Using Convolutional Neural Networks to automate data entry of paper surveys

Kristofer Walker, Robin A. Donatello DrPH

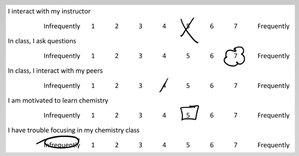
Department of Mathematics and Statistics, California State University, Chico

INTRODUCTION

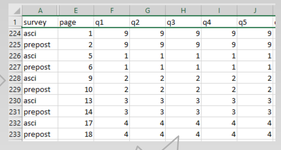
- Commonly used for their high response rates, paper surveys are easy to administer for instructors used to creating, distributing, and collecting class handouts (McNulty, 2008).
- To assess the effectiveness of a major overhaul in how General Chemistry is taught, paper assessment surveys were administered before and after the semester, for 4 years.
- Chemistry professors created and administered the survey using MS Word document for ease of creation, printing, and in-class administration. Data entry for analysis was an afterthought.
- This work was completed as part of a Summer Research Program for the College of Natural Sciences at California State University, Chico.

GOALS

Turn these

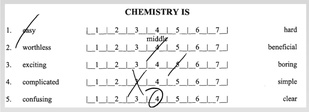


into this



	A	E	F	G	H	I	J
1	survey	page	q1	q2	q3	q4	q5
224	asci	1	9	9	9	9	9
225	prepost	2	9	9	9	9	9
226	asci	5	1	1	1	1	1
227	prepost	6	1	1	1	1	1
228	asci	9	2	2	2	2	2
229	prepost	10	2	2	2	2	2
230	asci	13	3	3	3	3	3
231	prepost	14	3	3	3	3	3
232	asci	17	4	4	4	4	4
233	prepost	18	4	4	4	4	4

Analyzable data format!



1,500 students, 60 questions, 8 semesters = 720,000 data points
(not including identifiers such as class, section, year, pre vs post, survey ID)

BACKGROUND & RATIONAL FOR NEW TOOLS

- Available services such as flatworldsolutions.com and hyperscience.com expect data to be prerecorded in online databases, worksheets, or product listings.
- These services also require forms with fields containing expected values or information typical of employment/financial applications and healthcare/disability forms.
- Tools like Tabula that extract data from a PDF expect a tabular structure to return a spreadsheet.
- Outsourcing data entry is not scalable and requires significant overhead.
- Automated PDF extraction software using OCR and/or enterprise ready document processing and workflow platforms require forms with fields containing expected values or information
- Document parsers are great for extracting text and numerical information from various documents, but we are not dealing with traditional text or image input.

DEEP LEARNING FOR IMAGE PROCESSING

- ***Deep Learning:*** subfield of machine learning that maps the input to target using successive layers of data transformation.
- ***Loss score*:** How far are the model predictions away from the true values? (larger score = further away)
- ***Optimizer*:** Implements the *back-propagation* algorithm to adjust the weights, which will result in a lower loss score (increased prediction accuracy)
  - Gradient based optimization
  - ***Back-propagation*** (aka. reverse-mode differentiation) : chain rule applied to the gradient values of neural network
- All layers are estimated jointly – not successively.

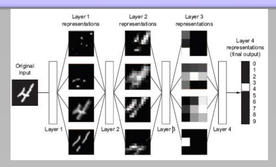


Fig 1. Deep representations learned by a digit-classification model

Example layer:

$X' = max(X \cdot W + b, 0)$

```
W : 2D tensor
b : vector of weights
X : input
X': output
```

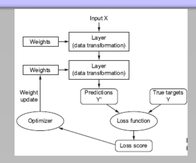


Fig 2. The loss score is used as a feedback signal to adjust weights.

Stochastic gradient (∇) descent

1. Draw a sample of training data (x) and true values (y)
2. Run network and calculate y_{pred}
3. Compute the loss on the batch:$y_{pred} - y$
4. Compute the ∇ of *loss* wrt network parameters (W)
5. Move parameters a little in the opposite direction of the gradient: $W = W - (step * \nabla)$

METHODS (data pre-processing)

1. **Recording:** Training data is created by manually filling out responses on the paper surveys.
2. **Scanning:** Physical surveys are scanned to PDF
3. **Assign Labels:** Training data is manually recorded into a spreadsheet, each row an observation, each column corresponds to a survey question such that there is a one-to-one correspondence between the PDF and row in our spreadsheet.
4. **Preprocessing:**
   - PDF image files are read into R and converted to a matrix of binary indicators for the color of each pixel (0=white, 1=black).
   - The edges of the matrix (indicating words) are cropped out, and rows are sliced up to creates "strips" for each question, isolating the field where a labeled response exists.

- The "strips" are then transformed into a *tensor* array format that conforms with the CNN expected array shape.
- *Tensors:* generalization of vectors/matrices into arbitrary number of dimensions. A typical matrix is a 2D Tensor.

METHODS (modeling)

***Convolutional neural networks (CNN)*:**

- Special type of deep learning model typically used in computer vision applications.
- Layers learn local representations and patterns found in small 2D windows of their inputs.
- These local representations are translation invariant and need fewer training samples to learn representations that generalize well.

