Structured Output Prediction of Enzyme Function via Reaction Kernels

Juho Rousu

Josef Stefan Institute, Ljubljana, Slovenia
4 September 2009

Acknowledgements: Katja Astikainen, Liisa Holm, Esa Pitkänen, Sandor Szedmak, Craig Saunders, John Shawe-Taylor

Structured Output Prediction

- A family of machine learning methods aiming to predict complex objects "in one shot" rather than predicting their components individually
- Utilize the structure of the objects both to improve accuracy and to make learning and prediction efficient
- Aims to benefit from the Support Vector Machine research:
  - convex optimization in dual representation to make learning in high-dimensional feature spaces efficient,
  - margin-maximization to provide resistance to overfitting
- Example problem domains: sequence annotation, statistical machine translation, image segmentation, hierarchical classification, ...

Example: sequence-to-sequence learning

Statistical Machine Translation:

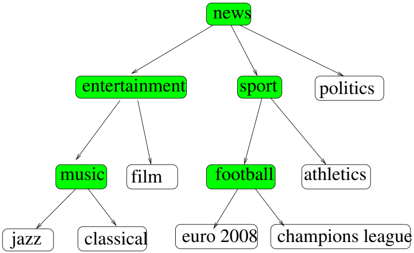
- Given a sentence in Finnish, output the English translation
- Both inputs and outputs may be represented as character, syllable, or word sequences

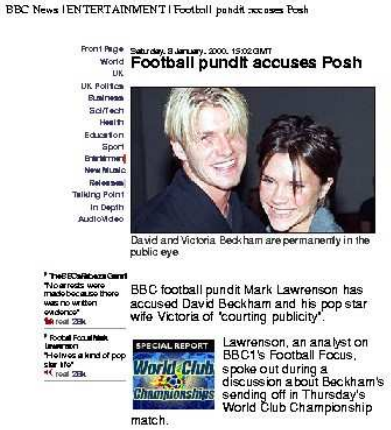
Sequence annotation:

- Input a DNA sequence
- Output the annotation (ORF, splice site, transcription factor binding)

Example: Hierarchical Multilabel Classification

Goal: Given document x, and hierachy $T = (V, E)$, predict multilabel consisting of a union of partial paths in T




BBC News | ENTERTAINMENT | Football pundit accuses Posh

Front Page
World
UK
UK Politics
Business
Sci/Tech
Health
Education
Sport
Entertainment
New Music
Releases
Talking Point
In Depth
AudioVideo

Saturday, 8 January, 2000, 15:02 GMT

Football pundit accuses Posh

David and Victoria Beckham are permanently in the public eye

The BBC's Rebecca Genn
"No arrests were made because there was no written evidence"
real 28k

Football Focus' Mark Lawrenson
"He lives a kind of pop star life"
real 28k

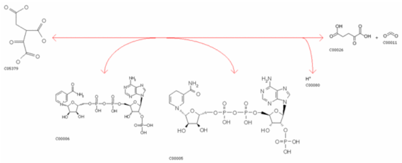
BBC football pundit Mark Lawrenson has accused David Beckham and his pop star wife Victoria of "courting publicity".

SPECIAL REPORT
World Club Championships

Lawrenson, an analyst on BBC1's Football Focus, spoke out during a discussion about Beckham's sending off in Thursday's World Club Championship match.

Predicting Enzyme Function

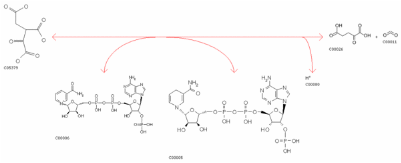
- Task: Given a enzyme sequence, predict the function i.e. biochemical reaction that is catalyzed
- Standard approaches:
  - Annotation transfer: BLAST an existing enzyme with similar sequence, predict the new enzyme has the same function
  - Classification: Learn a classifier to assign the sequence to some previously characterized function in a database



MSKKISGGSVVEMQGDEMTRIIWELIKEKLIFPYVELDLHSYDLGIENRD

Predicting Enzyme Function

- We wish to:
  1. improve on the accuracy on classifying into previously known classes under remote homology
  2. learn to build models that have a cabability to predict, new, previously uncharacterized, hypothetical functions
- The second goal is very ambitious, but reaching for it can give the first as a side effect :)



MSKKISGGSVVEMQGDEMTRIIWELIKEKLIFPYVELDLHSYDLGIENRD

Structured Output Prediction with Kernels

The general framework, used by most structured prediction approaches (M^3N, SVMstruct, ...)

