Self-Paced Learning for Semisupervised Image Classification

Kevin Miller
Stanford University
Palo Alto, CA
kjmiller@stanford.edu

Second Author
Institution2
First line of institution2 address
secondauthor@i2.org

Abstract

In this project, I plan to apply self-paced learning to the bounding-box problem using the VOC2011 dataset.

1. Preface

I'm going to assume that this is where I am supposed to put my actual report. Please let me know if it was supposed to go somewhere else (so I don't botch the final report).
Also, I'm currently working on this project as a research project for Daphne Koller's lab. I'm working with Rafi Witten, who is an undergrad who isn't taking CS231A. In this report, I'll mostly stick to things that I implemented on my own (or collaborated on very closely with Rafi), but there are a few things that Rafi worked on that I think are worth mentioning because they're interesting and relevant.

2. Problem Description

In the problem at hand, there is a set of images $x_1, ..., x_M$, each of which contains one of 20 objects (e.g. cars, sheep, humans, etc.). Let $y_1, ..., y_M$ denote the object that each image contains (e.g. $y_1 = 1$ means that image 1 contains a car). During learning, x and y are known for each image, and during classification, we want to predict y from x.

3. Technical Approach

The simplest approach would be to use a completely supervised algorithm, such as Structural SVM. However, semisupervised algorithms tend to perform much better by using latent variables to reduce noise. In this project, Latent Structural SVM is used; for each image x_i, the latent variable h_i represents the location and dimensions of a bounding box around (what is hopefully) the object in the picture.

One problem with Latent Structural SVM is the fact that training an LSSVM involves optimizing a non-convex function, meaning that the algorithm can end get stuck in bad local optima. Daphne Koller, M. Pawan Kumar, and Ben Packer recently showed that LSSVM could be improved by ignoring "difficult" examples during each inner loop of the LSSVM optimization algorithm; specifically, examples with slack above some threshold would be ignored, and between each inner loop this threshold would be increased until all examples were included and the algorithm converged. This extension of LSSVM is known as Self-Paced Learning. If each inner loop of LSSVM is viewed as a Structural SVM problem, then Self-Paced Learning can be seen as making each of these Structural SVM's more robust to outliers.

The problem we deal with in this project is complicated by the fact that the images in the dataset often contain multiple types of objects (e.g. a person and a car). Our baseline is therefore a bit different from the normal LSSVM algorithm. Each time an image has multiple correct classes (say m correct classes), we create m duplicates of that image and feed the algorithm a different "correct" class for each duplicate. However, for each duplicate, we also create a "whitelist" of classes that are also correct, and the inner loop of our LSSVM doesn't include "whitelist" classes in the SSVM constraints. This means that if an image contains both a person and a car, our algorithm won't penalize a model based on the score it gives to the "car" prediction relative to the score given to the "person" prediction. This does admittedly make classification a bit more complicated, since it's hard to define the accuracy of a classifier in this situation, so instead we will use the classifier's scores for each class to compute an average precision score for each one and use the mean average precision as our evaluation metric. Thus, it will be as if we were separately evaluating 20 binary classifiers.

We ultimately plan to test out three different levels of self-paced learning on this problem. First we will try out a modified version of the SPL algorithm that Koller, Kumar, and Packer first discovered. In the original algorithm, there was a term in SSVM objective that rewarded the inclusion of examples, which would offset the extra penalty created by that example's slack. In our version, we will replace the reward term with a constraint on the number of examples for each class that are included and increase this number as the algorithm progresses. We used this version during the summer after finding that normal SPL (and especially SPL+ - to be described momentarily) had problems when the dataset was unbalanced (as our current dataset will undoubtedly be).

The next level currently goes by the name "SPL+"; in this level, the algorithm can include or ignore different feature sets for different images. Intuitively, this should be useful in situations where different images have different features that are "difficult" (e.g. an image of a car with a weird shape but normal texture, and an image of a car with normal shape but weird texture). We include or exclude (image, feature) pairs by turning different parts of each images's feature vector on and off, and scaling the margin term by the number of features included for each image. For each (class, feature) pair, we have an equality constraint on the number of images with that particular class for which that particular feature is included; having a strict constraint is helpful because it prevents situations where e.g. only cars have the shape feature included, only sheep have the color feature included, etc., and nothing useful is learned. SPL+ was implemented over the summer as well, but for a binary classification problem on a smaller dataset with fewer features.

Finally, we hope to implement an aglorithm that goes by the name "SPL++". SPL++ is like SPL+, except that now we add a degree of freedom by including and excluding different features for different class-comparison constraints for different images. For example, if we have an image of a car, then we might include color when comparing the "car" prediction to the "sheep" prediction but avoid including color when comparing the "car" prediction to the "bus" prediction. Again, we constrain the number of examples included for each (class, class to compare to, feature, image) 4-tuple.

