The Thirtieth AAAI Conference on Innovative Applications of Artificial Intelligence (IAAI-18)

Is a Picture Worth a Thousand Words? A Deep Multi-Modal Architecture for Product Classification in E-Commerce

Tom Zahavy,[1] Abhinandan Krishnan,[2] Alessandro Magnani,[2] Shie Mannor[1]
[1] The Technion - Israel Institute of Technology, Haifa, Israel
[2] Walmart Labs, Sunnyvale, California

Abstract

Classifying products precisely and efficiently is a major challenge in modern e-commerce. The high traffic of new products uploaded daily and the dynamic nature of the categories raise the need for machine learning models that can reduce the cost and time of human editors. In this paper, we propose a decision level fusion approach for multi-modal product classification based on text and image neural network classifiers. We train input specific state-of-the-art deep neural networks for each input source, show the potential of forging them together into a multi-modal architecture and train a novel policy network that learns to choose between them. Finally, we demonstrate that our multi-modal network improves classification accuracy over both networks on a real-world large-scale product classification dataset that we collected from Walmart.com. While we focus on image-text fusion that characterizes e-commerce businesses, our algorithms can be easily applied to other modalities such as audio, video, physical sensors, etc.

Introduction

Product classification is a key issue in e-commerce businesses. A product is typically represented by metadata such as its title, image, color, weight and so on, and most of them are assigned manually by the seller. Once a product is uploaded to an e-commerce website, it is typically placed in multiple categories. Categorizing products helps e-commerce websites to provide costumers with a better shopping experience, for example by efficiently searching the products catalog or by developing recommendation systems. A few examples of categories are internal taxonomies (for business needs), public taxonomies (such as groceries and office equipment) and the product's shelf (a group of products that are presented together on an e-commerce web page). These categories vary with time to optimize search efficiency and to account for special events such as holidays and sports events. To address these needs, e-commerce websites typically hire editors and use crowdsourcing platforms to classify products. However, due to the high amount of new products uploaded daily and the dynamic nature of the categories, machine learning solutions for product classification are appealing as means to reduce the time needed to classify products as well as cost. Thus, precisely categorizing items emerge as a significant issue in e-commerce domains.

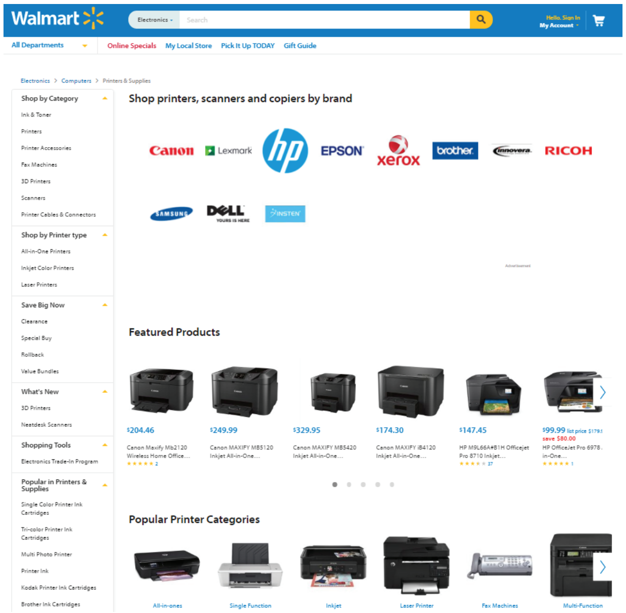

Figure 1: The printers & supplies shelf from Walmart.com. Specific products (printers in this case) are presented using meta data such as image, title, and price.

In this paper, we refer to a **shelf** as a group of products presented together on an e-commerce website page and usually contain products with a given category (see Figure 1 for an example of the printers&supplies shelf on Walmart.com). Product to shelf classification is a challenging problem due to data size, category skewness, and noisy metadata and labels. In particular, it presents three fundamental challenges for machine learning algorithms. First, it is typically a multi-class problem with thousands of classes. Second, a product may belong to multiple shelves making it a multi-label problem. And last, a product has both an image and a text input making it a multi-modal problem.