- Inputs and outputs mapped to a joint feature space via a feature map $\varphi(x, y)$
- Margin-based learning of a linear compatibility score function with parameters w, b

$$F_w(x, y) = w^T \varphi(x, y) \ [+b]$$

- Optionally, use of *kernels* for inputs $\kappa(x, y; x', y') = \varphi(x, y)^T \varphi(x', y')$ making learning in high-dimensional spaces efficient.
- Prediction via *preimage* computation

$$\hat{y}(x) = \mathbf{argmax}_y F_w(x, y)$$

The “standard” optimization problem

Commonly used optimization task for learning model:

$$\begin{aligned} \min_{w,\xi \geq 0} \quad & \frac{1}{2} ||w||^2 + C \sum_{i=1}^{m} \xi_i \\ \text{s.t.} \quad & \xi_i \geq \mathbf{argmax}_y F_w(x_i, y) - F_w(x_i, y_i) + \ell(y_i, y) \; \forall x_i, y \, (1) \end{aligned}$$

- Intuitively: try to push the score of the reference pairs $F_w(x_i, y_i)$ higher than competing incorrect pairs $F_w(x_i, y)$ with a margin depended on the loss $\ell(y, y_i)$
- Algorithms typically solve the preimage problem (1) repetitively—this is the major efficiency bottleneck

Max-Margin Regression, MMR (Szedmak et al. 2005)

$$\begin{array}{ll} \min_{\mathbf{w},\mathbf{b},\xi} & \frac{1}{2}\left|\left|w\right|\right|^2 + C\sum_i \xi_i \\ \text{s.t.} & F_w(x_i, y_i) \geq 1 - \xi_i \\ & \boldsymbol{\xi} \geq \mathbf{0},\ \ i = 1, \ldots, m. \end{array}$$

- Intuitively, aims to separate the scores

  $$F_w(x_i, y_i) = \langle w, \varphi(x_i, y_i)\rangle$$

  from the origin with maximum margin (c.f. One-class SVM)
- Does not require solving the preimage during learning!
- Extremely efficient Augmented Lagrangian optimization, comparable to binary SVM learning
- Accuracy many times on par with M^3N and SVMstruct

MMR available from Sandor Szedmak's homepage: http://users.ecs.soton.ac.uk/ss03v/mmr.html

Towards simple(r) structured prediction

In learning with kernels we have two components that affect the learning results:

- The joint feature space induced by the kernels—decides the features that are available for learning
- The weight vector—corresponds to giving some features more importance than the others in the compatibility score $F_w(x, y)$

Question: can we circumvent learning the weight vector and only use the joint kernel density for prediction?

Structured prediction via Kernel density estimation (Astikainen, Szedmak, Rousu; unpublished)

- We define the score function as:

$$F(x, y) = \sum_i K(x_i, y_i; x, y)$$

- Prediction via the preimage computation:

$$y(x) = \mathbf{argmax}_y F(x, y)$$

- Essentially a Parzen window classifier in the joint feature space, assuming equal class weights

Tensor product feature map

Joint feature map as the tensor product $\varphi(x,y) = \psi(x) \otimes \psi(y)$ of feature maps for the inputs $\phi(x)$ and outputs $\psi(y)$.

- Contains all product features $\varphi(x,y)_{h,j} = \phi(x)_h \psi(y)_j$
- Assumes no prior alignment information of input and output structures
- Joint kernel given by the elementwise product

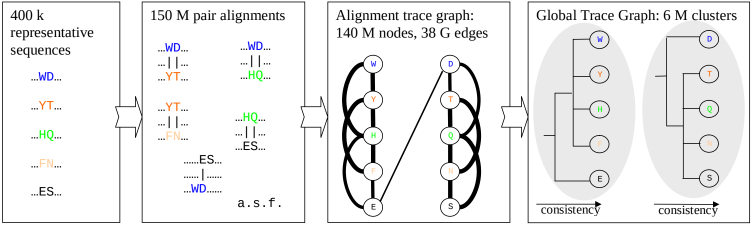
$$K_\varphi(x,y;x',y') = K_\phi(x,x') \cdot K_\psi(y,y')$$

Input feature maps 1: string kernels

- We considered the following sequence features and corresponding kernels $k(s, s') = \phi(s)^T \phi(s')$:
  - STRx: contiquous substrings of length x.
  - GAPxyz: subsequences of length x with y gaps of length z
- Different Combinations of the above:
  - Sum: $k(x, z) = \sum_q k_q(x, z)$
  - Polynomial kernel: $k_{poly}(x, z) = (k(x, z) + 1)^d$
- Unfortunately, none of these worked really well ...

