Machine Learning Approaches to Information Extraction from Text and Images in Biomedical Journal Articles

Robert F. Murphy
Departments of Biological Sciences, Biomedical Engineering and Machine Learning and

Goal of tutorial
- Introduce problem of automated interpretation of articles containing text and images
- Describe relevant methods, mostly in context of SLIF (Subcellular Location Image Finder) system
- Describe future directions for field

Ultimate Goal of the field
- Machine understanding of biological journal articles (text and image)
- Criteria for success: Turing test - have machine be able to answer questions about an article as well as a human scientist

Intermediate Goal
- Extract information from combination of text and any kind of image in biological journal article
- Criteria for success: Achieve high precision and recall for extracted assertions (compared to expert scientist)

Immediate Goal (SLIF)
- Extract information about subcellular location from captions and figures containing fluorescence microscope images in biological journal articles
- Criteria for success: Achieve high precision and recall for extracted assertions (compared to expert scientist)

State of art: Bio Journal Information Extraction
- A number of systems to index literature via extracted terms
- A few systems to index image content in literature
- A few systems for document classification
Practices in Biological Journal Articles

- Articles not monolithic: they can support more than one biological conclusion
- Different types of data often combined in one article and in one figure
- Assume knowledge of basic biology
- Captions should be understandable without reference to paper
- Materials often defined in separate section

Introduction to Protein Subcellular Location

Eukaryotic cells have many parts

Protein localization

- The sequence of each protein determines where it is localized in cells
- Subsequences ("motifs") within a protein’s sequence are responsible for targeting it to one (or more) locations (structures/organelles)

Open questions

- How many distinct locations can proteins be found in? What are they?
- How many distinct motifs direct proteins to those locations? What are they?

Proteomics

- The set of proteins expressed in a given cell type or tissue is called its proteome
- Proteomics projects
  - sequence
  - structure
  - activity
  - partners
  - location
Location information in protein databases: Traditional approach

- conduct experiments of various types
  - Cell fractionation
  - Electron microscopy
  - Fluorescence microscopy
- describe the results in unstructured text (first in journal articles and then in summaries in databases)
  - “Protein X is located primarily in protrusions from the early endosomal membrane but is also found in the plasma membrane”

Location information in protein databases: Ontology approach

- Systematic analysis and comparison of these descriptions were made difficult by both the unstructured nature of the text and the variation in terminology used from one laboratory to another
- To address this problem, a restricted vocabulary for cellular components was created by the Gene Ontology consortium

Restricted Vocabulary Approaches

- Databases such as SwissProt use manual curation to assign GO terms to proteins based on reading of relevant literature
- A major problem is consistency of application of terms

Use of GO terms

- Databases such as SwissProt use manual curation to assign GO terms to proteins based on reading of relevant literature
- A major problem is consistency of application of terms

Comparison of GO terms for two proteins

<table>
<thead>
<tr>
<th>GolgB1</th>
<th>GPP130</th>
</tr>
</thead>
<tbody>
<tr>
<td>Integral to membrane; Golgi membrane; Golgi stack;</td>
<td>Integral to membrane; Golgi cis-face; Golgi lumen; endocytotic transport vesicle</td>
</tr>
</tbody>
</table>

Source: SwissProt
Determining protein location

- The primary method used to determine the subcellular location of a protein is to “tag” it with a fluorescent probe and then image its distribution within cells using fluorescence microscopy (abbreviate resulting Fluorescence Microscope Image as FMI).

Tagging proteins for fluorescence microscopy

- Immunofluorescence
  - “primary” antibody against the target, “secondary” antibody against the “primary” and conjugated with a fluorescent probe
  - Fixed-cells only
- Gene/cDNA-tagging
  - merge DNA coding for a naturally fluorescent protein (or vital probe binding sequence) with coding sequence of a protein of interest
  - Live-cell possible

Tagging proteins for fluorescence microscopy

- GFP-tagging
  - Can create fusion between GFP and a cDNA, in which case all regulatory sequences that control expression of the corresponding protein is lost
  - Can create fusion between GFP and the genomic sequence of a gene, in which case regulatory sequences preserved
  - Example: CD-tagging

Major information to extract for FMI in article figures

- Sample
  - Cell or tissue type
  - Treatments (drug addition, fixation)
  - Probes (fluorophores, targets)
- Acquisition
  - Microscope type
  - Magnification
- Display
  - Color mapping
- Internal Annotations
  - Panel labels

