Optimisation of Robust Loss Functions for Weakly-Labelled Image Taxonomies

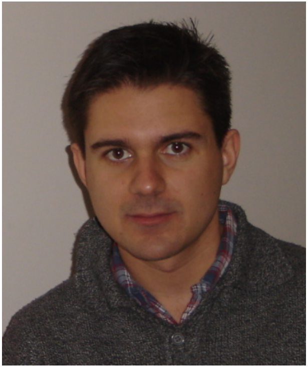
Julian McAuley

ANU $\rightsquigarrow$ Stanford

July 25, 2011

Thanks to my collaborators

Arnau Ramisa



Tibério Caetano

One-slide summary

- Recently, a classification competition was held using the ImageNet dataset (Berg et al., 2010)[1]
- Entrants were evaluated using a structured performance measure
- None of the top entrants optimised this performance measure directly

Can we do better if we use structured learning techniques?

[1] `http://www.image-net.org/challenges/LSVRC/2010/index`

IMAGENET

12,184,113 images, 17624 synsets indexed

Explore New! Download New! **Challenge** People Publication About

Not logged in. Login | Signup

ImageNet is an image database organized according to the WordNet hierarchy (currently only the nouns), in which each node of the hierarchy is depicted by hundreds and thousands of images. Currently we have an average of over five hundred images per node. We hope ImageNet will become a useful resource for researchers, educators, students and all of you who share our passion for pictures.
Click here to learn more about ImageNet, Click here to join the ImageNet mailing list.

SEARCH

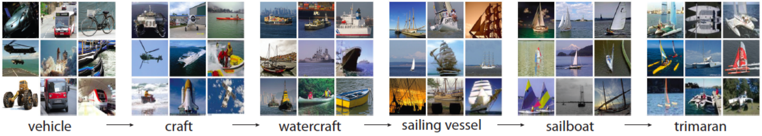
What do these images have in common? *Find out!*

(Deng et al., 2009)

The ImageNet Dataset

- Over 12 million images
- Over 17 thousand categories
- Categories are organised in a taxonomy, derived from Wordnet
- Each image is annotated with a single category


mammal
placental
carnivore
canine
dog
working dog
husky


vehicle
craft
watercraft
sailing vessel
sailboat
trimaran

Building ImageNet

- Query several image search engines with WordNet nouns
- Additional queries by translating the nouns into other languages
- Cleaning by asking Turkers to check which images correspond to each noun
- Disambiguate mistakes from different Turkers by voting

Result: very accurate labelling of one label per image

Iron

Orchestra Pit

Dalmatian

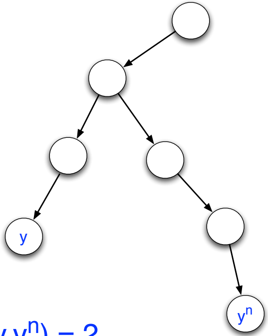
African Marigold

Evaluating a Classifier

1. A classifier should not be heavily penalised if its output is 'close to' the correct output
2. A classifier should not be penalised for predicting objects that appear in the image, but were not labelled

Evaluating a Classifier (1)

- A classifier should not be heavily penalised if its output is 'close to' the correct output
- $d(y, y^n)$ is the distance between the node y^n and the nearest common ancestor of y and y^n



$d(y,y^n) = 2$

Evaluating a Classifier (2)

- A classifier should not be penalised for predicting objects that appear in the image, but were not labelled
- The classifier is allowed to output a set of labels Y
- Only the most accurate label is considered

The ImageNet Loss Function

$$d(Y, y^n) = \min_{y \in Y} d(y, y^n)$$

Obviously, the total number of labels K is limited to avoid degeneracy. $K = 5$ in the competition.

The Competition

- Subset of ImageNet: 1000 categories and 1.2 million images
- Competition winners: one-vs-all, 1000 linear binary SVMs trained with SGD and proprietary features
- Disregard the competition loss
- Secret sauce: features + efficient implementation
- Obvious question: can the taxonomy improve classification at all?

