Image Labelling using Feature Learning and Boltzmann Machine-Augmented CRFs

Charles Lo
Department of Electrical and Computer Engineering
University of Toronto
locharl1@ece.utoronto.ca

Abstract

Image labelling is an important and challenging task in computer vision. Recent bottom-up approaches to the problem have made use of deep neural networks for feature learning, as well as fully-connected CRFs and Boltzmann machine-augmented CRFs to improve labelling consistency. This work explores the combination of these techniques to label images at superpixel granularity. We find that a classifier based solely on superpixels of a finely segmented image does not perform well, but is greatly improved by the combination of a fully-connected CRF augmented with a restricted Boltzmann machine.

1 Introduction

Semantic scene understanding is an important task for several applications such as in autonomous robotics as well as medical imaging. However, pixel-level segmentation and labelling can be very challenging; approaches must take into account both appearance and prior contextual knowledge to obtain precise labelling.

The most common method of tackling the labelling problem is through a bottom-up approach. First, pixels are classified using local appearance information taking into account texture and colour of the different object categories. Second, local and global context are used to smooth the predications across pixels. Local context encodes the knowledge that objects are generally contiguous across pixels whereas global context relates sets of objects that commonly appear together as well as their relative positions and shapes. The most common approach for ensuring consistency across pixels is Conditional Random Fields (CRFs).

In this paper, a method combining advances in feature learning and augmented CRFs is applied to the image labelling problem. Rather than designing appearance features by hand, a multilayer neural network is applied directly to pixels. Deep neural networks have shown promise recently in supervised classification tasks [1, 2] and can automatically learn hierarchical layers features. The pixel-wise, appearance-based labelling task is a large-scale classification problem and is thus well-suited to deep neural networks.

The second technique applied in this paper is the use of Restricted Boltzmann Machines [3] (RBMs) to augment CRFs. CRFs can efficiently ensure local consistency in labelling, but they do not perform as well at encoding global information. Thus, the local connectivity of the CRF is augmented with connections to global hidden units forming a system known as a semi-restricted Boltzmann machine [4] (SRBM).

Finally, this work applies the previously discussed techniques on groups of pixels, called *superpixels* rather than individual pixels. Although there has been progress in improving performance of inference and training in the aforementioned techniques, applying the techniques to high-definition images or video can still be a challenge. Superpixels are formed by partitioning the image in a way that maintains local appearance consistency and can dramatically reduce the computational complexity of learning and inference.

2 Related Work

The majority of methods for image labelling use manually defined features that are known to work well. However, there has been some investigation into automated feature learning for this task. Farabet et al. [5] investigate the use of deep convolutional neural networks to learn pixel-level features that are then combined with segmentation and local CRF models to perform scene labelling. An important characteristic of their work is the use of multi-scale networks to incorporate context around a pixel at multiple distances. This allows their classifier to perform well even without smoothing from the CRF. In this work, a single receptive field of one superpixel is used and predictions are made at the superpixel level rather than for individual pixels to reduce computational complexity.

The basic form of CRF for smoothing labels is a graph over labels of pixels where an edge exists between labels of adjacent pixels as well as between a label and its associated pixel. Such a CRF has the form:

$$p(Y|X,\theta) = \frac{1}{Z}\exp(-\sum_{i\in X}\psi_u(x_i,y_i,\theta) - \sum_{y_i,y_{i'}\in E}\psi_p(y_i,y_{i'},\theta)) \tag{1}$$

where X is the set of (super)pixels, Y is the set of labels and E is the set of edges between labels. The appearance potential ψ_u is the contribution by a local classifier given information about the pixel appearance while the pairwise potential ψ_p encourages local uniformity in the labels. Generally, CRFs for image labelling have had connectivity limited to adjacent pixels. Recently, Krähenbühl et al. [6] have described a method for efficient inference in fully connected CRFs when using Gaussian pairwise potentials. While these CRFs are able to ensure local smoothness, they do not capture other contextual information in the scene such as the likelihood of certain classes co-occuring, shape of the objects or relative location of them.

