

eCommerce Product Search Relevance using Text and Image Embeddings

Janardhana Swamy Adapa
janardhan_adapa@tamu.edu
Texas A&M University

Nikita Duseja
nduseja@tamu.edu
Texas A&M University

Andrew Bregger
andrewbregger@tamu.edu
Texas A&M University

Sai Eswar Epuri
saieswar@tamu.edu
Texas A&M University

ABSTRACT

Traditional query-document relevance prediction systems rely extensively on hand crafted features like distance features, counting features and vector space similarity metrics. In our project, we propose a neural learning method to use pre-trained distributed vector representations of product descriptions and product image embeddings to predict the relevance of the product given the query. We have tested the methods on a search relevance dataset in the eCommerce setting.

1 INTRODUCTION

Predicting search relevance and learning to rank is a fundamental and challenging task in the field of information retrieval. It has applications in several domains like product recommendations, eCommerce product search and blog recommendation systems. In the absence of any implicit feedback like click-through data, it becomes crucial to have robust relevance prediction and ranking algorithms for **ad-hoc** query-content matching.

Traditional approaches rely extensively on counting measures: TF-IDF and probabilistic model, BM25. The drawback with these counting measures is that they do not weigh the non-query terms in the documents. However, there may be several instances to have sufficient presence of query terms in document, without being relevant. This gives way to feature engineering, which is an extremely challenging and laborious task. We build upon this drawback to develop a neural model to learn a similarity metric between queries and products in the eCommerce setting. Instead of using distance metrics and count features, we use distributed vector representations of queries and product descriptors to build a model for generating query relevance when given the query, product description, and product image.

Our goal in this project was to create a neural learning model to accurately predict the relevance of search results against the given query. Given the queries, resulting product descriptions and images from leading eCommerce sites like Walmart, eBay and HomeDepot, we have developed different models to use both the image information and text information to construct a relevance prediction function. We build upon the work of [2] to develop a regression models for query-product relevance.

Search in the context of eCommerce benefits from the abundance of visual data. We build on the product images to enhance the inputs to the model using visual cues of the product. Our main goal in this project is to analyse if eCommerce search can benefit from

visual inputs. We build our experiments on text, image and joint representations embeddings and report our results on the Kaggle eCommerce search relevance dataset.

Section 2 of the report deals with the related work and literature review of the deep learning methodologies that are commonly used in the query-content matching problems. In section 3 we describe the dataset that we have used for implementing our proposed method. In section 4 and 5 we describe the various models we built for relevance prediction(regression) and the results we observed. In section 6 we conclude with the challenges we faced and the scope of future work.

2 RELATED WORK

Query-content matching has been studied extensively and has evolved from traditional count based measures, to statistical and language dependent modeling[10]. Traditional approaches in the query document relevance and ranking, **TF-IDF** were studied mainly in [8]. One breakthrough in the count based measures was the introduction of BM25, which still provides results on par with feature engineering and neural methods.

Deep learning applications to information retrieval have led to several **representation** based neural models for query document ranking. [7] propose a latent semantic model to capture contextual representations in queries and documentations with focus to improve the ranking performance in learning to rank(**LTR**) tasks. [3] **interaction** based deep learning architectures for document relevance ranking, building on query and document term based interaction models. They propose scoring query terms for documents and then aggregating several query scores to predict the overall relevance score of the document.

3 DATASET

The eCommerce search relevance dataset we use in this project was provided by CrowdFlower [1]. The dataset contains 32,671 entries of product, query information: webpage, product image, product description, query, relevance, and rank. There are 264 unique queries with an average of 124 products, a maximum of 298 products and a minimum of 1 product for a given query. Several entries in the dataset were missing the product image or descriptions both of which were needed for our proposed methods. This resulted in many being discarded and finally it resulted in 7203 entries.

Queries The dataset had 261 query samples. Since the query text is generally brief and concise, we made it more descriptive using synonym based query expansion. On exploration we found some

spellcheck issues, hence we also preprocessed the query tokens using spelling correction techniques.

Product Description Text: The dataset had verbose and noisy product descriptions, with HTML tags, hyperlinks and numbers, hence we preprocessed the dataset using libraries, BeautifulSoup.

Relevance Scores The relevance score of each product against a query ranges from 1 to 4 and are curated from the mean of 3 human evaluators. This is our golden truth to evaluate the model upon.