This is our research question

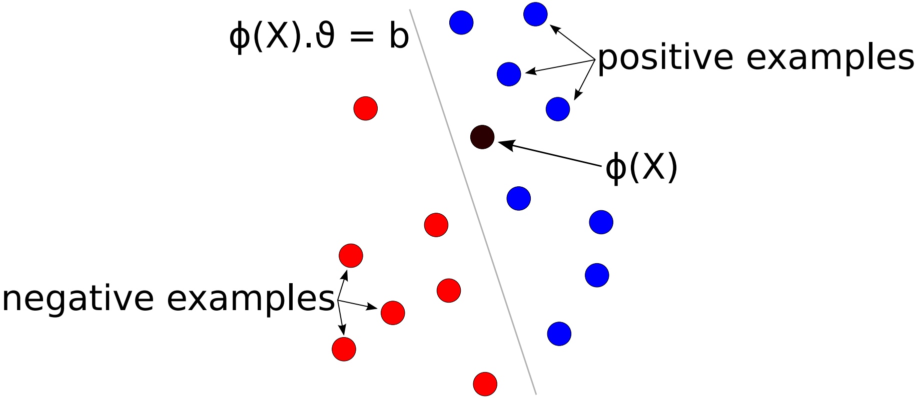
Basic Strategy

- Given the success of one-vs-all SVMs, our focus is on refining rather than replacing them

- Re-weight the SVM parameter vectors with a single weighting vector so as to minimise an upper bound on competition's loss function

$\rightarrow$ The hypothesis to be tested is: does such refinement improve accuracy?

Support Vector Machines



Support Vector Machines

'Soft-margin' formulation

$$\min_{\theta,\xi} \tfrac{1}{2}||\theta||^2 + C\textstyle\sum_i \xi_i$$
$$\text{subject to} \quad c_i(\theta \cdot \Phi(\mathbf{x}_i) - b) \geq 1 - \xi_i, \quad \xi_i \geq 0$$

- We want to be confident about correct predictions
- We want to be doubtful about incorrect predictions

Prediction for one-vs-all SVMs

In the one-vs-all SVM methods, prediction of a single category y for a given image x amounts to finding

$$\bar{y}_{\text{binary}}(x) = \underset{y \in \mathcal{C}}{\operatorname{argmax}} \left\langle x, \theta^{y}_{\text{binary}} \right\rangle$$

($\mathcal{C}$ is the set of classes)

For predicting 5 labels, return those with higher scores

$$\bar{Y}_{\text{binary}}(x) = \underset{Y \in \mathcal{Y}}{\operatorname{argmax}} \sum_{y \in Y} \left\langle x, \theta^{y}_{\text{binary}} \right\rangle$$

($\mathcal{Y}$ is the set of sets of 5 labels)

Diminishing Returns

- Correct label: African Marigold
- Plausible (but incorrect) labels:
  - European rabbit
  - Cottontail rabbit
  - New England cottontail
  - Mexican cottontail
  - Mountain cottontail

Prediction in our model

- Let $\{\theta^y_{\text{binary}}\}$ be the set of parameters learned by the binary SVMs. *We prepared our own features: no access to the proprietary features of the winners*
- We introduce a single parameter vector θ to parameterise each θ^y_{binary}
- We propose the following predictor

$$\bar{Y}(x;\theta) = \operatorname*{argmax}_{Y \in \mathcal{Y}} \sum_{y \in Y} \left\langle x \odot \theta^y_{\text{binary}}, \theta \right\rangle, \text{ or}$$

$$\bar{Y}(x;\theta) = \operatorname*{argmax}_{Y \in \mathcal{Y}} \left\langle \underbrace{\sum_{y \in Y} \overbrace{x \odot \theta^y_{\text{binary}}}^{:=\phi(x,y)}}_{:=\Phi(x,Y)}, \theta \right\rangle$$

- For $\theta = \mathbf{1}$ we recover the one-vs-all linear predictor

The Convex Relaxation

$$[\theta^*, \xi^*] = \operatorname*{argmin}_{\theta,\xi} \left[\frac{1}{N} \sum_{n=1}^{N} \xi_n + \lambda \|\theta\|^2 \right]$$

$$\text{s.t. } \underbrace{\langle \Phi(x^n, y^n), \theta \rangle - \langle \Phi(x^n, Y), \theta \rangle}_{\text{margin}} \geq \Delta(Y, y^n) - \underbrace{\xi_n}_{\text{slack}}$$

$$\xi_n \geq 0$$

$$\forall n, Y \in \mathcal{Y}$$

Theorem (Tsochantaridis et al., 2005): $\Delta(Y_*^n, y^n) \leq \xi_n^*$
where $Y_*^n = \operatorname*{argmax}_{Y} \langle \Phi(x^n, Y), \theta^* \rangle$