In all of the variants of SPL, there's a relatively easy way to decide what is included and excluded. For SPL, we can simply sort the examples by their slack values and include the ones with the k lowest slack. For SPL+ and SPL++ we will do a greedy search by choosing the examples to include for one (feature, class) or (feature, class, class to compare to) tuple at a time.

4. Progress So Far

Unfortunately, we have not yet implemented any self-pace learning yet; we are currently working on rewriting the baseline in python (we felt that the C code was too unwieldy and unreliable to work with). We have been spending a lot of time trying to make the baseline algorithm run at a reasonable speed for VOC datasets from previous years and are almost ready to move to VOC2011. We'll be training on thousands of images, each one with hundreds of possible bounding boxes and 20 possible classes, so we've had to make some optimizations; these include caching our feature vectors to disk (in a way that makes them quickly retrievable), parallelizing parts of the cutting plane algorithm used to solve the LSSVM inner loop, and dropping inactive constraints from the cutting plane algorithm. Rafi has been working on the first two of these, while I've been working on the third.
Hopefully we'll be able to make a good amount of progress over Thanksgiving break and have some interesting results by December 15.

5. Appendix

Since this project is part of a larger research project, below is a description of who did what.
SPL was originally formulated by Koller, Kumar, and Packer. Rafi and I collaborated over the summer on implemented our first pass at SPL+, and I subsequently worked alone on devising and experimenting with different objectives and optimization strategies for SPL+ over the summer. I'll probably do a lot of the implementation for SPL+ and SPL++ since I've spent the most time with it over the summer, although it's very likely that I'll collaborate with Rafi on some of it. It really depends on what level of collaboration allows us to get the most done in the least amount of time.
Rafi has been working alone on the feature computation pipeline and on some of the speedups (mentione in previous section). We've both collaborated closely on getting the main LSSVM algorithm to work.
Needless to say, Koller, Kumar, and Packer have been supervising us and giving us advice.

6. References

http://www.cs.cornell.edu/∼cnyu/papers/icml09_latentssvm.pdf
http://ai.stanford.edu/∼pawan/publications/KPK-NIPS2010.html

Future Distribution Permission

The author(s) of this report give permission for this doc- ument to be distributed to Stanford-affiliated students taking future courses.

7. Introduction

Please follow the steps outlined below.

7.1. Language

All manuscripts must be in English.

7.2. The ruler

The LaTeX style defines a printed ruler which should be present in the version submitted for review. The ruler is provided in order that reviewers may comment on particular lines in the paper without circumlocution. If you are preparing a document using a non-LaTeX document preparation system, please arrange for an equivalent ruler to appear on the final output pages. The presence or absence of the ruler should not change the appearance of any other content on the page. The camera ready copy should not contain a ruler.

7.3. Mathematics

Please number all of your sections and displayed equations. It is important for readers to be able to refer to any particular equation. Just because you didn't refer to it in the text doesn't mean some future reader might not need to refer to it. It is cumbersome to have to use circumlocutions like "the equation second from the top of page 3 column 1". (Note that the ruler will not be present in the final copy, so is not an alternative to equation numbers). All authors will benefit from reading Mermin's description of how to write mathematics.

7.4. Miscellaneous

Compare the following:

`$conf_a$`	$conf_a$
`$\mathit{conf}_a$`	conf_a

See The TeXbook, p165.

The space after *e.g.*, meaning "for example", should not be a sentence-ending space. So *e.g.* is correct, *e.g.* is not. The provided `\eg` macro takes care of this.

When citing a multi-author paper, you may save space by using "et alia", shortened to "*et al.*" (not "*et. al.*" as "*et*" is a complete word.) However, use it only when there are three or more authors. Thus, the following is correct: " Frobnication has been trendy lately. It was introduced by Alpher [**?**], and subsequently developed by Alpher and Fotheringham-Smythe [**?**], and Alpher *et al.* [**?**]."

This is incorrect: "... subsequently developed by Alpher *et al.* [**?**] ..." because reference [**?**] has just two authors. If you use the `\etal` macro provided, then you need not worry about double periods when used at the end of a sentence as in Alpher *et al.*

For this citation style, keep multiple citations in numerical (not chronological) order, so prefer [**?**, **?**, **?**] to [**?**, **?**, **?**].