Analysis of example paper

- Direct Observation of Rapid Internalization and Intracellular Transport of Stemby/Neutrophile-Form Cells
Example gray scale image

Note panel labels, arrows, text annotation, scale bars (and inference needed to infer which panels they apply to)

Note phase contrast image in figure with mostly fluorescence images

Note correspondence between panels defined in caption
Information Extraction from Image and Text in Journal Articles

Figure 1: Information extracted from text and images in journal articles. The figure shows a page from a journal article containing text and images. The text is discussing the process of extracting information from images and text. The figures include overlays and panels with unusual labeling. The text mentions the extraction of overlapping graphs and micrographs in one figure. The goal is to improve the understanding and interpretation of the scientific content by leveraging both textual and visual information.
Inputs for automated paper interpretation

Data Sources
- All journals published electronically
- Many biological journals are open access
  - Pubmed Central collects them in one place
  - Biomed Central collection contains a number of journals in same style
- Many others have delayed open access
- Some have initial open access
- Those without open access have subscription access

Paper Formats
- All(?) journals use Publishing XML
- All provide PDF version

Biological Databases
- Many biological database containing structure information, especially about gene and protein names, sequences, structures, interactions
Basics of Supervised Machine Learning: Feature Selection and Classification

Feature selection
- Having too many features can confuse a classifier
- Can use comparison of feature distributions between classes to choose a subset of features that gets rid of uninformative or redundant features

Feature Selection Methods
- Principal Components Analysis
- Non-Linear Principal Components Analysis
- Independent Components Analysis
- Information Gain
- Stepwise Discriminant Analysis
- Genetic Algorithms

Simple two class problem

k-Nearest Neighbor (kNN)
- In feature space, training examples are

We want to label ‘?’
**k-Nearest Neighbor (kNN)**
- Find k nearest neighbors and vote

**Feature #1** (e.g., 'area')

**Feature #2** (e.g., roundness)

**So we label it** +

for k=3, nearest neighbors are

**Decision trees**
- Again we want to label ‘?’

**Feature #1** (e.g., 'area')

**Feature #2** (e.g., roundness)

**Decision trees**
- so we build a decision tree:

![](slide1)

<table>
<thead>
<tr>
<th>Feature #2 (e.g., roundness)</th>
<th>Feature #1 (e.g., 'area')</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>50</td>
</tr>
</tbody>
</table>

**Decision trees**
- Goal: split address space in (almost) homogeneous regions

**Support vector machines**
- Again we want to label ‘?’

**Feature #2** (e.g., roundness)

**Feature #1** (e.g., 'area')
Support Vector Machines (SVMs)

- Use single linear separator??

Slide courtesy of Christos Faloutsos

Support Vector Machines (SVMs)

- Use single linear separator??

Slide courtesy of Christos Faloutsos

Support Vector Machines (SVMs)

- Use single linear separator??

Slide courtesy of Christos Faloutsos

Support Vector Machines (SVMs)

- we want to label ‘?’ - linear separator??
- A: the one with the widest corridor!

Slide courtesy of Christos Faloutsos
Support Vector Machines (SVMs)

- We want to label '?' - linear separator??
- A: the one with the widest corridor!

Support vectors

Cross-Validation

- If we train a classifier to minimize error on a set of data, have no ability to generalize error that will be seen on new dataset
- To calculate generalizable accuracy, we use n-fold cross-validation
- Divide images into n sets, train using n-1 of them and test on the remaining set
- Repeat until each set is used as test set and average results across all trials

Describing classifier errors

- For multi-class classifiers, typically report
  - Accuracy = # test images correctly classified / # test images
- For binary classifiers (positive or negative), define
  - TP = true positives, FP = false positives
  - TN = true negatives, FN = false negatives
  - Recall = TP / (TP + FN)
  - Precision = TP / (TP + FP)
  - F-measure = 2*Recall*Precision/(Recall + Precision)

Design Issues

System structure considerations

- Even immediate goal requires complex mixture of functions to process papers
- Some functions require outputs of other functions as inputs
- Inputs and outputs may change as system evolves
- Functions may be written in different languages
- System uses and creates large number of images

System structure considerations

- Incremental nature of project argues for flexible pipeline system
  - Good choices available (not when we started SLIF project)
- Large numbers of papers and processing times for images argue for ability to compute (or recompute) only some results
- Large numbers and sizes argue for storage of images on disk rather than inside database
- Desire for modules using heterogeneous languages argues for use of scripting language to manage system
Labeling and evaluation