Query	Product	Description
16 gb memory card		The Lexar 16GB MicroSD Memory Card with SD Adaptor allows you to use the same MicroSD card...
Lenovo Laptop		This Lenovo ThinkPad T420 Laptop comes with Intel Core i5-520M 2.5GHz processor, 4096MB DDR3 memory, 750GB hard drive...

Figure 1: Sample instances from the Kaggle eCommerce Search relevance dataset. Our key inputs from the data were the product queries, the product descriptions and product images.

4 MODELS

4.1 Search relevance using Query and Product Description Embeddings

In this model, only text embeddings that are obtained from product descriptions are used to train our neural network model. Both Glove [5] and Doc2Vec [6] from genism are used to generate embeddings for our product description and the query. Both the methods used are pretrained methods as we wanted to increase the effectiveness of the vector space that is obtained through this embeddings. Another reason we used a pretrained model was because our dataset was not large enough to generate an effective word embeddings. After obtaining the embeddings to both query and product description, we process them through dense layers and then concatenate both these representations and pass through a fully connected Neural network to obtain the relevance. This model is the base model as it uses just the product descriptions to obtain the relevance.

4.2 Generating Image Embeddings

We have generated Image embeddings by two ways.

Pretrained Image Embeddings The first way is by predicting the product image using a pre-trained InceptionV3 model trained on the ImageNet Dataset [9]. We have generated the 1000 dimensional output vector of InceptionV3 model for each image, and used it as the image embedding in the upcoming models. Since the ImageNet contains millions of images, InceptionV3 is expected to extract reasonably good embeddings from the product images in our dataset.

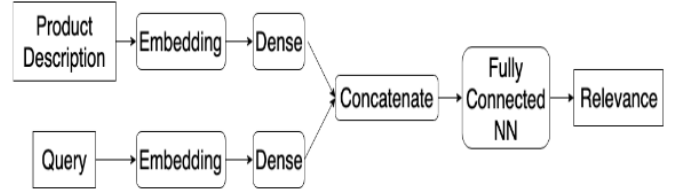


Figure 2: Baseline Model 1 : Neural regression model using product description and query text embeddings. The inputs to the model are the pretrained/trained distributed vector representations and the output is the relevance score on the range(1-4) from the dataset

Image2Text Embeddings-Joint Representation The second way we generated image embeddings is by training a fully connected neural network as shown in Figure 3. This neural network contains five dense layers, in which the first dense layer takes the as input the 1000-dimensional image embedding obtained from InceptionV3, whereas the output of this model is an 100-dimension product description embedding obtained for each product in section 4.1. This neural network generates word embedding given the image embedding. The intuition is that by generating captions given a product image, and using this caption in addition to the product description from the dataset can improve the prediction of search relevance. But since we lack data to train an RNN model to generate captions for an image, we are using the word embeddings obtained from Glove model which is pre-trained on very large text corpus and will produce a reasonably good text embedding given the product description from our dataset. The other advantage of using this method is the training time is around 5 minutes, whereas the traditional method of training a RNN to generates captions in the form of text will take around one to two days to train on our dataset.

Another way to look at this is that the output of this neural network is the transformation of image embedding into the product description textembedding, and this output contains the information from both the image and description of a product. In the rest of the document, this output is referred as **Image to descriptive embedding (Img2Desc)**.

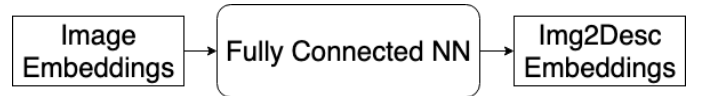


Figure 3: Generating Image to descriptive embedding from Image embedding using a fully connected neural network: A fully connected dense network used for transforming pre-trained product image embeddings to product description embeddings. This model yields a joint representation of visual and text data :Image2Desc Embeddings

4.3 Search relevance using Image Embeddings

In this model, we have tried using Image embeddings obtained directly from the InceptionV3 model as well as Img2Desc embeddings obtained by using the method mentioned in the previous section. We use query embeddings along with Image embeddings and pass them through dense layers and concatenate them in similar fashion as the first method to predict the relevance scores of the product given a query. This model can be used with either pure image embeddings or the Img2Desc embeddings. When using only image embeddings, the model is only using information from the product image. This is limiting because, most of the time, we are unable to gather detailed information about product such as electronic specifications or clothing material. Because of the Img2Desc embeddings, this model uses both image data and textual data from the descriptions when deciding on a relevance score.

