Poster H-56

ARTS: Accurate Recognition of Transcription Starts

Authors:
Sören Sonnenburg (*Fraunhofer FIRST*)
Alexander Zien (*MPI for biological Cybernetics and Friedrich Miescher Laboratory*)
Fabio de Bona (*Friedrich Miescher Laboratory*)
Gunnar Rätsch (*Friedrich Miescher Laboratory*)

Short Abstract: We present ARTS, a novel method for accurate recognition of transcription start sites in several species. Large scale training of a complex model is enabled by rapid kernel computations with suffix tries. In carefully designed experiments, we show that ARTS considerably outperforms recently published methods (for instance McPromotor and FirstEF).

Long Abstract:

Introduction

One of the most important features of genomic DNA are the protein-coding genes. While it is of great value to identify those genes and the encoded proteins, it is also crucial to understand how their transcription is regulated. To this end one has to identify the corresponding promoters and the contained transcription factor binding sites. TSS finders can be used to locate potential promoters. They may also be used in combination with other signal and content detectors to resolve entire gene structures. Neither massive mRNA sequencing nor comparative genomics will be able to solve these tasks completely in the near future.

Model and Methods

As most other TSS finders, ARTS combines several features, thereby utilizing prior knowledge about the structure of TSS's and their surroundings. We put particular emphasis on proximal features.

* The TSS is only determined up to a small number of base pairs. Additionally, nearby binding sites may also not be positionally fixed. In order to model the actual TSS site, we thus need to identify rather loosely localized sequence features. The recently proposed WD kernel (Weighted Degree kernel with shifts) allows for limited positional flexibility and is thus well suited for this task.
* Upstream of the TSS lies the promoter. It contains transcription factor binding sites, the ordering of which can differ quite drastically among promoters. This region is also often CpG-rich. Thus, we use the so-called spectrum kernel on a few hundred bps upstream of the TSS. The spectrum kernel is typically used to recognize regions in which certain k-mers are over- or under-represented.
* Downstream of the TSS follows the 5' UTR, and further downstream introns and coding regions. Since these sequences may significantly differ in oligo-nucleotide composition from intergenic or other regions, we use a second spectrum kernel for the downstream region.

* The 3D structure of the DNA near the TSS must allow the transcription factors to bind to the promoter region and the transcription to be started. To implement this insight, we apply two linear kernels to the sequence of twisting angles and stacking energies as estimated from dinucleotides.

The combined kernel used for TSS recognition is simply the sum of all sub-kernels, which is equivalent to appending the feature vectors in feature space.
Our model is complex in that it consists of several sophisticated kernels applied to rather long stretches of DNA. Furthermore, we have to train it on as many examples as possible (up to several hundred thousands), in order to attain a high prediction accuracy. Even with highly optimized general purpose SVM packages like LibSVM or SVM, training and tuning our model is intractable already for a tenth of the examples. The main reason is that the kernel computation, in particular of the WD kernel, is rather expensive, and that sufficient kernel caching is thwarted by the large number of support vectors. However, fast training is possible without kernel caching if the SVM output for any training point can be computed rapidly. For the kernels used in our TSS predictor, this can be implemented using suffix tries.

Experiments and Evaluation

For the human genome, we determine 8508 TSS positions based on the database DBTSS. To generate positive training data, we extract windows of size [-1000,+1000] around the TSS. It is rather unsafe to sample negative points randomly from the genome, since there are further yet unknown TSS hidden in it. Instead we extract negative points (again, windows of size [-1000,+1000]) from the interior of each gene: we draw 10 negatives at random from locations between 100bp downstream of the TSS and the end of the gene. We use 60% of these data for training a TSS classifier and 20% for validating it.
The validation accuracy is used as criterion for model selection: a total of 17 parameters (ranges and orders of the five kernels, and the SVM parameter “C”) have to be selected from respective finite sets. Since a full grid search is intractable, we carry out a local search heuristic based on axis-parallel searches. The chosen model yields 94% area under the ROC. Several computational investigations show that this good performance results largely from the WD kernel. We also show that the spectrum kernel mainly models the CpG island.
In a carefully designed experimental study, we compare our TSS finder on the human genome to state-of-the-art methods from the literature, namely McPromoter, Eponine, and FirstEF. A proper evaluation setting has to take into account that the TSS position is only determined up to a couple of bps (maybe around 20). We do so by looking at the genome at a smaller resolution, e.g. partitioned into chunks of length 50. Two neighboring chunks may both be considered positive, if a TSS is known to be close to their border. Chunks downstream of a TSS, but still inside the gene, served as known negatives. We derived our test set from genes new to version 5 of DBTSS, which were used neither for training ARTS nor for the other methods. For given false positive rates within a reasonable range, we consistently achieve considerably higher true positive rates. For instance, ARTS finds about 24% true positives at a false positive rate of 1/1000, where the other methods find less than half (10.5%).
We train ARTS for several other model organisms, including mouse and fruit fly. For all of them, we achieve similar accuracies, thus empowering an improved genome annotation. We provide our datasets as well as the predictions for free usage. Further, we even provide the library of machine learning tools which forms the basis of ARTS. More details are found at

http://www.fml.mpg.de/raetsch/projects/arts.