- Hand label as many cases as possible for each step to enable machine learning for that step and evaluation of effectiveness of each step in pipeline

SLIF design

- Preprocessing job to take PXML or PDF files and convert to “standard” organization
- Pipeline to process each paper and store results on disk and in relational database
  - Use machine learning as much as possible
  - Web application to interface between user and database

SLIF Preprocessor

- Can handle small differences between input formats
- Spiders source directories
  - creating a directory for each paper it finds
  - remembering Pubmed ID for each paper
  - creating subdirectories for each figure it finds
    - extracting figure as JPEG image
    - extracting caption as plain text

SLIF Pipeline

- Master Controller script in Perl
- Inputs and outputs for each module defined in terms of files that they need or create
- Controller can be asked to make any target
- Order that modules are run defined by dependencies
- Processing of each paper independent so compute cluster can be used for collection
- Results stored in PostgreSQL database

SLIF Web Application

- Java Server Pages to define queries and display results
- Programmatic access support through modifiers on URL
  - SOAP interface written and being tested
Panel Splitting [image]
- Difficult task in general case
- SLIF focuses on images, so chose approach with high precision and recall for images
- Recursive detection of light areas between panels with trimming

Panel Splitting
- Find horizontal or vertical line through figure with lowest average intensity
- If lowest is above threshold, stop
- Cut figure into two pieces
- Trim horizontal or vertical lines from edges of pieces if those lines have average intensity close to white or black
- If piece too small, discard
- Recurse on resulting pieces

Semi-automated labeling tool
1. Initialize list of previously labeled results to empty; initialize panel splitter parameters
2. Run initial panel splitter on some figures; output is coordinates of each putative panel
3. Compare to list of previously labeled putative panels
4. If match, assign previous label (correct or incorrect)
5. If not, display figure/panel and get label
6. If desired, change algorithm/parameters and go to step 2
7. Run again on new set of figures and just save initial results as unbiased estimate of accuracy
Information Extraction from Image and Text in Journal Articles

Image pointer detection

- Parse caption using set of rules to identify potential image pointers
- Single letters followed by period or comma
- Single letters or short phrases followed or surrounded by parentheses or brackets

Figure 1. (A) Single confocal optical section of BY-2 cells expressing U2B0-GFP, double labeled with GFP (left panel) and antibody against coilin (right panel). Three nuclei are shown, and the bright GFP spots colocalize with bright foci of anti-coilin labeling. There is some labeling of the cytoplasm by anti-p80 coilin antibody. Single confocal optical section of BY-2 cells expressing U2B0-GFP, double labeled with GFP (left panel) and 4G3 antibody (right panel). Three nuclei are shown. Most coiled bodies are in the nucleoplasm, but occasionally are seen in the nucleolus. Bars, 5 mm.

Identifying Image Pointers: Learning vs Hand-coded Heuristics

<table>
<thead>
<tr>
<th>Method</th>
<th>Precis</th>
<th>Recall</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>HC-1</td>
<td>98.5</td>
<td>45.6</td>
<td>62.3</td>
</tr>
<tr>
<td>HC-2</td>
<td>74.5</td>
<td>90.0</td>
<td>84.6</td>
</tr>
<tr>
<td>ABWI</td>
<td>89.7</td>
<td>91.0</td>
<td>90.3</td>
</tr>
<tr>
<td>SABWI</td>
<td>85.9</td>
<td>92.2</td>
<td>89.0</td>
</tr>
<tr>
<td>+ NA</td>
<td>88.6</td>
<td>93.8</td>
<td>91.1</td>
</tr>
</tbody>
</table>

Panel Label Finding

- Finding annotations is not difficult
- Interpreting annotations (what letter is it?) is hard
- Complex backgrounds
- Partially occluded letters

Method:
- Find candidate regions (using position & size)
- Enhance, rescale, binarize
- Apply OCR to regions
- Match possible label patterns to labels from text

Label matching

- Labels from caption (sorted): A, C, D, E
- OCR candidate patterns, based on layout
  - ABW0 (column-major)
  - ABW0 (row-major)
- Closest match by dynamic programming
  - ABW0_F ~ ABW0_F

Evaluation

- OCR directly on panels
  - # panels: 427
  - # text regions: 380
- OCR on intensity-normalized text regions
  - # panels: 271
  - # text regions: 316

