Predicting mental functions from brain activations

Madhura Parikh
Department of Computer Science
University of Texas, Austin
mparikh@cs.utexas.edu

Subhashini Venugopalan
Department of Computer Science
University of Texas, Austin
vsubhashini@utexas.edu

Abstract

Over the past few decades neuroscientists have studied brain images from EEG/MEG, fMRI and other sources to identify associations between psychological tasks and activity in brain regions [**?**]. Although these studies have led to large amounts of literature and several discoveries of cognitive functions associated with certain brain regions (or networks) the mapping between functions to brain regions and vice-versa still remains largely unclear. For the purposes of this project, we look at enhancing a new automated framework NeuroSynth [**?**] that combines text-mining and machine learning techniques to generate probabilistic mappings between cognitive and neural states. Starting from their Naive Bayes classifier, we apply more sophisticated binary classifiers to the problem and also consider multi-label predictions and transfer-learning(?).

1 Introduction and Related work

In this project, we build on the existing NeuroSynth framework [1]. While the NeuroSynth framework offers tools for several types of meta-analyses, we primarily address the problem of Reverse Inference. This can be stated more precisely as: *Given a signature of neural activity, identify the cognitive state(s) and functions that the activations correspond to* (see fig 1). The scientific community typically uses fMRI scans for reporting this neural activity. Reverse inference is an extremely challenging problem since multiple cognitive states could have very similar neural signatures [**?**] but it is also of major interest to the neourimaging community at large.

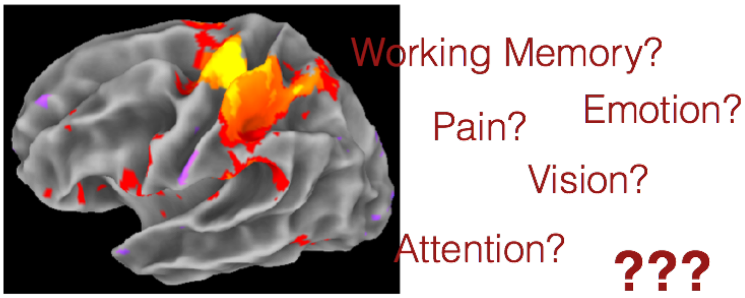


Figure 1: The reverse inference problem

Forward and reverse inference problems have been addressed by several contemporary works [**?**, **?**, **?**, **?**]. Previous approaches have generally tackled the Reverse Inference problem by manually analyzing fMRI scans of subjects, collected from the laboratory. There are several limitations of

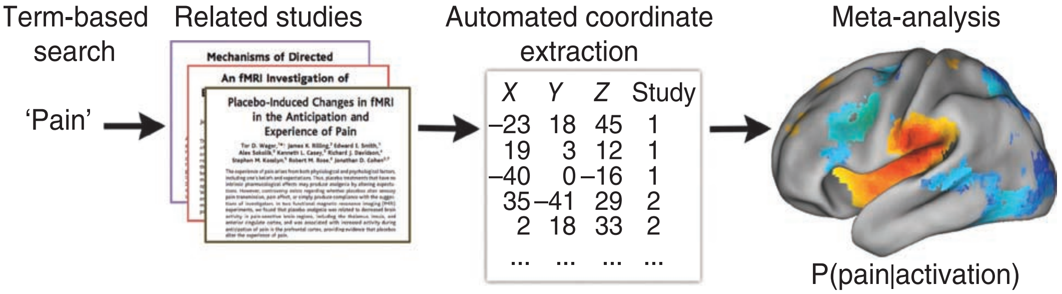
[1] neurosynth.org

such an approach - for instance, involving human subjects for fMRI scans is labor and cost intensive and the number of data samples that can be gained from such efforts is also very less. Moreover all the meta-analyses based on such data is carried on a very small-scale at individual research labs, and fails to take advantage of the vast knowledge embodied in the entire research community.