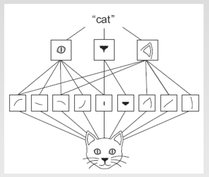


Fig 3. CNN's learn the spatial hierarchies of patterns necessary to learn increasingly complex and abstract visual concepts.

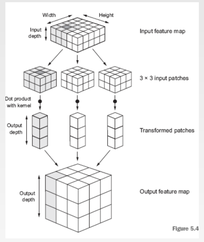
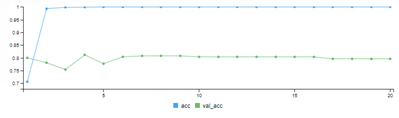


Fig 4. Convolution operations extract patches from input feature maps which are 3D tensors (array of 2D matrices), applying the same transformation to produce an output feature map – another 3D tensor.

RESULTS

- Both models: 5,200 training sample strips, 100 strips set aside for external validation
- Amazon Web Service EC2 instance with TensorFlow(+Keras2) with Python3 NVIDIA(CUDA 10.0 and Intel MKL-DNN)
- Both models have 3 convolution layers while the Lenet model has a dropout (0.5) and an optimizer learning rate of .0001.

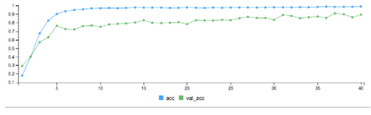
- Small CNN: 3 convolution layers
- 4680 samples across 20 epochs



91% Accuracy

3:26 minutes

- Lenet CNN: 3 convolution layers
- 4680 samples across 40 epochs



89% Accuracy

5:41 minutes

CONCLUSIONS / IMPLICATIONS

This research successfully demonstrates that using machine learning techniques such as Convolution Neural Networks can successfully capture survey data collected using paper and pencil surveys with moderately high accuracy.

Potential drawbacks

- To run and/or adjust the model, knowledge of CNNs are required
- Sufficient quantity and variety of training data takes time to create.
- Deep learning at scale requires cloud-server quality computing capabilities. This can come with a cost:
  - 6-8 hours of model development on AWS can cost ~$10-$15
  - a few days of intense training/modeling can cost 2-3x that amount.

Potential benefits

- Reduced time for data entry.
- Reduced data entry error rate
- Customizable for any ad-hoc survey of similar form that may be created.
- Deep Learning models have consistently shown promising results in many fields and benefit from their *simplicity* (layers are auto-tuned), *scalability* (easy to parallelize), and *versatility* (easy to use one fully trained model on new output)

NEXT STEPS

- Experiment with Amazon Deep Lens to standardize PDF image scale input.
- Experiment with Amazon Rekognition pre-built deep learning models– now that you can provide custom labels.
- Build more training data! The large variety of marks require a large variety of training data.
- Assess if existing model is being overfit
- Refactor code for production at scale
- Evaluate algorithm performance against manual data entry
  - Can this method have equal to or superior accuracy?
  - How many surveys are necessary before any time-saving benefits from automation is realized, given time required for creating training data and tuning the model.
  - What is the cost savings compared to hiring data entry personnel?

SOFTWARE / REFERENCES

- R version 3.4.4: R Core Team (2018). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/
- JJ Allaire and François Chollet (2019). keras: R Interface to 'Keras'. R package version 2.2.5.0. https://CRAN.R-project.org/package=keras
- JJ Allaire and Yuan Tang (2019). tensorflow: R Interface to 'TensorFlow'. R package version 2.0.0. https://CRAN.R-project.org/package=tensorflow
- Simon Barthelme (2019). imager: Image Processing Library Based on 'CImg'. R package version 0.41.2. https://CRAN.R-project.org/package=imager
- Jeroen Ooms (2019). magick: Advanced Graphics and Image-Processing in R. R package version 2.2.https://CRAN.R-project.org/package=magick
- Jeroen Ooms (2019). pdftools: Text Extraction, Rendering, and Converting of PDF documents. R package version 2.2. https://CRAN.R-project.org/package=pdftools.
- Hadley Wickham and Lionel Henry (2019). tidyr: Tidy Messy Data. R package version 1.0.0. https://CRAN.R-project.org/package=tidyr
- AWS Deep Learning AMI with NVIDIA CUDA 10.0
- Chollet F, Allaire J, Deep Learning with R (Also Fig 1-Fig4 screenshot)
- McNulty, Duncan D. "*The adequacy of response rates to online and paper surveys: What can be done?*" Assessment & Evaluation in Higher Education:Vol33, No.3, June 2008, 301-314

ACKNOWLEDGEMENTS

- Chico Stem Connections Collaborative for funding this research,
- California State University, Chico: College of Natural Sciences, Department of Chemistry and Biochemistry
- Lisa Kendhammer PhD (CSU, Chico Chemistry) for providing the surveys and consulting.

CONTACT

kswalker@mail.csuchico.edu rdonatello@csuchico.edu