Correcting using best alignment with respect to Needleman-Wunsch edit distance, using model of common OCR errors to set weights
Information Extraction from Image and Text in Journal Articles

Limitations
- Only looks for labels within panels (misses labels next to panel)
- Can’t assign same label to set of panels
- Only recognizes single letter labels (does not recognize “control”)

Annotation removal [image]
- All candidate annotations (including panel labels) are removed (set to background)
- Future: could define filters to recognize non-alpha symbols (arrows)

Scale bar finding [image and text]
- In image, look for solid, horizontal black or white bars
- In text, look for strings of form “(B)b(ar)” followed by number followed by “m”
- Assume number is in µm (microns)
- Scale in microns per pixel is number divided by length of bar in pixels

SLIF Pipeline components
- Paper
- Figure
- Panels
- Image analysis
- SLIF database
- Protein source, cell types
- Subcellular pattern assignment
- Matched panels
- Label finding
- Entity extraction [see text]

Caption scoping [text]
- Goal is to try to determine which words in the caption refer to which parts of the figure

SLIF Pipeline components

Classify image pointers as citation-style or bullet-style.

Figure 1. (A) Single confocal optical section of BY-2 cells expressing U2B 0-GFP, double labeled with GFP (left panel) and autoantibody against p80 coilin (right panel). Three nuclei are shown. Bright GFP spots colocalize with bright foci of anti-coilin labeling. There is some labeling of the cytoplasm by anti-p80 coilin. (B) Single confocal optical section of BY-2 cells expressing U2B 0 -GFP, double labeled with GFP (left panel) and 4G3 antibody (right panel). Three nuclei are shown. Most coiled bodies are in the nucleoplasm, but some are also seen in the nucleus (arrows). All coiled bodies that contain U2B 0 also express the U2B 0-GFP fusion. Bars 5 µm. Movement of Coiled Bodies Vol. 10, July 1999 2299
Figure 1. (A) Single confocal optical section of BY-2 cells expressing U2B 0-GFP, double labeled with GFP (left panel) and autoantibody against p80 coilin (right panel). Three nuclei are shown, and the bright GFP spots colocalize with bright foci of anti-coilin labeling. There is some labeling of the cytoplasm by anti-p80 coilin. (B) Single confocal optical section of BY-2 cells expressing U2B 0-GFP, double labeled with GFP (left panel) and 4G3 antibody (right panel). Three nuclei are shown. Most coiled bodies are in the nucleoplasm, but occasionally are seen in the nucleolus (arrows). All coiled bodies that contain U2B 0 also express the U2B 0-GFP fusion. Bars, 5 mm.

**Style determines scope:**
- The scope of a bullet-style image pointer is all words between it and the next “bullet”
- The scope of a citation-style image pointer is some set of words nearby it (heuristically determined by separating words and punctuation)

**Protein Name Recognition**

Two potentially oncogenic cyclins, cyclin A and cyclin D1, share common properties of subunit configuration, tyrosine phosphorylation and physical association with the Rb protein.

**Named entity recognition (NER)**

- Need to match results of image analysis of panel contents with words describing the image
- Name of protein visualized, cell type used, etc.
- Very hard task because names of biological entities not used consistently
Use cases

- Possible query: “find all images of some protein involved in ribosome assembly that appears to be located in the cytoplasm”
  - “Proteins involved in ribosome assembly” determined by membership in a database (e.g., PIR,…)
- A high recall protein name extractor is preferred
- We care most about proteins from databases of all known proteins

Dictionary based algorithms for protein name recognition

Families of potentially oncogenic cyclins, cyclin A and cyclin D1…

Problems with dictionary based algorithms

- Words in a dictionary may not always be proteins, particularly after generalization to a pattern (e.g., “AT”, “fragment”, …)
  - Dictionaries must be first curated by removing such words
  - Constructing patterns requires engineering

Context based algorithms

- Context based algorithms learning algorithms
- Hidden Markov Models (HMMs) can be used to extract names from text

An HMM for protein name extraction
An HMM for protein name extraction

Two potentially oncogenic cyclins, *cyclin A* and *cyclin D1*
An HMM for protein name extraction

Two potentially oncogenic cyclins, cyclin A and cyclin D1

Discriminative versions of HMMS (CRFs, MEMMs/MaxEnt Taggers)

- New HMM-like methods:
  - Each token can have many features associated with it (isCapitalized, containsNumber, containsGreekLetter) as well as an “identity” (“alpha-3")
  - State is predicted with a linear weighting scheme that considers features and previous state