NeuroSynth's (and therefore our) approach is unique - in that we tackle the Reverse Inference problem not by requiring actual fMRI scans, but rather by exploiting the relatively large repository of neuro-imaging publications using text-mining and machine learning techniques. There are many motivations and benefits that lead to this. For one, while fMRI scans are very few, there has been a growing body of publications related to neuro-imaging, thus offering a much larger source of data. Further by using machine learning techniques the decoding is possible without any real training data (fMRI scans) and at the same time incorporates the knowledge base derived from several researcher. Also to the best of our knowledge, this is the first approach that is fully automated, thus making it possible to perform several meta-analyses on a much larger scale than could ever be possible by individual researchers. In the next few paragraphs, we introduce the NeuroSynth framework and some of the techniques it utilizes.

1.1 The NeuroSynth framework

The figure 2 gives a high-level view of NeuroSynth.



X	Y	Z	Study
–23	18	45	1
19	3	12	1
–40	0	–16	1
35	–41	29	2
2	18	33	2
...	...	...	...

Figure 2: The NeuroSynth framework [**?**]

A detailed description of NeuroSynth may be found in [**?**]. Here we give only a high-level view - figure 2. For reverse inference, NeuroSynth performs the following three steps:

Step 1: Extract high frequency terms First, a database of nearly 3000-odd [2] studies is scrapped to extract the most frequent terms and their frequency of occurence across these studies. Thus corresponding to each study we get a set of terms. These set of terms serve as labels for that study, during the classification.

Step 2: Extract coordinates of activation foci and synthesize sparse image Using simple template matching, all probable activation foci mentioned in a study are extracted. Once we have a list of these coordinates, corresponding to each study, they are used to synthesize extremely sparse brain images with the corresponding brain regions activated. After some preprocessing (which we describe in detail in section 2.1) the vectorized image serves as the feature vector for that study.

Step 3: Use classification to build a predictive model Use classification to train a model, give the labels and the feature vectors from steps (1) and (2). They use an extremely simple approach in which they apply a Naive Bayes classifier to make single-label predictions . They pick up 25 of the most frequent terms arbitarily and train $\binom{25}{2}$ models (i.e one-vs-one) corredponding to every term pair and 10-fold cv to make the predictions.

There have been other approaches that have further built on NeuroSynth. For instance in [**?**], the authors use label decomposition techniques in a one-vs-all setting to predict multiclass labels for the reverse inference problem. They use Support Vector Machines (l_2 regularized), logistic regression

[2] The number has grown to over 8000, since 2011 when [**?**] was published.

and ridge classifier for these tasks, comparing the outcomes based on different criteria like precision, recall and Hamming loss, to show that the multiclass approach is effective for reverse inference. They also propose to use other multiclass approaches and regularization techniques and analyze their performance as future work.

In another similar work [**?**] uses a Generalized Linear Model(GLM) for forward inference. For the reverse inference they use logistic regression with Ward clustering to counter the high dimensionality of the problem.

Building up on these approaches, in the following sections, we shall describe our own extensions and experiments toward building a better model for the reverse inference problem.

2 Data

Similar to [**?**], we primarily use the repository available with NeuroSynth for our project. This has nearly 8067 studies that are scrapped from online resources, spanning 15 journals. For our transfer-learnign approach, we would like to use real fMRI scans, and adapt our models tuned with the synthesized data on this real data. We use data from 2 sources. We use the open source openfMRI [3] project [**?**] to gain the training data. This has individual subject level images for 26 contrasts, resulting in a total of 479 data samples. Since it is always more advisable to work with group level statistics, we also consider the NeuroVault[4] data that has 17 studies, but with group level images. Ideally we would have liked to use this data entirely but the major issues is that it is too less to be meaningful on its own.

2.1 Pre-processing

Here we describe mainly the preprocessing pipe for the NeuroSynth data. The code repository for NeuroSynth already has the labels and coordinates that are extracted from the 8000 studies. Beginning with this, we used various open source toolkits like NumPy tand existing utiliities in NeuroSynth o synthesize images corresponding to the the activated coordinates.

Step 1: First for each (x, y, z) coordinate extracted from a study, we transformed it to the MNI space - which is the standard space used for neuro-imaging.