More recent works on image labelling have incorporated higher-order potentials into the CRF to take such factors into account. In particular, leading image labelling schemes [7, 8] on the MSRC-21 dataset [9] use a combination of potentials in hierarchal CRFs to encode prior information on shape and co-occurance of classes within a scene.

An alternative approach for augmenting CRF models has been the use of restricted Boltzmann machines (RBMs) [3] to provide global features. Restricted Boltzmann machines (RBMs) [3] are well studied generative models where visible and hidden nodes form two partitions of a bipartite graph. RBMs can be efficiently trained in an unsupervised manner to maximize the probability of a training set on its visible nodes. Several existing works [10, 11, 12] have made use of RBMs to augment local CRF connections with long-range information. In this work, we use RBMs over superixels as in [10], but do so over fully connected CRFs.

3 Local Classification

The problem of independently predicting labels of individual pixels is a challenging task. Generally, features are designed to measure the appearance of a patch of pixels around the one of interest. Two critical decisions to be made when performing local classification are determining the size of patch and types of features to use. In this work, labelling is performed at superpixel granularity and features are automatically learned from the raw pixels contained within the superpixels.

3.1 Superpixel segmentation

The problem of ensuring long-range consistency in labelling becomes difficult in large images. Even a relatively small image of 320 × 213 resolution, would contain 68,160 pixel labelling predications. Individually classifying each pixel is also computationally expensive in such a scheme when using a complex classifier. Thus, many approaches perform image labelling at the level of *superpixels*.

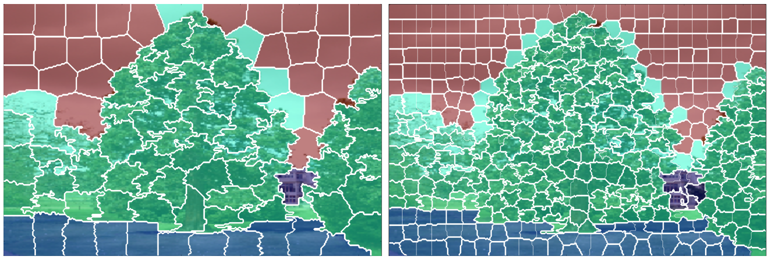

Figure 1: SLIC superpixel image segmentation with 100 (left) and 500 (right) segments

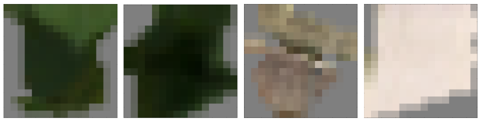

Figure 2: Superpixel patches resized into 16×16 windows

Simple linear iterative clustering (SLIC) [13] was used to segment scenes into superpixels. This method is based on local K-means clustering in LAB colour space and position. The algorithm is very fast and has been shown to provide good performance at image segmentation tasks.

Each image was segmented into approximately 500 superpixels with a compactness setting of 25.[1] This size presented a good trade-off between providing the local classifier with enough pixel data to work with while being fine enough to discriminate borders between classes. An example of segmentation with 100 and 500 superpixels is shown in Fig. 1. The images, overlaid with best-case labelling of superpixels, demonstrate that 500 superpixels provides finer discrimination between the road and tree in the scene.

3.2 Neural Network Feature Learning

SLIC segmentation produces partitions of irregular size and shape. To provide a standard receptive field for the neural network feature learning system, the partitions found by segmentation were isolated and scaled into 16×16 windows. All dataset images were resized to 320×213, and after segmentation into 500 superpixels, the mean extent of each segment was found to be approximately 16×16. Some rescaled superpixels are shown in Fig. 2; these patches retain shape, colour and texture information from the superpixel segmentation.

After experimenting with several neural network topologies, the best architecture had the form 768-512-512-output with two hidden layers and a final one-hot encoded output layer. The colour pixels in the receptive field were modelled as Gaussian units with zero mean and unit variance. All hidden units were rectified linear units [14] which we found worked better and faster than binary sigmoid units. Finally, the output layer was modelled as a softmax:

$$p(y = l) = \frac{\exp(\sum_m w_{lm} h_m)}{\sum_{l' \in \text{labels}} \exp(\sum_m w_{l'm} h_m)} \tag{2}$$

where h_m is the m'th unit of the last hidden layer and w are the weights between the last hidden layer and the label units.