SemiCRFs

- Semi-markov version of CRFs
- Viterbi search replaced with search for best sequence of segments
- Distance to dictionary is feature of segments

Combining a dictionary with a hidden Markov model (Dictionary-HMM)

- Dictionary based algorithms can take advantage of existing resources, such as protein names in PIR database
- Context based algorithms do not in principle need updating
- Dictionary-HMM: learn how to do a soft match based on a small number of training data
Information Extraction from Image and Text in Journal Articles

Soft match to a path
With jumps and loops, path is like a profile-HMM

Dictionary-HMM
We need to specify:
- Structure: states and transitions
- Alphabet: set of emissions
- Initial Probability, Transition matrix, Emission matrix

Building the structure of the dictionary- HMM
- Strategies of introducing paths
  - Integrate the whole dictionary: huge structure will bring huge transition and emission matrix
  - Use heuristics to choose a small number of likely paths

Building the alphabet
- Emissions we have
  - Tokens from training data
  - Tokens from dictionary
  - Subsampling to avoid too many emitted words
  - Unknown token

Initial probability
- Initial probability
  - Learn from data
  - \( \pi(GE) = \pi_0 \)
  - \( \pi(S_{i,j}) = (1 - \pi_0) / N \)

Transition matrix A:
- Depends on a small number of parameters \( a, b, g \)

\[
P(S_{i,j} = G | S_{k,l}) = \frac{a^k b^l}{Z_i}
\]

\[
P(GE | S_{i,j}) = 1
\]

\[
P(GE | GE) = \gamma
\]

\[
P(S_{i,j} = GE | GE) = \frac{b^l (1 - \gamma)}{Z_i N}
\]

0 < \( a, b, \gamma \) < 1

\( Z_i, Z_j \) are for normalization, \( N \) is the number of paths
Emission matrix B

- $P(W_{ij} | S_{ij}) = \delta P(W_{ij} | Dict)$: $W_{ij}$ is a word only in the dictionary of protein names, except $S_{ij}$.
- $P(W_{ij} | S_{ij}) = \alpha P(W_{ij} | GE)$: $W_{ij}$ is a word only observed in GE.
- $P(a_{ij} | S_{ij}) = \frac{1 - \delta}{m}$: $a_{ij}$ is any token in $S_{ij}$.

Experiments

- Available datasets:
  - Univ. of Texas: 700 Medline abstracts
  - GENIA 3.04: 2000 Medline abstracts
  - Yapex: 200 Medline abstracts
- None of these is completely appropriate for us.
- Baseline methods:
  - CRFs, MaxEnt
- Features (for CRF, MaxEnt) and tokenization (for dictHMM)

Learning the parameters

- EM approach based on Baum-Welch
  - E-step: run B-W on the test data to learn $A$, $B$, then estimate the average parameters $\alpha, \beta, \gamma, \epsilon, \delta$ from $A$, $B$.
  - M-step: Use these estimated $\alpha, \beta, \gamma, \epsilon, \delta$ to recalculate $A$, $B$.

Performance of different algorithms on different datasets

<table>
<thead>
<tr>
<th></th>
<th>U. of Texas</th>
<th>GENIA</th>
<th>YAPEX</th>
</tr>
</thead>
<tbody>
<tr>
<td>Previously published methods</td>
<td>73.4 / 47.8 / 57.9</td>
<td>49.2 / 66.4 / 56.5</td>
<td>67.8 / 66.4 / 67.1</td>
</tr>
<tr>
<td>Bunescu’s dictionary-based method</td>
<td>62.3 / 45.9 / 52.8</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>MaxEnt</td>
<td>67.2 / 57.3 / 69.1</td>
<td>67.3 / 65.4 / 66.2</td>
<td>69.3 / 58.1 / 63.2</td>
</tr>
<tr>
<td>CRFs</td>
<td>83.5 / 66.1 / 73.8</td>
<td>75.0 / 67.6 / 71.1</td>
<td>76.0 / 59.5 / 66.7</td>
</tr>
<tr>
<td>SemiCRFs</td>
<td>83.1 / 66.8 / 73.9</td>
<td>74.8 / 68.3 / 72.3</td>
<td>76.1 / 58.9 / 66.1</td>
</tr>
<tr>
<td>dictHMM</td>
<td>46.0 / 69.2 / 55.2</td>
<td>44.8 / 70.1 / 54.7</td>
<td>42.4 / 64.1 / 51.0</td>
</tr>
</tbody>
</table>

Performance on U. of Texas dataset
Two strategies to improve the Dictionary-HMM (1)

- Boosting-like strategy:
  - Step 1. build a Dictionary-HMM on a test sentence. If no protein found, end.
  - Step 2. learn the dictionary-HMM and calculate the optimal state sequence. Find the single protein path with highest likelihood and report it.
  - Step 3. remove the protein found in step 2 from test sentence. Go to step 1 with the reduced test sentence.