The Convex Relaxation

$$[\theta^*, \xi^*] = \operatorname*{argmin}_{\theta,\xi} \left[\frac{1}{N} \sum_{n=1}^{N} \xi_n + \lambda \left\| \theta \right\|^2 \right]$$

$$\text{s.t. } \underbrace{\langle \Phi(x^n, y^n), \theta \rangle - \langle \Phi(x^n, Y), \theta \rangle}_{\text{margin}} \geq \Delta(Y, y^n) - \underbrace{\xi_n}_{\text{slack}}$$

$$\xi_n \geq 0$$

$$\forall n, Y \in \mathcal{Y}$$

Theorem (Tsochantaridis et al., 2005): $\Delta(Y_*^n, y^n) \leq \xi_n^*$
where $Y_*^n = \operatorname*{argmax}_{Y} \langle \Phi(x^n, Y), \theta^* \rangle$

Missing Labels

- Problem: $\Phi(x, y^n)$ is not directly comparable to $\Phi(x, Y)$
- This is because y^n has only one label while Y has 5
- Solution: define $Y^n := (y^n, z_1^n, z_2^n, z_3^n, z_4^n)$, where $z^n = (z_1^n, z_2^n, z_3^n, z_4^n)$ is a vector of latent variables
- Use latent structured learning (Yu and Joachims, 2009)
- Alternate optimisation over z^n and θ

Latent Structured Learning

- If the latent variables are observed, we can perform structured learning as usual
- Having learned a model, we can estimate new values for the latent variables
- Alternating between these steps is guaranteed to monotonically decrease the objective and reach a local minimum
- The 'boosted' model is guaranteed to perform at least as well as the original classifier (at least on the training set!)

First Problem: Infering Missing Labels

- Optimisation over z^n can be done greedily since Φ decomposes linearly over z_i^n:

$$
\begin{aligned}
z_*^n &:= \operatorname*{argmax}_{z^n} \langle \Phi(x^n, (y^n, z^n)), \theta \rangle \\
&= \operatorname*{argmax}_{z^n : z_i^n \neq y^n} \left\langle \phi(x^n, y^n) + \sum_i \phi(x^n, z_i^n), \theta \right\rangle \\
&= \operatorname*{argmax}_{z^n : z_i^n \neq y^n} \sum_i \langle \phi(x^n, z_i^n), \theta \rangle
\end{aligned}
$$

- Which is identical to the prediction problem, but restricted to 4 classes distinct from y^n, and therefore can be easily solved in linear time.

Second Problem: Constraint Generation

- With Y^n 'completed', we can optimise for θ:

$$
\begin{aligned}
&[\theta^*, \xi^*] = \operatorname*{argmin}_{\theta,\xi} \left[\frac{1}{N} \sum_{n=1}^{N} \xi_n + \lambda \|\theta\|^2 \right] \\
&\text{s.t. } \langle \Phi(x^n, Y^n), \theta \rangle - \langle \Phi(x^n, Y), \theta \rangle \geq \Delta(Y, y^n) - \xi_n \\
&\xi_n \geq 0 \\
&\forall n, Y \in \mathcal{Y}
\end{aligned}
$$

- There are $\binom{1000}{5} \times N + N$ constraints $\approx 10^{13} \times N \rightarrow$ too many
- Use constraint generation

Second Problem: Constraint Generation

- Constraint generation amounts to finding the constraint $\hat{Y}^n$ maximising the violation margin ξ_n, which consists of solving

$$\hat{Y}^n = \operatorname*{argmax}_{Y \in \mathcal{Y}} \left\{ \min_{y \in Y} d(y, y^n) + \sum_{y \in Y} \langle \phi(x^n, y), \theta \rangle \right\}$$

Naively: requires enumeration of $\binom{1000}{5} \approx 10^{13}$ states

$\boxed{\text{How to solve this efficiently?}}$

Second Problem: Constraint Generation

- Assume we knew $c = \operatorname{argmin}_{y \in \hat{Y}^n} d(y, y^n)$
- Then the problem becomes