To learn discriminative features, cross-entropy error was used as the loss function. Mini-batch stochastic gradient descent with Nesterov momentum [15] was used in combination with dropout [16] to improve generalization performance. The parameters producing the best validation error were chosen and training was stopped after 500 epochs at which point validation error was increasing.

[1] The original SLIC author implementation was used in this work. In the proposal, the scikit-learn implementation was used with lower compactness. The current settings produced better results.

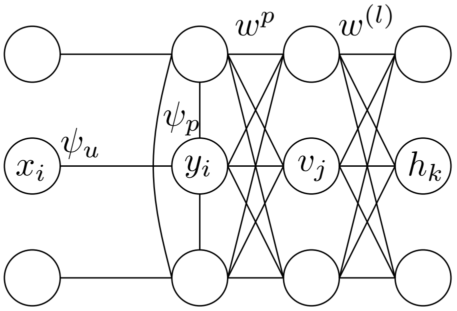

Figure 3: CRF-RBM schematic view

4 Consistent Scene Labelling

Predications made by local classifiers tend to be very noisy since they do not consider the labelling of adjacent pixels or the global context of the scene. This work makes use of a CRF-RBM structure to ensure consistent labelling; the architecture is shown in Fig. 3. The fully-connected CRF model (Eqn. 1) is combined with an RBM model:

$$p(V,H) = \frac{1}{Z}\exp(\sum_{l}^{L}\sum_{j}^{V}\sum_{k}^{H} w_{j,k}^{(l)} v_{lj} h_k + \sum_{l}^{L}\sum_{j}^{V} d_{lj} v_{lj} + \sum_{k}^{H} e_k h_k) \tag{3}$$

where the visible nodes are multinomial random variables. For a particular label, l, the connections between visible and hidden nodes are written $w^{(l)}$, visible biases are denoted d_l and hidden node biases e.

4.1 Conditional Random Field

Each superpixel i consists of a set of colour pixels x_i and is given a label $y_i \in \{1 \ldots L\}$. The unary potential $\psi_u(x_i, y_i)$ in this work is given directly from the neural network $\psi_u(x_i, y_i = l) = -\sum_m w_{lm} h_m$. Following [6], two pairwise Potts potentials are used for local consistency:

$$\psi_p(y_i, y_{i'}) = \mathbf{1}_{[y_i \neq y_{i'}]}(w_1 \exp(-\frac{\|p_i - p_{i'}\|^2}{2\sigma_p^2} - \frac{(c_i - c_{i'})^2}{2\sigma_c^2})) + w_2 \exp(-\frac{\|p_i - p_{i'}\|^2}{2\sigma_\gamma^2})$$

where p and c represent mean position and colour of the superpixels respectively.

4.2 Restricted Boltzmann Machine

For each image, the number of superpixels and their positions can vary. To relate visible nodes of the RBM to positions in the scene, a *virtual* visible layer (V) is used as in [10]. Virtual visible nodes are distributed across the scene in fixed locations to spatially anchor the RBM. Unlike the previous work, superpixels are related continuously to each virtual visible node with a Gaussian kernel $w^p(y_i, v_j) = \exp(-\|p_i - pv_j\|^2/(2\sigma_v^2))$, where p_i is the mean position of superpixel i and pv_j is the position of the virtual visible node j. Rectified linear units were used for the hidden nodes rather than stochastic binary units. As with the pairwise and unary potentials, the potential projected onto the label nodes from the RBM are weighted with a parameter w_r.

4.3 Inference

Inference was performed using a mean-field approximation i.e. $p(Y, H|X) \approx \prod_i^Y Q_i(y_i) \prod_k^H Q_k(h_k)$. The iterative algorithm for inference is shown in Alg. 1; in practise 10 iterations of inference were used.

