Predicting bacterial virulence factors – evaluation of machine learning and negative data strategies

Robert Rentzsch, Carlus Deneke, Andreas Nitsche and Bernhard Y. Renard

review by Dominik

Agenda

- Problem
- Data Selection
- Algorithm
- Results

Predicting VF

- Virulence factors (VF) are those components of an organism that determine its capacity to cause disease but are not required for its viability per se.
- To save experimental resources, the detection of known and the prediction of yet unknown bacterial VFs *in silico* have been the goals of different earlier studies (Virulence Searcher, VirulentPred, MP3, Virulent-GO).

Positive data selection

- PATRIC, 26.02.2016 -- 8662 entries
- We then followed a mapping protocol that compiles an appropriate set of background proteomes for the subsequent selection of negatives and, at the same time, automatically resolves redundancy in the positive data set arising from identical sequences with different identifiers. (UniProt ID mapping service)
- Intuitively, an appropriate sequence pool for deriving negatives is the entire set of VF-contributing proteomes. However, (...) we instead approximated the smallest set of proteomes covering all unique VF sequences (...).

Negative data selection

Specifically, there is no single obvious approach for identifying negative example sequences (‘negatives’ henceforth) when trying to predict VFs. Three existing strategies are

(i) random sampling from the VF-contributing proteomes,
(ii) semi-automatic selection from said proteomes based on (enzyme) annotations and manual curation
(iii) focussing on proteins deemed ‘essential’ for survival.

Their corresponding shortcomings are, briefly,

(i) the possibility of selecting yet undocumented or uncharacterized VFs,
(ii) labour-intensive curation that has to be repeated whenever the dataset or study target changes
(iii) the overlap between essential and VF proteins observed in reality, again necessitating curation.

We therefore explore a novel, more consistent and generalizable strategy for selecting negatives that relies on protein function annotation data.

Negative data selection -- NExIGO

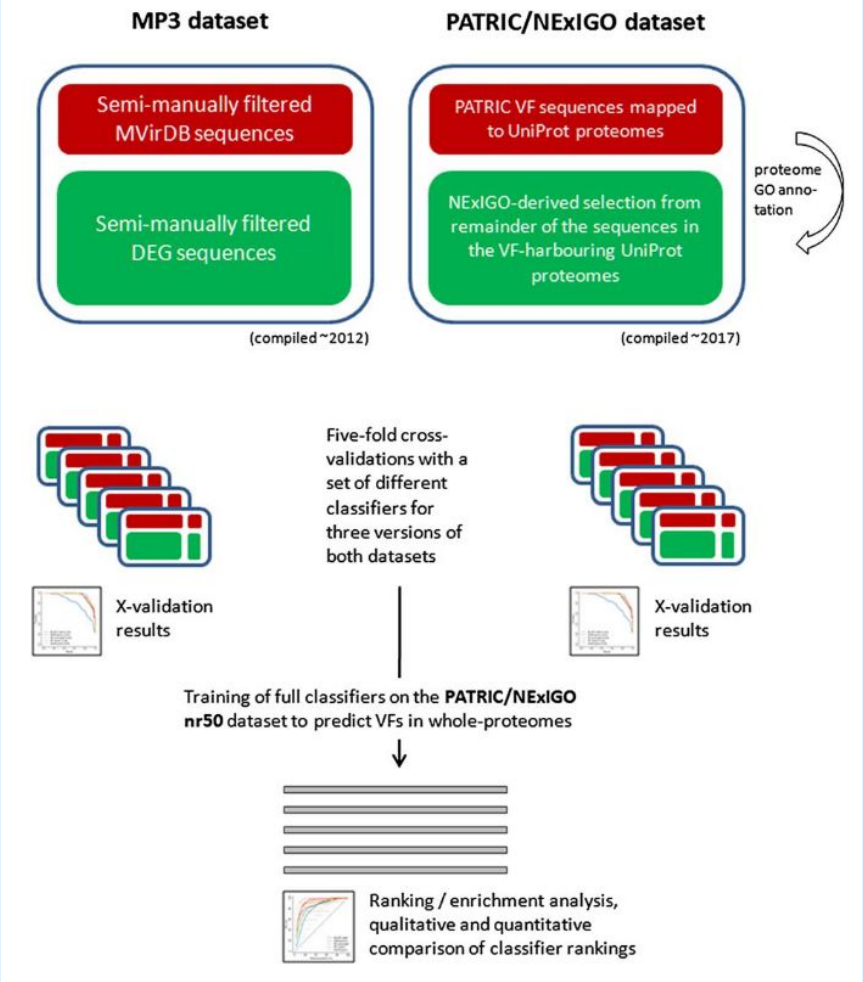
- All background proteomes were annotated automatically using the Argot2 server (They were assigned with Gene Ontology terms basing on sequence similarity and terms semantic similarity).
- Each GO term was assessed annotation frequency relative to the background of all VF-contributing proteomes.
- A hypergeometric distribution and Bonferroni correction is used to obtain a P-value for each term, with low values expressing statistical evidence for enrichment.
- Each protein is then assigned the lowest P-value observed among its annotated terms.
- Depending on the P-value distribution and the required number of sequences, a threshold is applied to obtain the final selection.