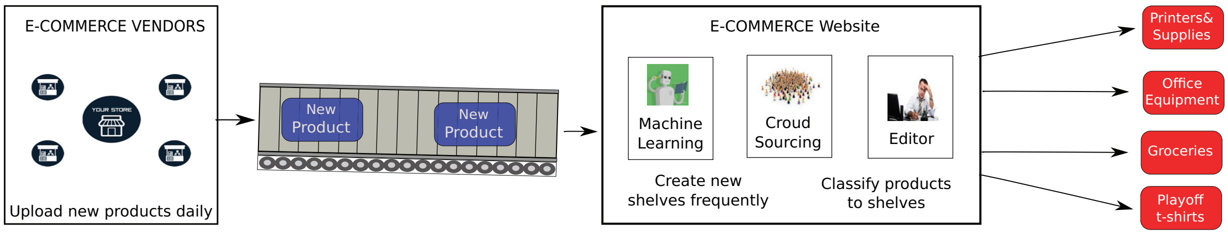

Figure 2: E-commerce classification diagram.

Products classification is typically addressed as a text classification problem because most metadata of items are represented as textual features (Pyo, Ha, and Kim 2010). Text classification is a classic topic for natural language processing, in which one needs to assign predefined categories to text inputs. Standard methods follow a classical two-stage scheme of extraction of (handcrafted) features, followed by a classification stage. Typical features include bag-of-words or n-grams, and their TF-IDF. On the other hand, Deep Neural Networks use generic priors instead of specific domain knowledge (Bengio, Courville, and Vincent 2013) and have been shown to give competitive results on text classification tasks (Zhang, Zhao, and LeCun 2015). In particular, Convolutional neural networks (CNNs) (Kim 2014; Zhang, Zhao, and LeCun 2015; Conneau et al. 2016) and Recurrent NNs (Lai et al. 2015; Pyo, Ha, and Kim 2010; Xiao and Cho 2016) can efficiently capture the sequentiality of the text. These methods are typically applied directly to the distributed embedding of words (Kim 2014; Lai et al. 2015; Pyo, Ha, and Kim 2010) or characters (Zhang, Zhao, and LeCun 2015; Conneau et al. 2016; Xiao and Cho 2016), without any knowledge on the syntactic or semantic structures of a language. However, all of these architectures were only applied on problems with a few labels ($\sim$ 20) while e-commerce shelf classification problems typically have thousands of labels with multiple labels per product.

In Image classification, CNNs are widely considered the best models, and achieve state-of-the-art results on the ImageNet Large-Scale Visual Recognition Challenge (Russakovsky et al. 2015; Krizhevsky, Sutskever, and Hinton 2012; Simonyan and Zisserman 2014; He et al. 2015). However, as good as they are, the classification accuracy of machine learning systems is often limited in problems with many classes of object categories. One remedy is to leverage data from other sources, such as text data. However, the studies on multi-modal deep learning for large-scale item categorization are still rare to the best of our knowledge.

In this work, we propose a multi-modal deep neural network for product classification. Our design principle is to leverage the specific prior for each data type by using the current state-of-the-art classifiers from the image and text domains. The final architecture has 3 main components: a text CNN (Kim 2014), an image CNN (Simonyan and Zisserman 2014) and a **deep policy network** that learns to choose between them. We collected a large-scale data set of 1.2 million products from the Walmart.com website. Each product has a title and an image and needs to be classified to a shelf (label) with 2890 possible shelves. Examples from this dataset can be seen in Figure 3 and are also available online at the Walmart.com website. For most of the products, both the image and the title of each product contain relevant information for customers. However, it is interesting to observe that for some of the products; both input types may not be informative for shelf prediction (See Figure 3 for examples). This observation motivates our work and raises interesting questions: which input type is more useful for product classification? Is it possible to forge the inputs into a better architecture?

Our experiments suggest that the text information is more informative than the images for shelf classification. However, for a relatively large number of products ($\sim 8\%$), the image CNN is correct while the text CNN is wrong, indicating a potential gain from using a multi-modal architecture. We also show that we can train a deep policy to choose between the two models and give a performance improvement over both state-of-the-art networks.

To the best of our knowledge, this is the first work that achieves a performance improvement on classification accuracy by using multi-modality on a large-scale classification problem (see Table 1 for more details). In particular, our main contributions are: (1) We solve a challenging real-world **e-commerce classification** problem using state-of-the-art CNNs and demonstrate that text-based classification is better than image-based for this dataset. (2) We propose a new algorithm that **learns a decision fusion rule** using a deep network. And (3), we demonstrate that our multi-modal architecture **improves classification accuracy** over both input-specific networks.