4.4 Learning

Piecewise learning was used in this work. First, the local classifier was trained to independently label superpixels as discussed in section 3. Concurrently, the RBM was trained given the labels and pro-

Algorithm 1 Mean-field Inference

```
Initialize Q
$Q_i(y_i) \leftarrow \frac{1}{Z_i} \exp(-\psi_u(x_i, y_i))$
$Q_k(h_k) \leftarrow \max(0, e_k + \sum_{l,j} w_{jk}^{(l)} (\sum_i w_{ij}^p y_i))$
while not converged do
    $K_j^1(y_j = l) \leftarrow w_1 \sum_{j' \neq j} \exp(-\frac{\|p_j - p_j'\|^2}{2\sigma_p^2} - \frac{(c_j - c_j')^2}{2\sigma_c^2}) Q_{j'}(l)$
    $K_j^2(y_j = l) \leftarrow w_2 \sum_{j' \neq j} \exp(-\frac{\|p_i - p_j\|^2}{2\sigma_\gamma^2}) Q_{j'}(l))$
    $R_i(y_i = l) \leftarrow w_r \sum_j w_{ij}^p ((\sum_k w_{jk}^{(l)} Q_k(h_k)) + d_{lj})$
    $Q_i(y_i = l) \leftarrow \exp\{-\psi_u(x_i, y_i) - K_i^1(y_i) - K_i^2(y_i) + R_i(y_i)\}$
    Normalize $Q_i(y_i)$
    $Q_k(h_k = 1) \leftarrow \max(0, e_k + \sum_{l,j} w_{jk}^{(l)} (\sum_i (w_{ij}^p y_i)))$
```

jection matrix using contrastive divergence (CD-1). Finally, CRF weights were determined through a search using a validation set. Further improvements could be made by performing mean-field contrastive divergence learning on the combined model, but training using the current implementation proved too slow.

5 Experiments

5.1 Datasets

There are a number of datasets available for image labelling. One of the most popular is the MSRC-21 dataset [9] assembled by Microsoft Research. The most popular version consists of 591 images with 23 classes. Two of the classes (mountain and horse) are infrequent and are often removed from testing, resulting in a 21-class dataset. This dataset has a popular pre-defined split into training, validation and test sets which were used in this work.

Another popular dataset is the Stanford Background Dataset [17]. This dataset features 715 images, hand labelled via the Amazon Mechanical Turk service. Unlike the MSRC-21 dataset, all scenes are of the outdoors and there are fewer classes. This dataset was randomly split into 429(60%) training, 143(20%) validation and 143(20%) testing sets.

In both datasets, there are void or unknown regions in certain scenes where pixels do not belong to a specific category. Pixels containing these labels are ignored during training and testing.

5.2 Implementation

All dataset images were resized to 320×213 to ensure a consistent position for the virtual visible nodes. These virtual nodes were distributed with 32 nodes along the width and 21 along the height of each image and σ_v was set to 10. The RBMs were trained with 100 hidden nodes.

After evaluating against the validation set, the values $\sigma_p = 60$, $\sigma_c = 20$, $\sigma_\gamma = 3$, $w_1 = 3$, $w_2 = 1$ and $w_r = 0.02$ were found to give the best performance.

The implementations of deep neural network training and restricted Boltzmann machines were written from scratch using gnumpy [18] to make use of a GPU. The dense CRF code [6] was modified to include inference of the RBM.

5.3 Results

The per-class accuracies as well as pixel-level global and average class accuracies are reported in tables 1 and 2. Images were classified using the independent unary (neural network) classifier alone as well as with the unary classifier augmented with a pairwise CRF and global RBM.

A clear trend in both of the datasets is that augmenting the independent labelling with CRFs and RBMs dramatically improve classification performance. A sample test image is shown in Fig. 4

	Build.	Grass	Tree	Cow	Sheep	Sky	Aero.	Water	Face	Car	Bicycle	Flower	Sign	Bird	Book	Chair	Road	Cat	Dog	Body	Boat	Global	Average
Unary Only	34	94	67	33	15	86	4	49	53	26	52	40	6	0	33	2	62	12	8	21	0	53.9	33.4
Unary + CRF	41	97	76	39	14	93	1	55	59	28	64	45	5	0	37	1	72	15	6	24	0	58.7	36.8
Unary + RBM	45	99	74	58	45	94	48	52	67	20	46	52	8	3	55	5	67	11	6	26	2	61.4	41.9
Unary + RBM + CRF	59	99	79	65	47	93	32	56	72	34	78	59	6	3	69	0	78	11	8	29	1	66.8	46.5