Step 2: Second, since the coordinates were extracted using simple template matching, there would likely be some spurious numbers that were mistaken as coordinates. It is also possible that the study mentioned coordinates for some different region other than the brain. To deal with such anomalies, we then applied a 2mm MNI mask on the coordinates, to validate that they indeed lay in the brain space and discarded the invalid ones. The 3D mask has the shape $(91x109x91)$

Step 3: Since each study mentions only a very few coordinates from the entre brain space, the resulting image we would get would be extremely sparse. Based on connectivity and correlation amongst the brain regions, given an activated focus, it is highly likely that the voxels in its neighbourhood would be actvated as well. Using this fact we considered all voxels in a 6 mm radius around the activated foci to also be activated and thus get a more dense and smoother image.

Step 4: Once we get this synthesized image in 3D MNI space, we reshape it to a 1D vector, further removing all zero-columns, to end up with a $1x228453$ feature vector.

Since the data is extracted based on simple text processing, there will likely be irrelevant data that is also included. To deal with this we apply some further steps, in which we only consider terms that have a normalized frequency count > 0.001. Further we only consider those studies to be valid, that have atleast 500 activated voxels in the final feature vector. Further for single-label classification we only consider studies, that have a unique label with normalized frequency > 0.001. Studies that have multiple labels at a frequency > 0.001 cannot e clearly assigned a single correct label, and we therefore do not consider them for the single-label classification problem.

[3] https://openfmri.org

[4] http://neurovault.org

For the NeuroVault data we extracted various labels corresponding to each study by crawling the publication links mentioned at the online API, applying text processing with python to get the data in the required format. Similarly the preprocessing for openfMRI data was coding-intensive rather than of much value from a machine-learning perspective, so we do not further describe it here.

NIPS requires electronic submissions. The electronic submission site is

`http://papers.nips.cc`

Please read carefully the instructions below, and follow them faithfully.

2.2 Style

Papers to be submitted to NIPS 2013 must be prepared according to the instructions presented here. Papers may be only up to eight pages long, including figures. Since 2009 an additional ninth page *containing only cited references* is allowed. Papers that exceed nine pages will not be reviewed, or in any other way considered for presentation at the conference.

Please note that this year we have introduced automatic line number generation into the style file (for LaTeX $2_\varepsilon$and Word versions). This is to help reviewers refer to specific lines of the paper when they make their comments. Please do NOT refer to these line numbers in your paper as they will be removed from the style file for the final version of accepted papers.

The margins in 2013 are the same as since 2007, which allow for $\approx 15\%$ more words in the paper compared to earlier years. We are also again using double-blind reviewing. Both of these require the use of new style files.

Authors are required to use the NIPS LaTeX style files obtainable at the NIPS website as indicated below. Please make sure you use the current files and not previous versions. Tweaking the style files may be grounds for rejection.

2.3 Retrieval of style files

The style files for NIPS and other conference information are available on the World Wide Web at

`http://www.nips.cc/`

The file `nips2013.pdf` contains these instructions and illustrates the various formatting requirements your NIPS paper must satisfy. LaTeX users can choose between two style files: `nips11submit_09.sty` (to be used with LaTeX version 2.09) and `nips11submit_e.sty` (to be used with LaTeX2e). The file `nips2013.tex` may be used as a "shell" for writing your paper. All you have to do is replace the author, title, abstract, and text of the paper with your own. The file `nips2013.rtf` is provided as a shell for MS Word users.

The formatting instructions contained in these style files are summarized in sections 3, 4, and 5 below.

3 General formatting instructions

The text must be confined within a rectangle 5.5 inches (33 picas) wide and 9 inches (54 picas) long. The left margin is 1.5 inch (9 picas). Use 10 point type with a vertical spacing of 11 points. Times New Roman is the preferred typeface throughout. Paragraphs are separated by 1/2 line space, with no indentation.

Paper title is 17 point, initial caps/lower case, bold, centered between 2 horizontal rules. Top rule is 4 points thick and bottom rule is 1 point thick. Allow 1/4 inch space above and below title to rules. All pages should start at 1 inch (6 picas) from the top of the page.

For the final version, authors' names are set in boldface, and each name is centered above the corresponding address. The lead author's name is to be listed first (left-most), and the co-authors' names (if different address) are set to follow. If there is only one co-author, list both author and co-author side by side.