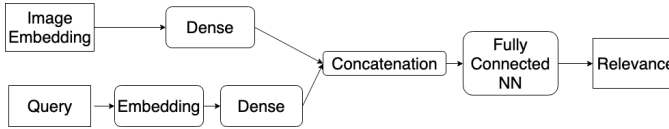


Figure 4: Baseline Model 2 : Neural regression model using pretrained product image embeddings and query text embeddings. The inputs to the model are the pretrained/trained distributed vector representations of the image and query and the output is the relevance score on the range(1-4) from the dataset.

4.4 Search relevance using Image and Description Embeddings

Using both product description embeddings obtained from section 4.1 and image embeddings obtained from section 4.2, search relevance of the product is predicted. The architecture of the model is shown in Figure 5. The query and product description are sent to the embedding layers which are frozen during training, so we are obtaining embeddings from the query and description and are sent to the dense layer. Here, the image embeddings are down-sampled from initial 1000-dimension vector to 300-dimension vector using two dense layers. We are doing this to prevent a bottleneck which might cause loss of information of the image embedding. As shown in figure 3, the output of all three dense layers corresponding to query, product description, and image embeddings are concatenated and sent to a 5-layer fully connected neural network which predicts the relevance of the product w.r.t the given query. Since we are using both image and description of the product, we are using all the information that we have on the product and by using this information, model will try to predict the search relevance for the product with the given search query.

Here, we have used two kinds of image embeddings as input to train two different models. These two image embeddings are obtained using the methods given in section 4.2.

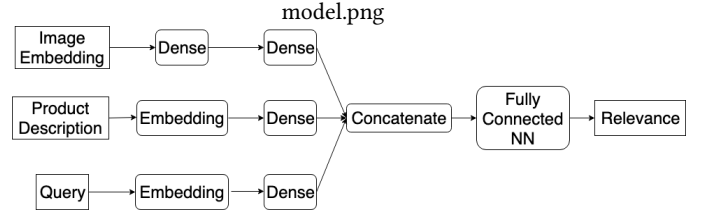


Figure 5: Neural regression model using product description, product image and query text embeddings. The inputs to the model are the pretrained/trained distributed vector representations of each input descriptor and the output is the relevance score on the range(1-4) from the dataset

5 EXPERIMENTS AND RESULTS

Evaluation metrics that we used to evaluate our models were **Root Mean Squared Error(RMSE)** and **Normalized Discounted Cumulative Gain(NDCG)**. RMSE gives the root mean squared error between the predicted relevance and the actual relevance on the test data. RMSE gives a good measure how far our predicted relevance is from the actual relevance. We have calculated both NDCG@5 and NDCG@10 to get a good measure of how the relevance scores of top 5 and 10 relevant pages to a query are getting effected.

From Figure 6, we can observe that **Img2Desc Embeddings + Query Embeddings (ID-Q) is the best model in terms of RMSE**. We can also observe that NDCG results are not consistent with RMSE as NDCG considers only relative ordering of top relevant products whereas RMSE takes into consideration all the products.

The high values of NDCG for all the models can be attributed to our models behaviour of predicting more relevant products more accurately compared to less relevant ones. The possible reason for ID-Q model to perform better than remaining models can be attributed to Img2Desc embeddings containing combined information of both text and image information of the product.

6 CONCLUSION

6.1 Challenges

One of our challenges was the limitations of dataset. As said previously, the original dataset contain 32,671 samples. Due the dataset being relatively old, some of the images were not available, since our method required images we had to disregard all entries with an invalid image URL. Also, there were many missing product descriptions which were also needed, these samples were also disregarded. This left only a fraction of the original dataset. In general the images that were left were good for our purposes; however, there were some descriptions that are verbose. Much of which were not beneficial to the generated models. We also has the opposite problem as well. Some of the descriptions were very short with not enough information to be able to match anything other than generic question.