Multi-Modality

Over the years, a large body of research has been devoted to improving classification using ensembles of classifiers (Kittler et al. 1998; Hansen and Salamon 1990). Inspired by their success, these methods have also been used in multi-modal settings, e.g., (Guillaumin, Verbeek, and Schmid 2010; Poria et al. 2016), where the source of the signals, or their modal-

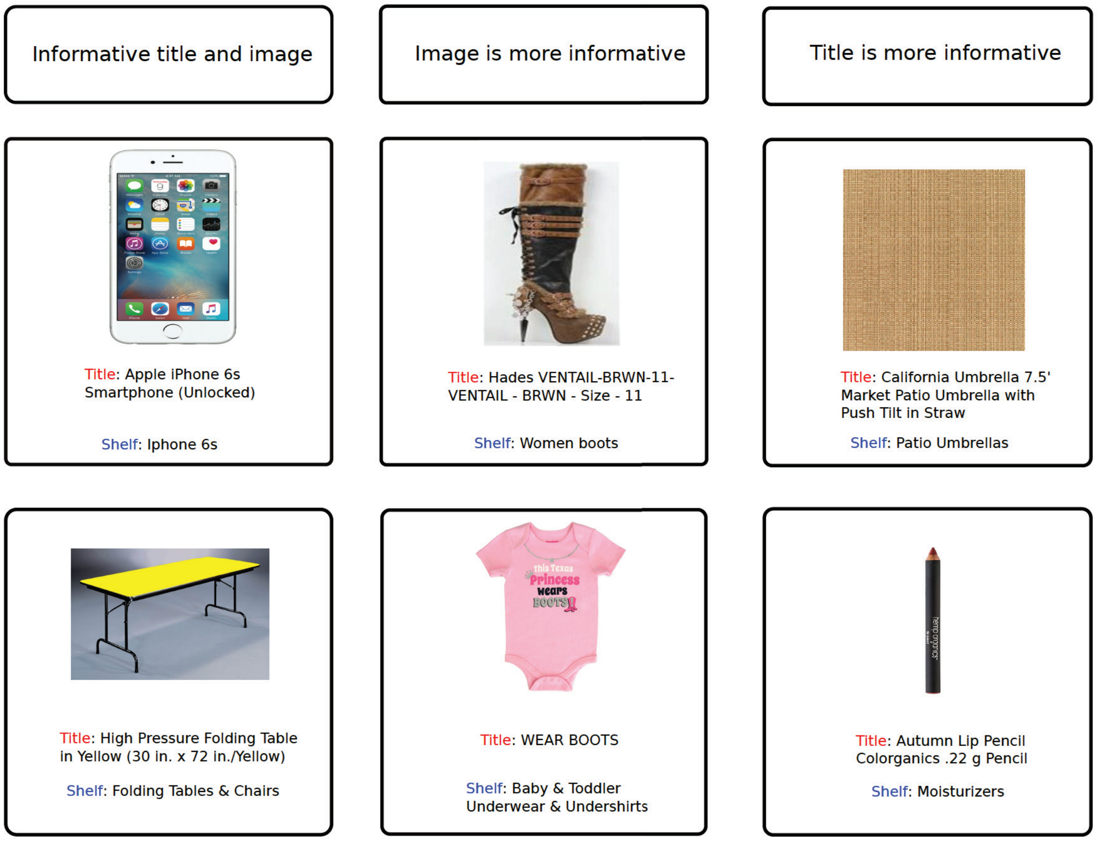
Informative title and image
Image is more informative
Title is more informative
Title: Apple iPhone 6s Smartphone (Unlocked)
Shelf: Iphone 6s
Title: Hades VENTAIL-BRWN-11-VENTAIL - BRWN - Size - 11
Shelf: Women boots
Title: California Umbrella 7.5' Market Patio Umbrella with Push Tilt in Straw
Shelf: Patio Umbrellas
Title: High Pressure Folding Table in Yellow (30 in. x 72 in./Yellow)
Shelf: Folding Tables & Chairs
this Texas Princess wears BOOTS
Title: WEAR BOOTS
Shelf: Baby & Toddler Underwear & Undershirts
Title: Autumn Lip Pencil Colorganics .22 g Pencil
Shelf: Moisturizers

Figure 3: Shelves from Walmart.com. *Left:* a product that has both an image and a title that contain useful information for predicting the product's shelf. *Center, top:* the boots title gives specific information about the boots but does not mention that the product is a boot, making it harder to predict the shelf. *Center, bottom:* the baby toddler shirt's title only refers to the text on the toddler shirt and does not mention that it is a product for babies. *Right, top:* the umbrella image contains information about its color, but it is hard to understand that the image is referring to an umbrella. *Right, bottom:* the lips pencil image looks like a regular pencil, making it hard to predict that it belongs to the moisturizers shelf.

ities, are different. Some examples include audio-visual speech classification (Ngiam et al. 2011), image and text retrieval (Kiros, Salakhutdinov, and Zemel), sentiment analysis and semi-supervised learning (Guillaumin, Verbeek, and Schmid 2010).