Please pay special attention to the instructions in section 5 regarding figures, tables, acknowledgments, and references.

4 Headings: first level

First level headings are lower case (except for first word and proper nouns), flush left, bold and in point size 12. One line space before the first level heading and 1/2 line space after the first level heading.

4.1 Headings: second level

Second level headings are lower case (except for first word and proper nouns), flush left, bold and in point size 10. One line space before the second level heading and 1/2 line space after the second level heading.

4.1.1 Headings: third level

Third level headings are lower case (except for first word and proper nouns), flush left, bold and in point size 10. One line space before the third level heading and 1/2 line space after the third level heading.

5 Citations, figures, tables, references

These instructions apply to everyone, regardless of the formatter being used.

5.1 Citations within the text

Citations within the text should be numbered consecutively. The corresponding number is to appear enclosed in square brackets, such as [1] or [2]-[5]. The corresponding references are to be listed in the same order at the end of the paper, in the **References** section. (Note: the standard BIBTEX style unsrt produces this.) As to the format of the references themselves, any style is acceptable as long as it is used consistently.

As submission is double blind, refer to your own published work in the third person. That is, use "In the previous work of Jones et al. [4]", not "In our previous work [4]". If you cite your other papers that are not widely available (e.g. a journal paper under review), use anonymous author names in the citation, e.g. an author of the form "A. Anonymous".

5.2 Footnotes

Indicate footnotes with a number[5] in the text. Place the footnotes at the bottom of the page on which they appear. Precede the footnote with a horizontal rule of 2 inches (12 picas).[6]

5.3 Figures

All artwork must be neat, clean, and legible. Lines should be dark enough for purposes of reproduction; art work should not be hand-drawn. The figure number and caption always appear after the figure. Place one line space before the figure caption, and one line space after the figure. The figure caption is lower case (except for first word and proper nouns); figures are numbered consecutively.

Make sure the figure caption does not get separated from the figure. Leave sufficient space to avoid splitting the figure and figure caption.

You may use color figures. However, it is best for the figure captions and the paper body to make sense if the paper is printed either in black/white or in color.

[5] Sample of the first footnote

[6] Sample of the second footnote

Figure 3: Sample figure caption.

Table 1: Sample table title

PART	DESCRIPTION
Dendrite	Input terminal
Axon	Output terminal
Soma	Cell body (contains cell nucleus)

5.4 Tables

All tables must be centered, neat, clean and legible. Do not use hand-drawn tables. The table number and title always appear before the table. See Table 1.

Place one line space before the table title, one line space after the table title, and one line space after the table. The table title must be lower case (except for first word and proper nouns); tables are numbered consecutively.

6 Final instructions

Do not change any aspects of the formatting parameters in the style files. In particular, do not modify the width or length of the rectangle the text should fit into, and do not change font sizes (except perhaps in the **References** section; see below). Please note that pages should be numbered.

7 Preparing PostScript or PDF files

Please prepare PostScript or PDF files with paper size “US Letter”, and not, for example, “A4”. The -t letter option on dvips will produce US Letter files.

Fonts were the main cause of problems in the past years. Your PDF file must only contain Type 1 or Embedded TrueType fonts. Here are a few instructions to achieve this.

- You can check which fonts a PDF files uses. In Acrobat Reader, select the menu Files>Document Properties>Fonts and select Show All Fonts. You can also use the program `pdffonts` which comes with `xpdf` and is available out-of-the-box on most Linux machines.
- The IEEE has recommendations for generating PDF files whose fonts are also acceptable for NIPS. Please see `http://www.emfield.org/icuwb2010/downloads/IEEE-PDF-SpecV32.pdf`

- LaTeX users:
  - Consider directly generating PDF files using `pdflatex` (especially if you are a MiKTeX user). PDF figures must be substituted for EPS figures, however.
  - Otherwise, please generate your PostScript and PDF files with the following commands:

    ```
    dvips mypaper.dvi -t letter -Ppdf -G0 -o mypaper.ps
    ps2pdf mypaper.ps mypaper.pdf
    ```