Input features 2: Bag of conserved residues - GTG

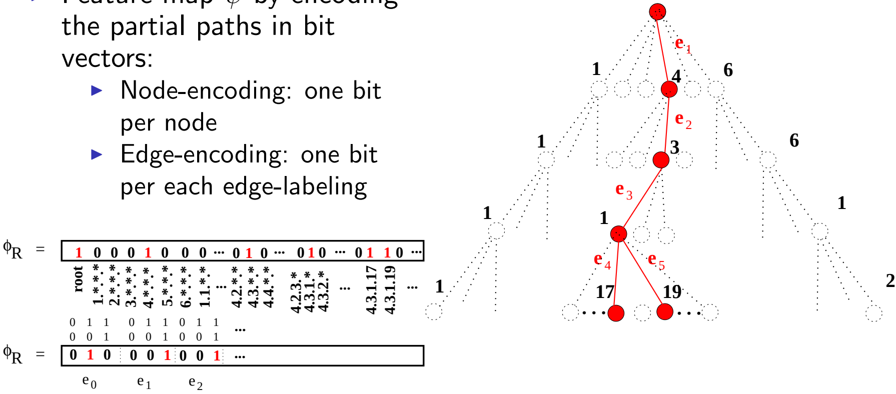
- We use predicted conserved residues given by Global Trace Graph, GTG (Heger et al. 2007) an all-against-all alignment graph of *all non-redundant protein sequences* (residues as nodes, edges from pairwise alignments)
- Conserved residues correspond to tight clusters in the alignment, some of these correlate with function



A. Heger, S. Mallick, C. Wilton, and L. Holm. Bioinformatics 2007 23: 2361-2367.

Output features 1: Embedding of a hierarchy

- Feature map ψ by encoding the partial paths in bit vectors:
  - Node-encoding: one bit per node
  - Edge-encoding: one bit per each edge-labeling



Classification hierarchies for protein function

EC (Enzyme Commission) hierarchy:

- Standard enzyme function taxonomy used by biologists
- Four levels (+ root), in total 1633 nodes.
- Classification by reaction mechanisms

Gold standard hierarchy:

- Brown, S., Gerlt, J., Seffernick J., Babbitt P. (2006). Genome Biology 7(1), 2006
- Two levels (+root): 5 superfamilies and 487 families
- Classification by evolutionary history

In addition many more: GO, MIPS, . . .

Output features 2: Reactant Matching kernel (Astikainen et al. 2009)

For reaction ρ we define feature map

$$\psi(\rho) = \sum_{M \in S_\rho} \phi(M) \otimes \sum_{M \in P_\rho} \phi(M),$$

- S_ρ and P_ρ are the set of substrates and products of ρ, respectively
- $\phi(M)$ is a feature map of molecule M (e.g. walk or subgraph spectrum)
- $\psi(\rho)$ contains pairs of substrate and product features (e.g. pairs of substrate and product subgraphs)
- The reactant matching kernel can be easily computed from an existing kernel between molecules.

Leveraging the kernel trick

- Here input and output kernels are fed through homogeneous polynomial kernel of degree d: $K_{poly}(z, z') = (K_{base}(z, z'))^d$
- The induced features are groups of GTG residues (input) and groups of molecule subgraphs (output)
- Degree of the polynomial kernel optimized independently for input and output