Combining classifiers from different input sources presents multiple challenges. First, classifiers vary in their discriminative power; thus, an optimal unification method should be able to adapt itself for specific combinations of classifiers. Second, different data sources have different state-of-the-art architectures, typically deep neural networks, which vary in depth, width, and optimization algorithm; making it non-trivial to merge them. Moreover, a multi-modal design potentially has more local minima that may give unsatisfying results. Finally, most of the publicly available real-world big data classification datasets, an essential building block of deep learning systems, typically contain only one data type.

Nevertheless, the potential performance boost of multi-modal architectures has motivated researchers over the years. Frome et al. 2013, combined an image network (Krizhevsky, Sutskever, and Hinton 2012) with a Skip-gram Language Model in order to improve classification results on ImageNet. However, they were not able to improve the accuracy prediction, possibly because the text input they used (image labels) didn't contain a lot of information. Other works, used multi-modality to learn good embedding but did not present results on classification benchmarks (Lynch, Aryafar, and Attenberg 2015; Kiros, Salakhutdinov, and Zemel ; Gong et al. 2014). Kannan et al. 2011, suggested to improve text-based product classification by adding an image signal, training an image classifier and learning a decision rule between the two. However, they only experimented with a small dataset and a low number of labels, and it is not clear how to scale their method for extreme multi-class multi-label applications that characterize real-world problems in e-commerce. Table 1 summarizes the contribution of prior works on multi-modal classification. The Table implies that our work is the first that shows improvement in classification accuracy on a large scale dataset.

	Data size	# labels	Multi-Modality	Classification improvement
Krizhevsky, Sutskever, and Hinton	1M	1K	✗[1]	✓
Pyo, Ha, and Kim	5M	500	✗[2]	✓
Poria et al.	30k	100	✓[3]	✓
Ngiam et al.	10k	100	✓	✗[4]
Kannan et al.	30K	17	✓	✓
Frome et al.	1M	1K	✓	✗[5]
Ours	**1M**	**3K**	✓	✓

Table 1: Previous works on multi-modal classification. Comments: 1-2 use pre-defined decision level fusion rules. 1 and 2 do fusion from the same modality. 4,5 show improvement in classification but not on accuracy (4 on a noisier test set and 5 on hierarchical accuracy).

Most unification techniques for multi-modal learning are partitioned between feature-level fusion techniques and decision-level fusion techniques (Figure 4, left).

Decision-level fusion. In this approach, an input-specific classifier is learned for each modality, and the goal is to find a decision rule that selects one from them. The decision rule is typically a pre-defined rule (Guillaumin, Verbeek, and Schmid 2010) and is not learned from the data. For example, Poria et al. 2016, chose the classifier with the maximal confidence, while Krizhevsky, Sutskever, and Hinton 2012, average classifier predictions. However, in our setting, there is a significant difference in discriminative power between the text and image networks. Thus, pre-defined rules suffer from bias and do not perform well. In this work, we suggest to solve this problem by **learning** the decision rule with a deep neural network and demonstrate that it yields significantly better results on our data.

Feature level fusion. In the deep learning context, there are two standard approaches. In the first method, we learn an **end-to-end** deep NN; the NN has multiple input-specific pipes that include a data source followed by input-specific layers. After a certain depth, the pipes are concatenated followed by additional layers such that the NN is trained end-to-end. In the second approach, **step-by-step**, input specific deep NNs are learned first, and a multi-modal representation vector is created by concatenating the data specific feature vectors (e.g., the neural network's last hidden layer). Then, an additional classifier learns to classify from the multi-modal representation vector.

Methods and architectures

Multi label cost function. We use the weighted sigmoid cross entropy with logits, a common cost function for multi-label problems. Let x be the logits, z be the targets, q be a positive weight coefficient, used as a multiplier for the positive targets, and $\sigma(x) = \frac{1}{1+exp(-x)}$. The loss is given by:

$$\text{Cost(x,z;q)} = -qz \cdot \log(\sigma(x)) - (1-z) \cdot \log(1-\sigma(x)) =$$
$$(1-z) \cdot x + (1 + (q-1) \cdot z) \cdot \log(1 + exp(-x)).$$

The positive coefficient q, allows one to trade off recall and precision by up- or down-weighting the cost of the positive error relative to the negative error.