Two strategies to improve the Dictionary-HMM (2)

- Dictionary-HMM with more states

Performance of improved Dict-HMMs

<table>
<thead>
<tr>
<th></th>
<th>U. of Texas</th>
<th>GENIA</th>
<th>YAPEX</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRFs</td>
<td>83.5 / 66.1 / 73.8</td>
<td>75.0 / 67.6 / 71.1</td>
<td>76.0 / 59.5 / 66.7</td>
</tr>
<tr>
<td>SemiCRFs</td>
<td>83.1 / 66.8 / 73.9</td>
<td>74.8 / 66.3 / 72.3</td>
<td>76.1 / 59.9 / 66.1</td>
</tr>
<tr>
<td>Dict-HMM</td>
<td>46.0 / 69.2 / 55.2</td>
<td>44.8 / 70.1 / 54.7</td>
<td>42.4 / 64.1 / 51.0</td>
</tr>
<tr>
<td>Dict-HMM + boosting-like method</td>
<td>49.8 / 74.3 / 59.6</td>
<td>48.3 / 73.9 / 58.5</td>
<td>45.1 / 69.7 / 54.8</td>
</tr>
<tr>
<td>Dict-HMM + additional states</td>
<td>51.8 / 72.3 / 60.4</td>
<td>51.3 / 72.4 / 60.1</td>
<td>45.1 / 65.7 / 53.5</td>
</tr>
</tbody>
</table>

Performance on words that match dictionary

- Many putative protein names by CRFs or semiCRFs are poor matches to dictionary entries
- Can measure similarity of a putative name to its closest match in dictionary using TFIDF (term frequency * inverse document frequency)
- Calculate as number of words in common divided by total number of words in both (weighted by frequency of words overall)
- Examine only putative protein names with TFIDF score greater than 0.9

Evaluation for protein names with TFIDF > 0.9

<table>
<thead>
<tr>
<th></th>
<th>U. of Texas</th>
<th>GENIA</th>
<th>YAPEX</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRFs</td>
<td>84.7 / 68.5 / 75.7</td>
<td>76.9 / 67.3 / 71.8</td>
<td>78.5 / 60.3 / 68.2</td>
</tr>
<tr>
<td>SemiCRFs</td>
<td>85.3 / 69.8 / 76.8</td>
<td>77.9 / 73.6 / 75.7</td>
<td>80.1 / 61.9 / 69.8</td>
</tr>
<tr>
<td>Dict-HMM</td>
<td>69.1 / 99.3 / 81.5</td>
<td>65.8 / 98.7 / 79.0</td>
<td>64.3 / 100 / 78.3</td>
</tr>
</tbody>
</table>
Conclusions

- SemiCRFs have higher precision, lower recall
- Dictionary-HMM has higher recall, lower precision
- Dictionary-HMMs have high recall for dictionary-like protein names

Panel typing [image and text]

- Goal is to identify the general type of each panel
- Possibilities are graph, cartoon, electron micrograph, light micrograph, fluorescence micrograph, gel picture

Observations/Assumptions

- Graphs and cartoons have very high contrast (black on white)
- Electron micrographs and light micrographs have gray background and little contrast
- Fluorescence micrographs and gel pictures have near black backgrounds and full range of gray levels

Initial approach (2001)

- Downloaded PDF files from Pubmed Central
- Extracted figures, split into panels
- Labeled 1586 panels as either FMI or non-FMI by viewing panel
- Made 64-bin histogram of gray levels for each panel

Initial approach

- Used 64 values as features to "train" k-nearest neighbor classifier for FMI vs. non-FMI
- Used labeled examples with leave-one-out cross validation to choose best k
- Calculate number of neighbors that are FMI
- Choose threshold on this number to trade precision vs. recall
Initial approach