Experimental setup

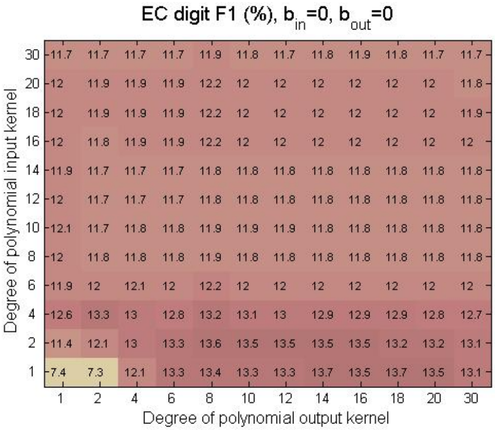
- 5-fold 'stratified' cross-validation: test fold EC-codes do not appear in the training folds
- Comparing: BLAST nearest neighbor, GTG nearest neighbor, MMR with EC hierarchy as the output,
- Prediction via 'trivial' preimage algorithm. We enumerate a fixed set of candidate functions $\mathcal{Y}_S$:

$$\hat{y}(x) = \mathbf{argmax}_{y \in \mathcal{Y}_S} F(x, y)$$

- In principle, the set $\mathcal{Y}_S$ could contain a much wider set of reactions, e.g. all reactions listed in KEGG. We only used the functions seen in the training and test sets.

Importance of the polynomial kernel

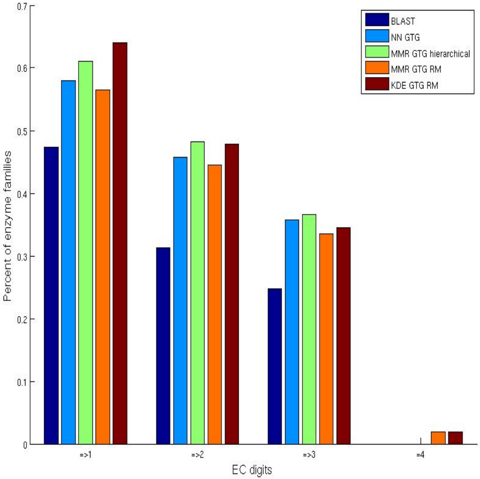
- Optimization of polynomial kernel degree for MMR on a tuning set
- Independent optimization of input and output kernels is beneficial
- Relatively high output kernel degrees give the best performance.
- The preimage problem is made correspondingly more difficult



	1	2	4	6	8	10	12	14	16	18	20	30
30	11.7	11.7	11.7	11.7	11.9	11.8	11.7	11.8	11.9	11.8	11.7	11.7
20	12	11.9	11.9	11.9	12.2	12	12	12	12	12	12	11.8
18	12	11.9	11.9	11.9	12.2	12	12	12	12	12	12	11.9
16	12	11.8	11.9	11.9	12.2	12	12	12	12	12	12	12
14	11.9	11.7	11.7	11.7	11.8	11.8	11.8	11.8	11.8	11.8	11.8	11.8
12	12	11.7	11.7	11.7	11.8	11.8	11.8	11.8	11.8	11.8	11.8	11.8
10	12.1	11.7	11.8	11.8	11.9	11.9	11.9	11.8	11.8	11.8	11.8	11.8
8	12	11.8	11.8	11.8	11.9	11.8	11.8	11.8	11.8	11.8	11.8	11.8
6	11.9	12	12.1	12	12.2	12	12	12	12	12	12	12
4	12.6	13.3	13	12.8	13.2	13.1	13	12.9	12.9	12.9	12.8	12.7
2	11.4	12.1	13	13.3	13.6	13.5	13.5	13.5	13.5	13.2	13.2	13.1
1	7.4	7.3	12.1	13.3	13.4	13.3	13.3	13.7	13.5	13.7	13.5	13.1

Predicting EC codes not seen in training

- Showing percentages of predictions with 1-4 first EC digits correct
- All methods based on GTG features outperform BLAST
- Both MMR and KDE with Reactant matching kernel can sometimes get the whole EC code correct
- KDE with RM kernel predicts 1, 2 and 4 first digits the best, hierarchical MMR predicts 3 first digits the best.



Conclusions

- Reaction kernels as means to improving accuracy in remote-homology enzyme function prediction
- Present results suggest:
  - Good kernels matter a lot: optimizing polynomial kernel gives big gains
  - How the hyperplane is learned (with KDE: not) less important for accuracy
- Structural learning made simple: With *MMR* and *KDE*, computing the kernels and the preimages efficiently are the only remaining bottlenecks for scalability to large data.