Text classification. For the text signal, we use the text CNN architecture of Kim (Kim 2014). The first layer embeds words into low-dimensional vectors using random embedding (different than the original paper). The next layer performs convolutions overtime on the embedded word vectors using multiple filter sizes (3, 4 and 5), where we use 128 filters from each size. Next, we max-pool-over-time the result of each convolution filter and concatenate all the results together. We add a dropout regularization layer (0.5 dropping rate), followed by a fully connected layer, and classify the result using a softmax layer.

Image classification. For the image signal, we use the VGG Network (Simonyan and Zisserman 2014). The input to the network is a fixed-size 224 x 224 RGB image. The image is passed through a stack of convolutional layers with a small receptive field: 3 x 3. The convolution stride is fixed to 1 pixel; the spatial padding of the convolutional layer is 1 pixel. Spatial pooling is carried out by five max-pooling layers, which follows some of the convolutional layers. Max-pooling is performed over a 2 x 2 pixel window, with stride 2. A stack of convolutional layers is followed by three Fully-Connected (FC) layers: the first two have 4096 channels each, the third performs 2890-way product classification and thus contains 2890 channels (one for each class). A ReLu non-linearity follows all hidden layers.

Multi-modal architectures. We experimented with four types of multi-modal architectures. *(1)* Learning decision-level fusion policies from different inputs. *(1a)* Policies that use the text and image CNNs **class probabilities** as input (Figure 4, right). For this input type, we experimented with architectures that have one or two fully connected layers (the two-layered policy is using 10 hidden units and a ReLu non-linearity between them). *(1b)* Policies that use the **text and/or image** as input. For these policies, the architecture of the policy network was either the text CNN or the VGG network. The labels for the policy training were collected

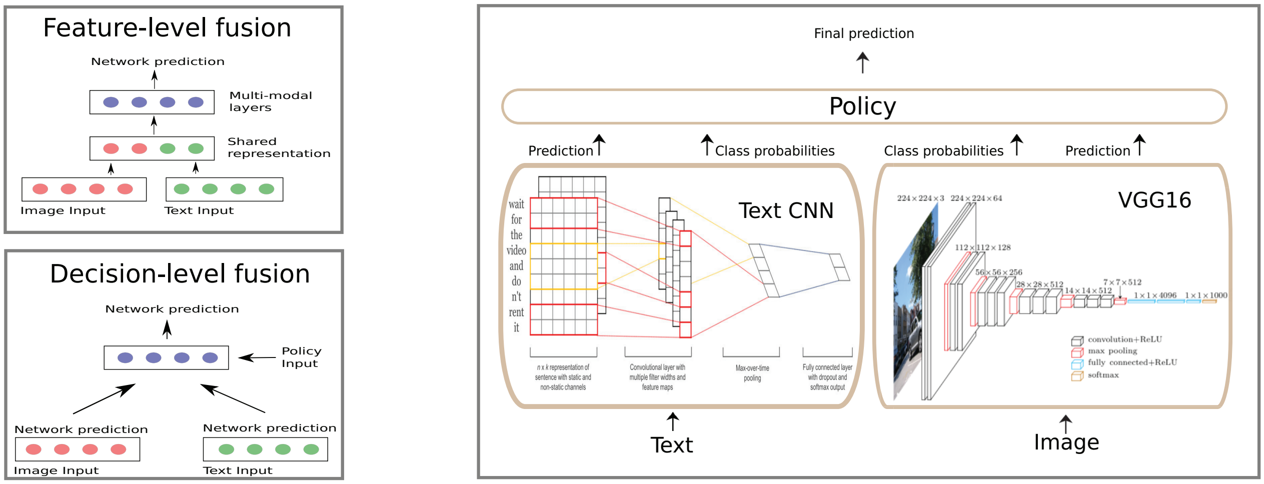

Figure 4: *Left:* multi-modal approaches. *Right:* The proposed, decison-level multi-modal fusion architecture.

from the image and text networks predictions, i.e., the label is 1 if the image network made a correct prediction while the text network made a mistake, and 0 otherwise. On evaluation, we used the policy predictions to select between the models, i.e., if the policy prediction is 1 we use the image network, and use the text network otherwise. *(2)* Pre-defined policies that average the predictions of the different CNNs or choose the CNN with the highest confidence. *(3)* End-to-end feature-level fusion, each input type is processed by its specific CNN. We concatenate the last hidden layers of the CNNs and add one or two fully connected layers. All the layers are trained together end-to-end (we also tried to initialize the input specific weights from pre-trained single-modal networks). *(4)* Multi-step feature-level fusion. As in (3), we create shared representation vector by concatenating the last hidden layers. However, we now keep the shared representation fixed and learn a new classifier from it.