    Check that the PDF files only contains Type 1 fonts.
  - xfig ”patterned” shapes are implemented with bitmap fonts. Use ”solid” shapes instead.
  - The `\bbold` package almost always uses bitmap fonts. You can try the equivalent AMS Fonts with command

    ```
    \usepackage[psamsfonts]{amssymb}
    ```

    or use the following workaround for reals, natural and complex:

    ```
    \newcommand{\RR}{I\!\!R} %real numbers
    \newcommand{\Nat}{I\!\!N} %natural numbers
    \newcommand{\CC}{I\!\!\!\!C} %complex numbers
    ```
  - Sometimes the problematic fonts are used in figures included in LaTeX files. The ghostscript program `eps2eps` is the simplest way to clean such figures. For black and white figures, slightly better results can be achieved with program `potrace`.
- MSWord and Windows users (via PDF file):
  - Install the Microsoft Save as PDF Office 2007 Add-in from `http://www.microsoft.com/downloads/details.aspx?displaylang=en&familyid=4d951911-3e7e-4ae6-b059-a2e79ed87041`
  - Select “Save or Publish to PDF” from the Office or File menu
- MSWord and Mac OS X users (via PDF file):
  - From the print menu, click the PDF drop-down box, and select “Save as PDF...”
- MSWord and Windows users (via PS file):
  - To create a new printer on your computer, install the AdobePS printer driver and the Adobe Distiller PPD file from `http://www.adobe.com/support/downloads/detail.jsp?ftpID=204` *Note:* You must reboot your PC after installing the AdobePS driver for it to take effect.
  - To produce the ps file, select “Print” from the MS app, choose the installed AdobePS printer, click on “Properties”, click on “Advanced.”
  - Set “TrueType Font” to be “Download as Softfont”
  - Open the “PostScript Options” folder
  - Select “PostScript Output Option” to be “Optimize for Portability”
  - Select “TrueType Font Download Option” to be “Outline”
  - Select “Send PostScript Error Handler” to be “No”
  - Click “OK” three times, print your file.
  - Now, use Adobe Acrobat Distiller or ps2pdf to create a PDF file from the PS file. In Acrobat, check the option “Embed all fonts” if applicable.

If your file contains Type 3 fonts or non embedded TrueType fonts, we will ask you to fix it.

7.1 Margins in LaTeX

Most of the margin problems come from figures positioned by hand using `\special` or other commands. We suggest using the command `\includegraphics` from the graphicx package. Always specify the figure width as a multiple of the line width as in the example below using .eps graphics

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

or

```
\usepackage[pdftex]{graphicx} ...
\includegraphics[width=0.8\linewidth]{myfile.pdf}
```

for .pdf graphics. See section 4.4 in the graphics bundle documentation (`http://www.ctan.org/tex-archive/macros/latex/required/graphics/grfguide.ps`)

A number of width problems arise when LaTeX cannot properly hyphenate a line. Please give LaTeX hyphenation hints using the `\-` command.

Acknowledgments

Use unnumbered third level headings for the acknowledgments. All acknowledgments go at the end of the paper. Do not include acknowledgments in the anonymized submission, only in the final paper.

References

References follow the acknowledgments. Use unnumbered third level heading for the references. Any choice of citation style is acceptable as long as you are consistent. It is permissible to reduce the font size to 'small' (9-point) when listing the references. **Remember that this year you can use a ninth page as long as it contains *only* cited references.**

[1] Alexander, J.A. & Mozer, M.C. (1995) Template-based algorithms for connectionist rule extraction. In G. Tesauro, D. S. Touretzky and T.K. Leen (eds.), *Advances in Neural Information Processing Systems 7*, pp. 609-616. Cambridge, MA: MIT Press.

[2] Bower, J.M. & Beeman, D. (1995) *The Book of GENESIS: Exploring Realistic Neural Models with the GEneral NEural SImulation System.* New York: TELOS/Springer-Verlag.

[3] Hasselmo, M.E., Schnell, E. & Barkai, E. (1995) Dynamics of learning and recall at excitatory recurrent synapses and cholinergic modulation in rat hippocampal region CA3. *Journal of Neuroscience* **15**(7):5249-5262.