8. Formatting your paper

All text must be in a two-column format. The total allowable width of the text area is $6\frac{7}{8}$ inches (17.5 cm) wide by $8\frac{7}{8}$ inches (22.54 cm) high. Columns are to be $3\frac{1}{4}$ inches (8.25 cm) wide, with a $\frac{5}{16}$ inch (0.8 cm) space between them. The main title (on the first page) should begin 1.0 inch (2.54 cm) from the top edge of the page. The second and following pages should begin 1.0 inch (2.54 cm) from the top edge. On all pages, the bottom margin should be 1-1/8 inches (2.86 cm) from the bottom edge of the page for 8.5×11-inch paper; for A4 paper, approximately 1-5/8 inches (4.13 cm) from the bottom edge of the page.

Figure 1. Example of a short caption, which should be centered.

8.1. Margins and page numbering

All printed material, including text, illustrations, and charts, must be kept within a print area 6-7/8 inches (17.5 cm) wide by 8-7/8 inches (22.54 cm) high.

8.2. Type-style and fonts

Wherever Times is specified, Times Roman may also be used. If neither is available on your word processor, please use the font closest in appearance to Times to which you have access.

MAIN TITLE. Center the title 1-3/8 inches (3.49 cm) from the top edge of the first page. The title should be in Times 14-point, boldface type. Capitalize the first letter of nouns, pronouns, verbs, adjectives, and adverbs; do not capitalize articles, coordinate conjunctions, or prepositions (unless the title begins with such a word). Leave two blank lines after the title.

AUTHOR NAME(s) and AFFILIATION(s) are to be centered beneath the title and printed in Times 12-point, non-boldface type. This information is to be followed by two blank lines.

The ABSTRACT and MAIN TEXT are to be in a two-column format.

MAIN TEXT. Type main text in 10-point Times, single-spaced. Do NOT use double-spacing. All paragraphs should be indented 1 pica (approx. 1/6 inch or 0.422 cm). Make sure your text is fully justified—that is, flush left and flush right. Please do not place any additional blank lines between paragraphs.

Figure and table captions should be 9-point Roman type as in Figures **??** and 1. Short captions should be centred.

Callouts should be 9-point Helvetica, non-boldface type. Initially capitalize only the first word of section titles and first-, second-, and third-order headings.

FIRST-ORDER HEADINGS. (For example, **1. Introduction**) should be Times 12-point boldface, initially capitalized, flush left, with one blank line before, and one blank line after.

SECOND-ORDER HEADINGS. (For example, **1.1. Database elements**) should be Times 11-point boldface, initially capitalized, flush left, with one blank line before, and one after. If you require a third-order heading (we discourage it), use 10-point Times, boldface, initially capitalized, flush left, preceded by one blank line, followed by a period and your text on the same line.

8.3. Footnotes

Please use footnotes[1] sparingly. Indeed, try to avoid footnotes altogether and include necessary peripheral observations in the text (within parentheses, if you prefer, as in this sentence). If you wish to use a footnote, place it at the bottom of the column on the page on which it is referenced. Use Times 8-point type, single-spaced.

[1]This is what a footnote looks like. It often distracts the reader from the main flow of the argument.

Method	Frobnability
Theirs	Frumpy
Yours	Frobbly
Ours	Makes one's heart Frob

Table 1. Results. Ours is better.

8.4. References

List and number all bibliographical references in 9-point Times, single-spaced, at the end of your paper. When referenced in the text, enclose the citation number in square brackets, for example [**?**]. Where appropriate, include the name(s) of editors of referenced books.

8.5. Illustrations, graphs, and photographs

All graphics should be centered. Please ensure that any point you wish to make is resolvable in a printed copy of the paper. Resize fonts in figures to match the font in the body text, and choose line widths which render effectively in print. Many readers (and reviewers), even of an electronic copy, will choose to print your paper in order to read it. You cannot insist that they do otherwise, and therefore must not assume that they can zoom in to see tiny details on a graphic.

When placing figures in LaTeX, it's almost always best to use `\includegraphics`, and to specify the figure width as a multiple of the line width as in the example below

```
\usepackage[dvips]{graphicx} ...
\includegraphics[width=0.8\linewidth]
                {myfile.eps}
```

8.6. Color

Color is valuable, and will be visible to readers of the electronic copy. However ensure that, when printed on a monochrome printer, no important information is lost by the conversion to grayscale.

9. Appendix

If your course project is part of a larger project from another class or research lab, please fill in this section and clearly spell out the following items:

1. Explicitly explain what the computer vision components are in this course project;
2. Explicitly list out all of your own contributions in this project in terms of:
   (a) ideas
   (b) formulations of algorithms
   (c) software and coding
   (d) designs of experiments
   (e) analysis of experiments
3. Verify and confirm that you (and your partner currently taking CS231A) are the sole author(s) of the writeup. Please provide papers, theses, or other documents related to this project so that we can compare with your own writeup.