Table 1: Pixel-level accuracy (percent) on the MSRC-21 dataset

	Sky	Tree	Road	Grass	Water	Build.	Mount.	Foreg.	Global	Average
Unary Only	90	56	73	70	41	62	0	44	64.1	54.3
Unary + CRF	94	56	78	69	44	71	0	36	67.7	57.2
Unary + RBM	87	51	82	76	41	80	0	33	68.5	56.3
Unary + RBM + CRF	89	52	82	79	42	79	0	33	69.0	57.0

Table 2: Pixel-level accuracy (percent) on the Stanford dataset

classified with the various schemes in this paper. It is clear that unary classifier on its own provides very sporadic results since it has very little information to work with. In particular, the foreground objects tend to confuse the classifier since they tend to have much more variation than the background. The pairwise CRF across labels smooths out the predictions but many mis-predictions persist. Applying the global RBM provides context for the scene and reduces the labels to cow, sky and grass. The full model with the CRF-RBM smooths out this final prediction and provides slightly improves performance. Note that not all scenes benefit from the augmentations. In scenes where the unary classifier performs very poorly the CRF and RBM may move the labelling even farther from the ground truth.

Although the CRF and RBM improve performance of the unary classifier, the overall system still does not perform close to the state of the art. For instance, Farabet et al. [5] report 81% global and 76% average accuracy on the Stanford dataset and Krähenbühl et al. [6] report 86% global and 78% average accuracy on the MSRC-21 dataset. This gap is due to the much better pixel-wise, unary classifiers that the above works employ. Although Farabet et al. also use neural networks, they are only able to achieve competitive classification accuracy once they employ multi-scale nets to take in additional context. This result leads to the conclusion that RGB data from superpixels is not sufficient to correctly label them and the input should be augmented to achieve better performance.

6 Conclusions and Future Work

This work explored the use of neural networks to learn local features from superpixels as well as the combination of fully-connected CRFs and RBMs to improve consistency in labelling. We found that the local classifier based solely on superpixel information performed relatively poorly,indicating that there may not be enough information in the relatively small superpixels to produce good classification results. Future work should explore aggregating information from adjacent (super)pixels and consider the trade-off in labelling accuracy when using larger superpixels.

Although the neural network results were relatively poor, this work did reveal dramatic improvements to labelling performance with the CRF-RBM structure. Based on the obtained results, a system with a more accurate unary classifier should be able to benefit even further from the labelling consistency provided by the structure.

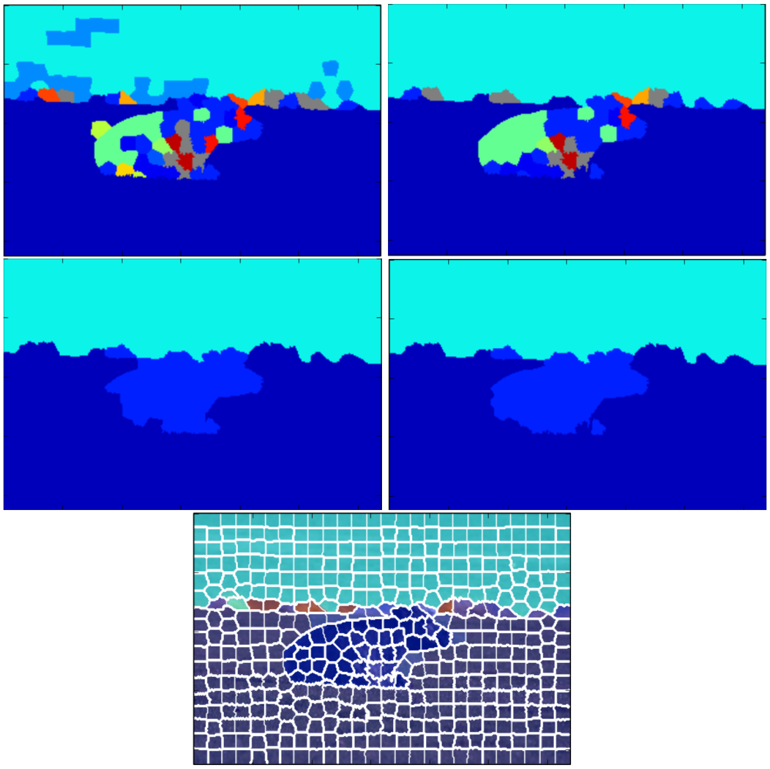

Figure 4: Example classification results: Unary only (top left), Unary+CRF (top right), Unary+RBM (bottom left), Unary+CRF+RBM (bottom right), Ground Truth (bottom).