- Best k was 9
  - Obtained recall of 70% and precision of 100% for high threshold
  - Obtained recall of 92% and precision of 97% for lower threshold
- Tested for another set of 100 panels
  - For k=11 and T=5, obtained recall of 90% and precision of 100%

Second approach

- For new collection of figures from PNAS, precision not as good (~50%)
- Especially observed gel pictures frequently being classified as FMI

Second approach

- Labeled 1993 panels (one panel each from 898 figures and all panels from 175 figures)
- Displayed both figure and caption during labeling to increase accuracy
- Initial labeling by one person, checked by another
  - 41% were FMI, 19% were gels

Second approach

- Calculated 64 histogram features
- Added 7 edge features measuring fraction of edge, homogeneity of edge direction and horizontal and vertical edge content
- Added “bag of words” text features
  - One feature for each word found in all of the training examples (20,767 words)
  - For each panel, words in the scope of that panel and words in the scope of the entire caption were counted

Performance with different feature sets

- SVM classifier with all features
- Previous kNN classifier retrained on new data
- Previously trained classifier

IASTED BIOMed/SPPRA 2007 - R.F. Murphy
Experiments | Recall | Precision | Error Rate
--- | --- | --- | ---
50% training | SVM | 0.829 | 0.836 | 0.132
Co-training | SVM | 0.826 | 0.828 | 0.137
10% training | SVM | 0.561 | 0.791 | 0.229
Co-training | SVM | 0.666 | 0.349 | 0.179

Conclusion is that representation of classes among labeled examples is good.
**Precision Recall Analysis**

![Precision Recall Analysis Graph](image)

- Prior updating with $\lambda = 0$ and $\lambda = 2$
- No updating (baseline single cell classifier)

**SLIF Pipeline components**

- **Caption understanding**
  - [Cohen et al., 2003]
- **Label finding**
  - [Kou et al., 2003]
- **Image analysis**
  - [see text]
- **Entity extraction**
  - [see text]
- **Panels**
- **Figure**
- **Paper**
- **Classifier**

**Pattern classification** [image]

- For each panel that has an identified scale bar, calculate subset of Subcellular Location Features that do not require segmentation into single cells

**Approaches to classify protein patterns**

- Fluorescence micrographs can contain subcellular region, single cell, or multiple cells/tissues

**Approaches to classify protein patterns**

- Features can be calculated at each level and aggregated to higher levels

- Analyzing patterns at single cell level requires segmenting multi-cell images
- Not easy in general case (algorithms usually customized to type of data available)
**Field-level classification**
- Alternative: assume entire field has same subcellular pattern (mostly true)
- Use features that
  - don’t require cell segmentation
  - are not sensitive to number of cells in field
  - can be calculated without reference to nucleus

**Object features**
- Single Object
- Field features

**Field features**
- Object features (object size, shape)
- Edge features
- Texture features

**Scale normalization**
- Images in figures have widely varying scales
- Use of features for classification requires scale to be the same
- Can use pixel size to rescale images to common size

**Thresholding**
- First type of feature is morphological
- Morphological features require some method for defining objects
- Most common approach is global thresholding
- Methods exist for automatically choosing a global threshold (e.g., Riddler-Calvard method)

**Ridler-Calvard Method**
- Find threshold that is equidistant from the average intensity of pixels below and above it
Ridler-Calvard Method

Otsu Method
- Find threshold to minimize the variances of the pixels below and above it

Adaptive Thresholding
- Various approaches available
- Basic principle is use automated methods over small regions and then interpolate to form a smooth surface

Suitability of Automated Thresholding for Classification
- For the task of subcellular pattern analysis, automated thresholding methods perform quite well in most cases, especially for patterns with well-separated objects
- They do not work well for images with very low signal-noise ratio
- Can tolerate poor behavior on a fraction of images for a given pattern while still achieving good classification accuracies

Object finding
- After choice of threshold, define objects as sets of touching pixels that are above threshold

2D Features
- Morphological Features

<table>
<thead>
<tr>
<th>SLF No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SLF1.1</td>
<td>The ratio of the largest to the smallest object to COF distance</td>
</tr>
<tr>
<td>SLF1.2</td>
<td>The variance of object distances from the COF</td>
</tr>
<tr>
<td>SLF1.3</td>
<td>The average object distance to the cellular center of fluorescence (COF)</td>
</tr>
<tr>
<td>SLF1.4</td>
<td>The ratio of the size of the largest object to the smallest</td>
</tr>
<tr>
<td>SLF1.5</td>
<td>The variance of the number of above-threshold pixels per object</td>
</tr>
<tr>
<td>SLF1.6</td>
<td>The average number of above-threshold pixels per object</td>
</tr>
<tr>
<td>SLF1.7</td>
<td>The Euler number of the image</td>
</tr>
<tr>
<td>SLF1.8</td>
<td>The number of fluorescent objects in the image</td>
</tr>
</tbody>
</table>
Suitability of Morphological Features for Classification