We call this method ‘Negative Example Identification using the GeneOntology’ (NExIGO)

Algorithm

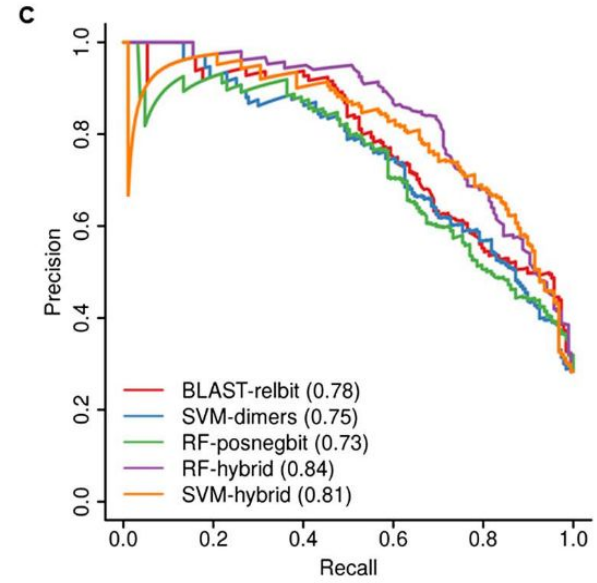
- To precisely reimplement the classifier type performing best in the MP3 paper (‘SVM-dimers’), we used SVM-Light v6.02 to train a polynomial-kernel SVM (‘-t1’) on 20×20 features corresponding to the dipeptide fractions (in percent) of a given sequence.
- The feature set was then extended by various seqsim-based features, all based on the best matches (during training: with non-matching identifier) found for a given query sequence in the positive and negative training databases,respectively.
- We further tested random forest (RF) classifiers (...). We chose the ranger v0.8.0 RF implementation, which is one of the fastest and provides a convenient R wrapper, to train probability forests, which return the fraction of votes for either class.
- The 5-fold CV benchmark from the MP3 paper was reimplemented (and extended).