References

[1] L. Wan, M. Zeiler, S. Zhang, Y. LeCun, and R. Fergus. Regularization of Neural Networks using DropConnect. In *International Conference on Machine Learning (ICML)*, pages 1058–1066, 2013.

[2] G. E. Dahl, T. N. Sainath, and G. E. Hinton. Improving Deep Neural Networks for LVCSR using Rectified Linear Units and Dropout. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8609–8613, 2013.

[3] P. Smolensky. Information Processing in Dynamical Systems: Foundations of Harmony Theory. In *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1*, pages 194–281. MIT Press, Cambridge, MA, USA, 1986.

[4] S. Osindero and G. E. Hinton. Modeling image patches with a directed hierarchy of Markov random fields. In *Advances in Neural Information Processing Systems (NIPS)*, 2008.

[5] C. Farabet, C. Couprie, L. Najman, and Y. Lecun. Learning Hierarchical Features for Scene Labeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 35(8):1915–1929, Aug 2013.

[6] P. Krähenbühl and V. Koltun. Efficient Inference in Fully Connected CRFs with Gaussian Edge Potentials. In *Advances in Neural Information Processing Systems (NIPS)*, 2011.

[7] X. Boix, J. Gonfaus, J. van de Weijer, A. Bagdanov, J. Serrat Gual, and J. Gonzàlez. Harmony Potentials - Fusing Global and Local Scale for Semantic Image Segmentation. *International Journal of Computer Vision (IJCV)*, 96(1):83–102, 2012.

[8] J. Yao, S. Fidler, and R. Urtasun. Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation. In *Computer Vision and Pattern Recognition (CVPR)*, pages 702–709, June 2012.

[9] J. Shotton, J. Winn, C. Rother, and A. Criminisi. TextonBoost for Image Understanding: Multi-Class Object Recognition and Segmentation by Jointly Modeling Texture, Layout, and Context. *International Journal of Computer Vision (IJCV)*, 81(1):2–23, 2009.

[10] A. Kae, K. Sohn, H. Lee, and E. Learned-Miller. Augmenting CRFs with Boltzmann Machine Shape Priors for Image Labeling. *Computer Vision and Pattern Recognition (CVPR)*, 0:2019–2026, 2013.

[11] Yujia Li, D. Tarlow, and R. Zemel. Exploring Compositional High Order Pattern Potentials for Structured Output Learning. In *Computer Vision and Pattern Recognition (CVPR)*, pages 49–56, June 2013.

[12] X. He, R. Zemel, and M.A. Carreira-Perpindn. Multiscale conditional random fields for image labeling. In *Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 695–702, June 2004.

[13] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and Sabine Süsstrunk. SLIC Superpixels Compared to State-of-the-Art Superpixel Methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 34(11):2274–2282, 2012.

[14] V. Nair and G. E. Hinton. Rectified Linear Units Improve Restricted Boltzmann Machines. In *International Conference on Machine Learning (ICML)*, pages 807–814, 2010.

[15] I. Sutskever, J. Martens, G. Dahl, and G. E. Hinton. On the importance of initialization and momentum in deep learning. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, volume 28, pages 1139–1147. JMLR Workshop and Conference Proceedings, May 2013.

[16] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint, arXiv:1207.0580*, 2012.

[17] S. Gould, R. Fulton, and D. Koller. Decomposing a Scene into Geometric and Semantically Consistent Regions. In *International Conference on Computer Vision (ICCV), 2009*, pages 1–8, Sept 2009.

[18] T. Tieleman. Gnumpy: an easy way to use GPU boards in Python. Technical Report UTML TR 2010-002, University of Toronto, Department of Computer Science, 2010.