Experiments

In this Section, we provide experimental results for deep multi-modal fusion. We start by describing our dataset and presenting exploratory analysis results for training single-modal deep neural nets on it. These results are summarized in Table 2. We then present results on multi-modal fusion techniques. We start by describing results on learning decision-level rules with neural networks (Table 3), which is the method that performed best in practice. For completeness, we also present results for using pre-defined decision rules and feature-level fusion (Table 4).

Setup. Our dataset contains 1.2 million products (title image and shelf) that we collected from Walmart.com and were deemed the hardest to classify by the current production system. We divide the data into training (1.1 million) validation (50k) and test (50k). We train both the image network and the text network on the training dataset and evaluate them on the test dataset. The policy is trained on the validation dataset and is also evaluated on the test dataset. The objective is to classify the product's shelf, from 2890 possible choices. Each product is typically assigned to more than one shelf (3 on average), and the network is considered accurate if its most probable shelf is one of them. Our code was implemented using TensorFlow and is available online at ⟨https://github.com/TomZahavy/multi_modality⟩.

Exploratory analysis

Text *Preprocess:* each word is embedded into a random vector in R^{100} using a dictionary of the 100k most common words in the training data. Titles with more than 40 words are trimmed, and shorter titles are padded with nulls.
Results: The best CNN that we trained classified 70.1% of the test set products correctly (Table 2). The best architecture was chosen after experimenting with different batch sizes, dropout rates, and filters strides. We found that this architecture is not sensitive to hyperparameters, which is consistent with the results of Zhang and Wallace 2015. We also tuned the cost function positive coefficient parameter q, and found out that the value 30 performed best.

Test	%
Text is correct	70.1
Image is correct	56.7
Text is correct, image is wrong	21.9
Image is correct, text is wrong	7.8
Both are correct	47.9
Both are wrong	22.4
At least one modality is correct	**79.9**

Table 2: Exploratory analysis.

Image: *Preprocess:* each image was re-sized to 224 x 224 pixels and the training set mean image was subtracted from

Policy network input	# layers	q	Policy	Optimal Policy	Policy accuracy
Top-1 class probability	1	5	71.4 (+1.3)	77.5 (+7.8)	86.4
Top-1 class probability	2	5	71.5 (+1.4)	77.6 (+7.5)	84.2
All class probability	2	5	71.4 (+1.3)	77.6 (+7.5)	84.6
Top-3 class probabilities	**2**	**5**	**71.8 (+1.7)**	**77.7 (+7.6)**	**84.2**
Top-3 class probabilities	2	1	70.2 (+0.1)	77.7 (+7.5)	92.5
Top-3 class probabilities	2	7	71.0 (+0.9)	77.5 (+7.4)	79.1
Top-3 class probabilities	2	10	70.7 (+0.6)	77.6 (+7.5)	75.0
Image	-	5	68.5(-1.6)	77.6 (+7.5)	80.3
Text	-	5	69.0 (-1.1)	77.6 (+7.5)	83.7
Both	-	5	66.1 (-4)	77.6 (+7.5)	73.7

Table 3: Decision-level fusion results. Each row presents a different policy configuration (defined by the policy input, the number of layers and the value of q), followed by the accuracy $\%$ of the image, text, policy and the optimal policy (if available, uses the correct modality) classifiers on the test dataset. The policy accuracy column presents the accuracy $\%$ of the policy in making correct predictions, i.e., choosing the image network when it made a correct prediction while the text network didn't. Numbers in $(+\cdot)$ refer to the performance gain over the text CNN. Class probabilities refer to the number of class probabilities (outputs of each single-modality networks last's layer) used as input.

it.

Results: The best VGG network that we trained classified 56.7% (Table 2) of the test products correctly. The performance of the VGG network on ImageNet, on the other hand, is $\sim$ 75%. The following postulates may explain this gap. First, Figure 3 implies that some of our images are not informative for shelf classification. As we deal with a real-world problem, there are no guarantees on how easy it to classify products correctly based on images alone. Imagenet, on the other hand, is a scientific dataset, and each image is related to its actual category. Second, our data has three times more classes and contains multiple labels per image, thus, making the classification harder.