- Images for some subcellular patterns, such as those for cytoskeletal proteins, are not well-segmented by automated thresholding
- When combined with non-morphological features, classifiers can learn to “ignore” morphological features for those classes

2D Features

Object Skeleton Features

<table>
<thead>
<tr>
<th>SLF No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SLF7.80</td>
<td>The average length of the morphological skeleton of objects</td>
</tr>
<tr>
<td>SLF7.81</td>
<td>The ratio of object skeleton length to the area of the convex hull of the skeleton, averaged over all objects</td>
</tr>
<tr>
<td>SLF7.82</td>
<td>The fraction of object pixels contained within the skeleton</td>
</tr>
<tr>
<td>SLF7.83</td>
<td>The fraction of object fluorescence contained within the skeleton</td>
</tr>
<tr>
<td>SLF7.84</td>
<td>The ratio of the number of branch points in the skeleton to the length of skeleton</td>
</tr>
</tbody>
</table>

2D Features

Edge Features

<table>
<thead>
<tr>
<th>SLF No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SLF1.9</td>
<td>The fraction of the non-zero pixels that are along an edge</td>
</tr>
<tr>
<td>SLF1.10</td>
<td>Measure of edge gradient intensity homogeneity</td>
</tr>
<tr>
<td>SLF1.11</td>
<td>Measure of edge direction homogeneity 1</td>
</tr>
<tr>
<td>SLF1.12</td>
<td>Measure of edge direction homogeneity 2</td>
</tr>
<tr>
<td>SLF1.13</td>
<td>Measure of edge direction difference</td>
</tr>
</tbody>
</table>

2D Features

Haralick Texture Features (SLF7.66-7.78)

- Correlations of adjacent pixels in gray level images
- Start by calculating co-occurrence matrix P: N by N matrix, N=number of gray level. Element P(i,j) is the probability of a pixel with value i being adjacent to a pixel with value j
- Four directions in which a pixel can be adjacent
- Each direction considered separately and then features averaged across all directions

Co-occurrence Matrices

Example image with 4 gray levels

<table>
<thead>
<tr>
<th>4</th>
<th>2</th>
<th>2</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

Illustration – Skeleton

2D Features

Example image with 4 gray levels
Pixel Resolution and Gray Levels

- Texture features are influenced by the number of gray levels and pixel resolution of the image
- Optimization for each image dataset required
- Alternatively, features can be calculated for many resolutions

Summary

- Find entity names in text, and panel labels in text and the image.
- Match panels labels in text to panel labels on the image.
- Associate entity names to textual panel labels using scoping rules.
Information Extraction from Image and Text in Journal Articles

**Acknowledgments**

- **Students**
  - Dr. Michael Boland
  - Dr. Mia Markey (ugrad)
  - Gregory Poncea (ugrad)
  - Dr. Meel Velliste
  - Dr. Kai Huang
  - Dr. Xiang Chen
  - Ting Zhao
  - Shann-Ching Chen
  - Juchang Hu

- **Collaborators/Consultants**
  - UCSB: B. S. Manjunath, Ambuj Singh

**Main SLIF project team**

- Meel Velliste
- William Cohen
- Zhenzhen Kou
- Juchang Hu

**Acknowledgements - SLIF support**

- Grant 017396 from the Commonwealth of Pennsylvania Department of Health
- NIH grant K25 DA017357
- NIH grant R01 GM078622

**Acknowledgements - underlying pattern recognition**

- NSF grant EF-0331657
- NIH grant R01 GM068845
Information Extraction from Image and Text in Journal Articles

References

All available from http://murphylab.web.cmu.edu/publications

Review Articles


First published system for recognizing subcellular location patterns - 2D CHO (5 patterns)


2D HeLa pattern classification (10 major patterns)


3D HeLa pattern classification (11 major patterns)


Classification of multi-cell images

Subcellular Location Trees - 3D 3T3 CD-tagged images


SLIF - Subcellular Location Image Finder


SLIF - Subcellular Location Image Finder