that preserves this difference, the multimodal model can effectively disentangle these two listings and provide a more accurate ranking. The query “wolf" saw an overall $\mathbf{3.07\%}$ **increase** in NDCG by moving to a multimodal representation. Similarly for Figure 5 (b), a text model for the query “leather bag" ranked the two listings 1 position apart, while the multimodal model ranked the right listing 660 positions higher. “leather bag" saw a $\mathbf{2.56\%}$ **increase** in NDCG by moving to a multimodal representation. For Figure 5 (c) a text model for the query “wedding band" ranked the left and right listing 4 positions apart, while the multimodal model ranked the right listing 427 positions higher. “wedding band" saw a $\mathbf{1.55\%}$ **increase** in NDCG by moving to a multimodal representation.

4. CONCLUSION

Learning to rank search results is one of Etsy and other e-commerce sites’ most fundamental problems. In this paper we describe how deep visual semantic features can be transferred successfully to multimodal learning to rank framework. We verify in a large-scale experiment that there is indeed significant complementary information present in images that can be used to improve search ranking quality. We visualize concrete examples of this marginal value: (i) the ability of the image modality to capture style information not well described in text. (ii) The ability of the image modality to disentangle highly different listings considered similar under a text-only modality.

Acknowledgments

The authors would like to thank Arjun Raj Rajanna, Christopher Kanan and Robert Hall for fruitful discussions during the course of this paper. We would also like to thank Ryan Frantz and Will Gallego for their insight and help in building the infrastructure required for running our experiments.

5. REFERENCES

[1] AYTAR, Y., AND ZISSERMAN, A. Tabula rasa: Model transfer for object category detection. In *Computer Vision (ICCV), 2011 IEEE International Conference on* (2011), IEEE, pp. 2252–2259.

[2] BAI, B., WESTON, J., GRANGIER, D., COLLOBERT, R., SADAMASA, K., QI, Y., CHAPELLE, O., AND WEINBERGER, K. Learning to rank with (a lot of) word features. *Information retrieval 13*, 3 (2010), 291–314.

[3] BURGES, C., SHAKED, T., RENSHAW, E., LAZIER, A., DEEDS, M., HAMILTON, N., AND HULLENDER, G. Learning to rank using gradient descent. In *Proceedings of the 22nd international conference on Machine learning* (2005), ACM, pp. 89–96.

[4] CHATFIELD, K., SIMONYAN, K., VEDALDI, A., AND ZISSERMAN, A. Return of the devil in the details: Delving deep into convolutional nets. *arXiv preprint arXiv:1405.3531* (2014).

[5] FROME, A., CORRADO, G. S., SHLENS, J., BENGIO, S., DEAN, J., MIKOLOV, T., ET AL. Devise: A deep visual-semantic embedding model. In *Advances in Neural Information Processing Systems* (2013), pp. 2121–2129.

[6] GATYS, L. A., ECKER, A. S., AND BETHGE, M. Texture synthesis and the controlled generation of natural stimuli using convolutional neural networks. *arXiv preprint arXiv:1505.07376* (2015).

[7] GONG, Y., WANG, L., HODOSH, M., HOCKENMAIER, J., AND LAZEBNIK, S. Improving image-sentence embeddings using large weakly annotated photo collections. In *Computer Vision–ECCV 2014*. Springer, 2014, pp. 529–545.

[8] GUILLAUMIN, M., VERBEEK, J., AND SCHMID, C. Multimodal semi-supervised learning for image classification. In *CVPR 2010-23rd IEEE Conference on Computer Vision & Pattern Recognition* (2010), IEEE Computer Society, pp. 902–909.

[9] HANG, L. A short introduction to learning to rank. *IEICE TRANSACTIONS on Information and Systems 94*, 10 (2011), 1854–1862.

[10] HERBRICH, R., GRAEPEL, T., AND OBERMAYER, K. Large margin rank boundaries for ordinal regression. *Advances in neural information processing systems* (1999), 115–132.

[11] JÄRVELIN, K., AND KEKÄLÄINEN, J. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems (TOIS) 20*, 4 (2002), 422–446.

[12] JOACHIMS, T. Optimizing search engines using clickthrough data. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining* (2002), ACM, pp. 133–142.

[13] KANNAN, A., TALUKDAR, P. P., RASIWASIA, N., AND KE, Q. Improving product classification using images. In *Data Mining (ICDM), 2011 IEEE 11th International Conference on* (2011), IEEE, pp. 310–319.

[14] KARPATHY, A., JOULIN, A., AND LI, F. F. F. Deep fragment embeddings for bidirectional image sentence mapping. In *Advances in neural information processing systems* (2014), pp. 1889–1897.

[15] KIROS, R., SALAKHUTDINOV, R., AND ZEMEL, R. Multimodal neural language models. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)* (2014), pp. 595–603.

[16] KRIZHEVSKY, A., SUTSKEVER, I., AND HINTON, G. E. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (2012), pp. 1097–1105.

[17] MIKOLOV, T., SUTSKEVER, I., CHEN, K., CORRADO, G. S., AND DEAN, J. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems* (2013), pp. 3111–3119.

[18] OQUAB, M., BOTTOU, L., LAPTEV, I., AND SIVIC, J. Learning and transferring mid-level image representations using convolutional neural networks. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on* (2014), IEEE, pp. 1717–1724.

[19] PAN, S. J., AND YANG, Q. A survey on transfer learning. *Knowledge and Data Engineering, IEEE Transactions on 22*, 10 (2010), 1345–1359.

[20] PEREIRA, J. C., AND VASCONCELOS, N. On the regularization of image semantics by modal expansion. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on* (2012), IEEE, pp. 3093–3099.

[21] RADLINSKI, F., AND JOACHIMS, T. Minimally invasive randomization for collecting unbiased preferences from clickthrough logs. In *Proceedings of the National Conference on Artificial Intelligence* (2006), vol. 21, Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, p. 1406.

[22] RAZAVIAN, A., AZIZPOUR, H., SULLIVAN, J., AND CARLSSON, S. Cnn features off-the-shelf: an astounding baseline for recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops* (2014), pp. 806–813.

[23] SIMONYAN, K., AND ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).

[24] SRIVASTAVA, N., HINTON, G., KRIZHEVSKY, A., SUTSKEVER, I., AND SALAKHUTDINOV, R. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research 15*, 1 (2014), 1929–1958.

[25] TOMMASI, T., ORABONA, F., AND CAPUTO, B. Learning categories from few examples with multi model knowledge transfer. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 36*, 5 (2014), 928–941.

[26] WANG, G., HOIEM, D., AND FORSYTH, D. Building text features for object image classification. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on* (2009), IEEE, pp. 1367–1374.

[27] WEINBERGER, K., DASGUPTA, A., LANGFORD, J., SMOLA, A., AND ATTENBERG, J. Feature hashing for large scale multitask learning. In *Proceedings of the 26th Annual International Conference on Machine Learning* (2009), ACM, pp. 1113–1120.

[28] WESTON, J., BENGIO, S., AND USUNIER, N. Large scale image annotation: learning to rank with joint word-image embeddings. *Machine learning 81*, 1 (2010), 21–35.

[29] ZEILER, M. D., AND FERGUS, R. Visualizing and understanding convolutional networks. In *Computer Vision–ECCV 2014*. Springer, 2014, pp. 818–833.