Error Analysis: Looking at Table 2 (top), we observe that the text network (70.1%) outperformed the image network (56.7%) on our dataset, so maybe, an image does not worth a thousand words after all. We note that similar results were reported in other e-commerce domains (Pyo, Ha, and Kim 2010; Kannan et al. 2011). Next, we were interested in measuring the potential of multi-modality. By analyzing the errors of each single-modal network (Table 2 , bottom), we can see that there is a relatively large potential (7.8%) to harness via multi-modality. We note that this large gap is an encouraging result for multi-modality, in particular since different neural networks applied to the same input source tend to make the same mistakes (Szegedy et al. 2013).

Multi-modal unification techniques

Our exploratory analysis experiments highlight the potential of merging image and text inputs. Still, we found it hard to achieve the optimal multi-modal fusion that was observed in the error analysis experiment. We now describe in detail the decision-level fusion policies that managed to reach the best performance boost in accuracy. We then provide results on pre-defined rules and feature-level fusion. Since these methods did not provide an improvement in accuracy, we only report the best configuration for each technique.

Decision-level fusion

Input type: We trained policies from the different data sources, i.e., title, image, and the class probabilities (the softmax probabilities) of the image and text CNNs as inputs. Looking at Table 3, we can see that the best policies were trained using class probabilities. The number of class probabilities that were used (top-1, top-3 or all) did not have a significant effect on the results, indicating that the top-1 probability contains enough information to learn good policies. This result makes sense since the top-1 probability can measure the confidence of the network in making a prediction. Still, the top-3 probabilities performed slightly better, indicating that the difference between the top probabilities may also matter. We also tried to learn policies from the text and/or the image input, using a policy network which is either a text CNN, a VGG network or a combination. For these policy networks, we experimented with early stopping criteria, various regularization methods (dropout, l1, l2) and reduced model size (best configuration reported). However, working with the text and/or image modalities as inputs to the policy network resulted in policies that overfitted the data and performed worse than the single-modal text network on the test data.

Hyperparameters: Looking at Table 3, we can see that the 2-layer architecture outperformed the 1-layer, indicating

that a linear policy is too simple, and non-linear policies can yield better results.
q : the cost function coefficient that trades off recall and precision, had a significant impact on the results. Recall that the policy network implicitly optimizes the multi-modal architecture accuracy, by explicitly learning to choose between single modality networks. As q increases, the policy networks learn to favor correct positive predictions (selecting the text network) over negative predictions (choosing the negative ones). Since the text network is more accurate than the image network, this results in higher accuracy of the multi-modal architecture but with a lower accuracy of the policy network (Table 3, columns four and six respectively).

Pre-defined rules:

we experimented with averaging the logits (Table 4, Mean), following (Krizhevsky, Sutskever, and Hinton 2012; Simonyan and Zisserman 2014), and with choosing the network with the maximal confidence (Table 4, Max) following (Poria et al. 2016). Both of these experiments yielded significantly worse results, probably due to the bias in accuracy between the input specific networks.

Feature-level fusion.

Training a feature-level fusion architecture end-to-end is not an easy task, as each input source has its best architecture, learning rate, and optimization algorithm. Therefore, we experimented with training the network end-to-end (Table 4, End-to-end), but also with first training each part separately and then learning the concatenated parts (Table 4, Step by step). We tried different unification approaches such as concatenating the features to one layer and using gating functions (Srivastava, Greff, and Schmidhuber 2015) or cross products between the embeddings of the two modalities. We also experimented with different architectures, for example, with the number of fully connected layers after the concatenation. Despite all of these experiments, the best results that we achieved for feature-level fusion were inferior to those of the text model. We do not claim that it is not possible to gain improvement from such methods, only that we were unable to find such. While this may seem surprising, the most successful feature level fusion that we are aware of (Frome et al. 2013), was not able to gain improvement in classification accuracy.

Fusion method	Text	Image	Policy
Fixed policy, mean	70.1	56.7	65.4 (-4.7)
Fixed policy, max	70.1	56.7	60.1 (-10)
Feature-level, end-to-end	70.1	56.7	69.1 (-1)
Feature-level, step by step	70.1	56.7	69.5 (-0.6)

Table 4: Pre-defined rules and feature-level fusion results.