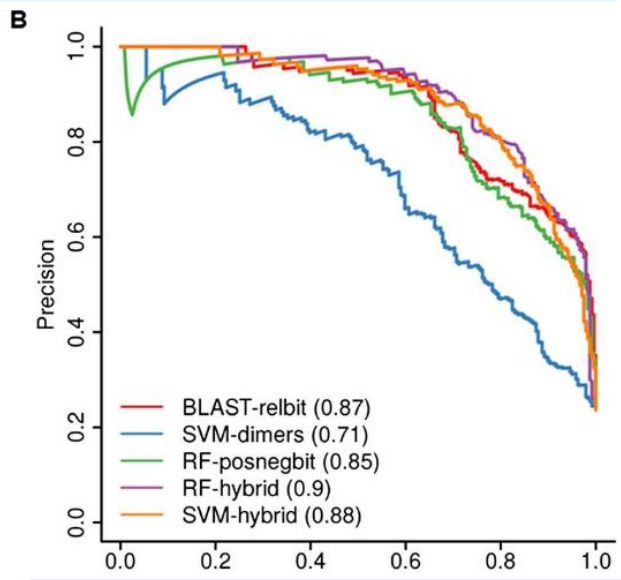
Study overview

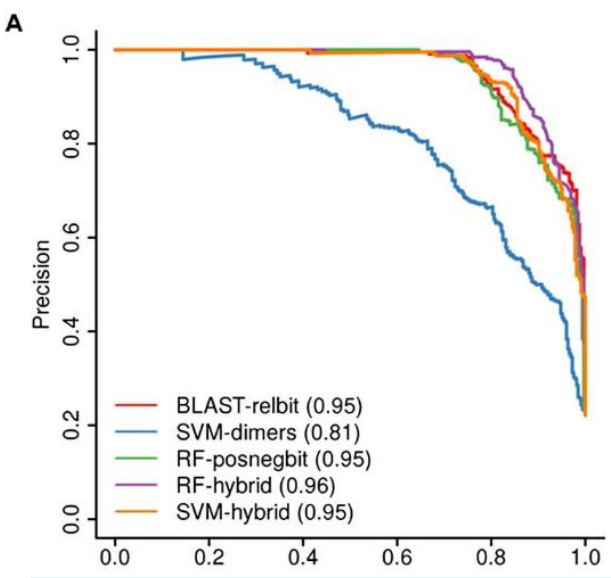


Results

for different levels
of CD-HIT redundancy
cut-off



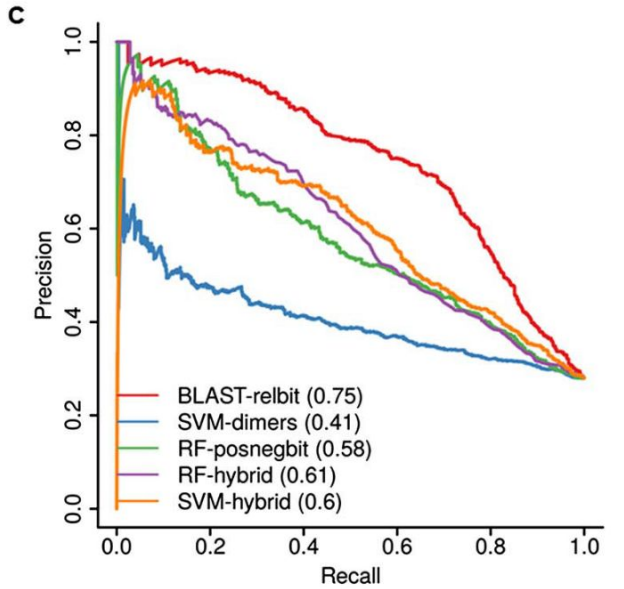


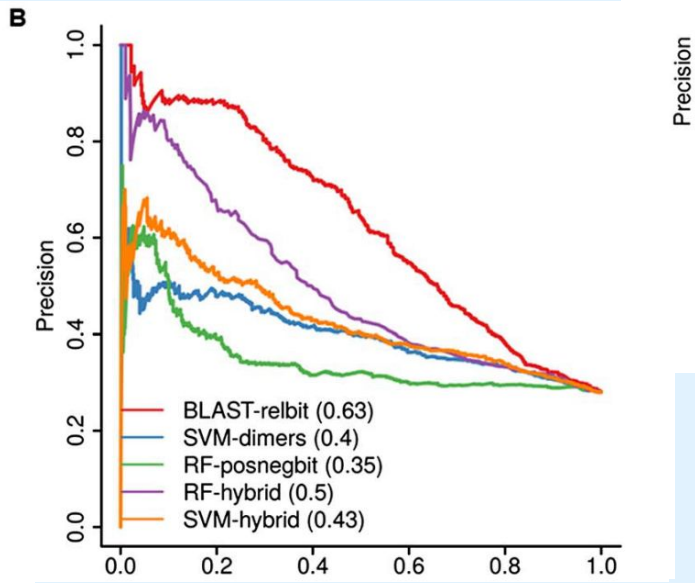


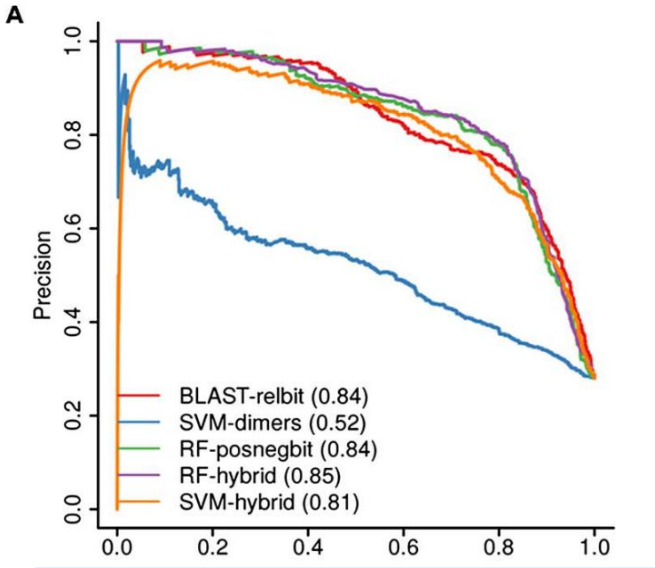
A) 90
B) 50
C) 30

Results of benchmarks

for different methods of sampling negative training data







A) NExIGO
B) random
C) DEG

Key points

- Bacterial virulence factor (VF) proteins are highly diverse in sequence and function, with the only obvious commonality of being involved in virulence mechanisms.
- The most commonly used generic tool for predicting VFs (and the only one from the last decade), MP3, use sa ML classifier that was not benchmarked against using simple (BLAST) sequence similarity.
- The performance of MP3's classifier turns out to be far worse than that of the similarity-based approach, on both its original data set and the novel, more diverse one presented here.
- The highest cross-validation performance is obtained when using 'hybrid' classifiers that combine both approaches; further, these should be trained using non-random negative data.