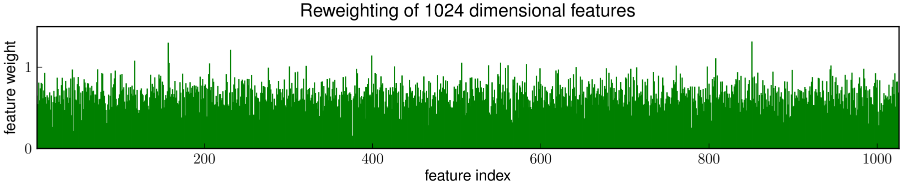
$$\hat{Y}^n = \operatorname*{argmax}_{Y \in \mathcal{Y}'} \left\{ d(c, y^n) + \sum_{y \in Y} \langle \phi(x^n, y), \theta \rangle \right\}$$

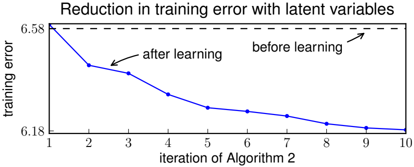
- Where $\mathcal{Y}'$ is simply $\mathcal{Y}$ restricted to include c and only containing other y respecting $d(y, y^n) \geq d(c, y^n)$).
- This can be solved by finding the 4 classes y with largest score $\langle \phi(x^n, y), \theta \rangle$: linear in # classes
- Of course we don't know c, so we have to try this for all possible c and pick the max: linear in # classes
- Total complexity: quadratic in # classes

Implementation Speed-ups

- We note that in the challenge dataset, $d(y, y^n) \in \{0, \ldots, 18\}$
- This means that $d(c, y^n)$ we can only attain 19 values
- Speeds-up constraint generation from $O(1000 \times 1000)$ to $O(19 \times 1000)$
- Also the inner products can be parallelised efficiently: GPU implementation

Results: 1024-dimensional feature vector

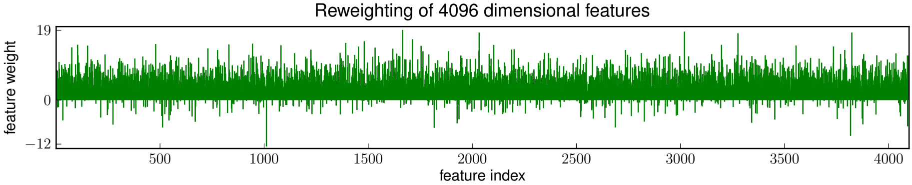


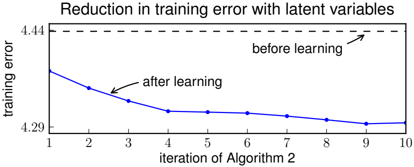


Test error:

	1nn	2	3	4	5
Before learning	11.35	9.29	8.08	7.25	6.64
After learning	10.88	8.85	7.71	6.93	6.36

Results: 4096-dimensional feature vector





Test error:

	1nn	2	3	4	5
Before learning	9.27	7.29	6.23	5.53	5.03
After learning	9.02	7.08	6.05	5.38	4.91

Final Remarks

- Can the taxonomy improve classification? Yes
- Our results are still not as good as the winners
- If we had their features, we might be able to boost their own results... but by how much?

Other Applications

Latent-variable Structured Learning appears to be useful when we have weak labels

Possible Applications

Application	Full Labelling	Weak Labelling
Classification	multiple labels	one label
Segmentation	label for each pixel	bounding box
Ranking	rank of each document	relevance of each document
Correspondence	match between parts	match between objects

Conclusion

- Structured energies, and structured error measures are natural for many computer vision problems
- 'Simple' classification schemes often fail to exploit this structure
- Structured learning aims to solve this problem, but may require rich labels that are expensive to produce
- Latent structured learning may allow us to apply structured learning techniques when rich labels are not available

Bibliography

Alex Berg, Jia Deng, and Fei-Fei Li. Imagenet large scale visual recognition challenge 2010. `http://www.image-net.org/challenges/LSVRC/2010/index`, 2010.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

Yuanqing Lin, Fengjun Lv, Shenghuo Zhu, Ming Yang, Timothee Cour, and Kai Yu. Large-scale image classification: fast feature extraction and SVM training. In *IEEE Conference on Computer Vision and Pattern Recognition*, page (to appear), 2011.

Jorge Sánchez and Florent Perronnin. High-Dimensional Signature Compression for Large-Scale Image Classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, page (to appear), 2011.

Ioannis Tsochantaridis, Thorsten Joachims, Thomas Hofmann, and Yasemin Altun. Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research*, 6:1453–1484, 2005.

Chun-Nam John Yu and Thorsten Joachims. Learning structural svms with latent variables. In *International Conference on Machine Learning*, 2009.