6.2 Summary

The main goal of our work was to verify whether including image information into IR models dealing with eCommerce improves the

Methods	RMSE	NDCG@5	NDCG@10
Description Embeddings + Query Embeddings (D-Q)	0.76	0.9564	0.9748
Product Image Embeddings + Query Embeddings (I-Q)	0.74	0.9895	0.9939
Img2Desc Embeddings + Query Embeddings (ID-Q)	0.67	0.9657	0.9800
Description Embeddings + Image Embeddings + Query Embeddings (D-I-Q)	0.77	0.9500	0.9710
Description Embeddings + Img2Desc Embeddings + Query Embeddings (D-ID-Q)	0.74	0.9596	0.9764

Table 1: Summary of the results: We evaluated our model on several metrics, the key metric being RMSE from the actual relevance scores in the dataset. The table summarizes the RMSE, NDCG@5 and NDCG@10 metrics on the test split of the dataset

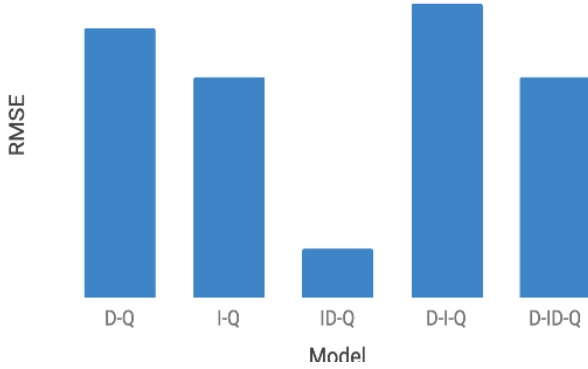


Figure 6: The RMSE scores for various models are summarized above on various models : Description Embeddings + Query Embeddings (D-Q), Product Image Embeddings + Query Embeddings (I-Q), Img2Desc Embeddings + Query Embeddings (ID-Q), Description Embeddings + Image Embeddings + Query Embeddings (D-I-Q) and Description Embeddings + Img2Desc Embeddings + Query Embeddings (D-ID-Q)

efficiency of models. We came up with five different combinations along with query embeddings to evaluate the results. Interestingly, we found that by taking RMSE as a evaluation measure Img2Desc embeddings gives the best results when compared with the base model which only considers text information. This opens up new space in the building of IR systems which use combined features of text and product images for predicting the relevance scores of product given a query.

6.3 Future Work

A continuance of this would require a larger, complete data set. From our current dataset, we learned that e-commerce data such as images and web-pages to not last for long. With a larger dataset we would be able to employ different techniques such as Dual Embedding Vector Space to more actually express the problem [4]. This technique is useful because it splits the embedding space of the query and documents into two separate vector spaces. The query and document have different semantics, as query is usually a short phrase or question while the documents are complete, grammatically correct sentences and paragraphs.

With the larger dataset, we would also explore caption generation of the products images. The generated caption could be used to

complement or supplement the description text. Our current work finds an embedding of the image features in the description text, this is a complementary approach which would generate the actual text of a caption instead of a text embedding.

REFERENCES

- [1] Crowdfunder. 2015. eCommerce search relevance. <https://data.world/crowdfunder/e-commerce-search-relevance>
- [2] Ruoxuan Xiong Luyang Chen. 2015. Predict the Relevance of Search Results on Homedepot.com. (2015).
- [3] Ryan McDonald, Georgios-Ioannis Brokos, and Ion Androutsopoulos. 2018. Deep relevance ranking using enhanced document-query interactions. *arXiv preprint arXiv:1809.01682* (2018).
- [4] Bhaskar Mitra, Eric Nalisnick, Nick Craswell, and Rich Caruana. 2016. A dual embedding space model for document ranking. *arXiv preprint arXiv:1602.01137* (2016).
- [5] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation. In *Empirical Methods in Natural Language Processing (EMNLP)*. 1532–1543. <http://www.aclweb.org/anthology/D14-1162>
- [6] Radim Rehůrek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. ELRA, Valletta, Malta, 45–50. <http://is.muni.cz/publication/884893/en>.
- [7] Yelong Shen, Xiaodong He, Jianfeng Gao, Li Deng, and Grégoire Mesnil. 2014. A latent semantic model with convolutional-pooling structure for information retrieval. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*. ACM, 101–110.
- [8] Karen Sparck Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation* 28, 1 (1972), 11–21.
- [9] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2818–2826.
- [10] Chengxiang Zhai and John Lafferty. 2017. A study of smoothing methods for language models applied to ad hoc information retrieval. In *ACM SIGIR Forum*, Vol. 51. ACM, 268–276.