Conclusions

In this work, we investigated a multi-modal multi-class multi-label product classification problem and presented results on a challenging real-world dataset that we collected from Walmart.com. We discovered that the text network outperforms the image network on our dataset, and demonstrated that by learning a decision rule with a deep neural network, it is possible to achieve better performance than with single-modal architectures.

When using state-of-the-art deep neural networks in production, practitioners are in a constant search for improving classification accuracy. In this work, we explored a method that is orthogonal to architecture search and demonstrated that it could achieve further improvement. While we only managed to reach 2% improvement in accuracy, we note that such an increase has a tremendous significance when deployed in production. Moreover, to the best of our knowledge, this is the first work that achieves an improvement in classification accuracy by using multi-modality on a large-scale classification problem. We also hope that this work will motivate others to explore the potential of multi-modal classification further. Indeed, after a workshop version of this paper was published, there have been successful attempts to use our methods on other e-commerce classification problems (Eskesen 2017).

Acknowledgments

This research was supported in part by the European Communitys Seventh Framework Programme (FP7/2007-2013) under grant agreement 306638 (SUPREL).

References

Bengio, Y.; Courville, A.; and Vincent, P. 2013. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence* 35(8).

Conneau, A.; Schwenk, H.; Barrault, L.; and Lecun, Y. 2016. Very deep convolutional networks for natural language processing. *arXiv preprint arXiv:1606.01781*.

Eskesen, S. 2017. Improving product categorization by combining image and title.

Frome, A.; Corrado, G. S.; Shlens, J.; Bengio, S.; Dean, J.; Mikolov, T.; et al. 2013. Devise: A deep visual-semantic embedding model. In *Advances in neural information processing systems*.

Gong, Y.; Wang, L.; Hodosh, M.; Hockenmaier, J.; and Lazebnik, S. 2014. Improving image-sentence embeddings using large weakly annotated photo collections. In *European Conference on Computer Vision*, 529–545. Springer.

Guillaumin, M.; Verbeek, J.; and Schmid, C. 2010. Multimodal semi-supervised learning for image classification. In *CVPR 2010-23rd IEEE Conference on Computer Vision & Pattern Recognition*, 902–909. IEEE Computer Society.

Hansen, L. K., and Salamon, P. 1990. Neural network ensembles. *IEEE transactions on pattern analysis and machine intelligence* 12:993–1001.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2015. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*.

Kannan, A.; Talukdar, P. P.; Rasiwasia, N.; and Ke, Q. 2011. Improving product classification using images. In *2011 IEEE 11th International Conference on Data Mining*. IEEE.

Kim, Y. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

Kiros, R.; Salakhutdinov, R.; and Zemel, R. S. Multimodal neural language models.

Kittler, J.; Hatef, M.; Duin, R. P.; and Matas, J. 1998. On combining classifiers. *IEEE transactions on pattern analysis and machine intelligence* 20(3):226–239.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*.

Lai, S.; Xu, L.; Liu, K.; and Zhao, J. 2015. Recurrent convolutional neural networks for text classification.

Lynch, C.; Aryafar, K.; and Attenberg, J. 2015. Images don't lie: Transferring deep visual semantic features to large-scale multimodal learning to rank. *arXiv preprint arXiv:1511.06746*.

Ngiam, J.; Khosla, A.; Kim, M.; Nam, J.; Lee, H.; and Ng, A. Y. 2011. Multimodal deep learning. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, 689–696.

Poria, S.; Cambria, E.; Howard, N.; Huang, G.-B.; and Hussain, A. 2016. Fusing audio, visual and textual clues for sentiment analysis from multimodal content. *Neurocomputing*.

Pyo, H.; Ha, J.-W.; and Kim, J. 2010. Large-scale item categorization in e-commerce using multiple recurrent neural networks.

Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; Berg, A. C.; and Fei-Fei, L. 2015. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)* 115(3):211–252.

Simonyan, K., and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Srivastava, R. K.; Greff, K.; and Schmidhuber, J. 2015. Highway networks. *arXiv preprint arXiv:1505.00387*.

Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; and Fergus, R. 2013. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Xiao, Y., and Cho, K. 2016. Efficient character-level document classification by combining convolution and recurrent layers. *arXiv preprint arXiv:1602.00367*.

Zhang, Y., and Wallace, B. 2015. A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. *arXiv preprint arXiv:1510.03820*.

Zhang, X.; Zhao, J.; and LeCun, Y. 2015. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems*.