Bioinformatics Advance Access published August 8, 2016

Data and Text Mining

RNAcommender: genome-wide recommendation of RNA-protein interactions

Gianluca Corrado [1,*], Toma Tebaldi [2], Fabrizio Costa [3], Paolo Frasconi [4] and Andrea Passerini [1,*]

[1]Department of Information Engineering and Computer Science, University of Trento, Trento, 38123, Italy. [2]Centre for Integrative Biology, University of Trento, Trento, 38123, Italy. [3]Department of Computer Science, Albert-Ludwigs-Universitaet Freiburg, Freiburg, 79110, Germany. [4]Dipartimento di Ingegneria dell'Informazione, University of Florence, Florence, 50139, Italy.

*To whom correspondence should be addressed.

Associate Editor: Prof. Ivo Hofacker

Abstract

Motivation: Information about RNA-protein interactions is a vital prerequisite to tackle the dissection of RNA regulatory processes. Despite the recent advances of the experimental techniques, the currently available RNA interactome involves a small portion of the known RNA binding proteins. The importance of determining RNA-protein interactions, coupled with the scarcity of the available information, calls for *in silico* prediction of such interactions.

Results: We present RNAcommender, a recommender system capable of suggesting RNA targets to unexplored RNA binding proteins, by propagating the available interaction information taking into account the protein domain composition and the RNA predicted secondary structure. Our results show that RNAcommender is able to successfully suggest RNA interactors for RNA binding proteins using little or no interaction evidence. RNAcommender was tested on a large dataset of human RBP-RNA interactions, showing a good ranking performance (average AUC ROC of 0.75) and significant enrichment of correct recommendations for 75% of the tested RBPs. RNAcommender can be a valid tool to assist researchers in identifying potential interacting candidates for the majority of RBPs with uncharacterised binding preferences.

Availability and implementation: The software is freely available at http://rnacommender.disi.unitn.it

Contact: gianluca.corrado@unitn.it, andrea.passerini@unitn.it

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Post-transcriptional regulation of gene expression is mediated by interactions between transcripts and regulatory molecules, mainly RNA-binding proteins (RBPs) and non-coding RNAs (ncRNAs), creating ribonucleoprotein complexes (RNPs). RBPs are one of the most numerous protein classes, crucial in driving multiple complex mechanisms, from splicing to translation. Many proteins with previously unsuspected RNA-binding properties are still being discovered, increasing the range of RNA interactors (Beckmann, 2015). Each RBP binds to a population of transcripts with diverse affinity and specificity, recognizing RNA targets by elements of sequence or structure (Lunde, 2007). RBPs often exert versatile roles in multiple cellular compartments, working in context with other RBPs or regulatory molecules (Hogan, 2008; Corrado, 2014). Many of the rules underlying this complexity are still unknown: information about RNA-protein interactions is a vital prerequisite to tackle the dissection of RNA regulatory processes.

From an experimental point of view, this field witnessed great advances. Initially, RNA-protein interactions were determined with low-throughput experimental techniques, testing the interaction of a single RBP with a single transcript. The development of high-throughput techniques, such as cross-linking and immunoprecipitation (CLIP) coupled with deep sequencing (König, 2012; McHugh, 2014), allowed to identify, in a

single experiment, the genome-wide interactome for an RBP. Despite the potential of these genome-wide techniques and the increase in available data, the RNA interactome is currently available for less than 100 human proteins, representing less than 10% of the known population of human RBPs (1,542 manually curated human RBPs in Gerstberger (2014)). The reason for this lack of information is not only related to the cost and time of performing the procedure, but also to experimental problems such as unavailability of reliable antibodies, scarcity of material (these approaches are still far from being single-cell), or chemical properties of the interaction that complicate clipping.

The importance of determining RNA-protein interactions, coupled with the scarcity of the available information, calls for *in silico* techniques for predicting such interactions. Pancaldi and Bähler (2011) used support vector machines and random forests to predict RNA-protein interactions. Their method relies on hundreds of different biological features extracted from literature, that are not available for all proteins and transcripts. This limits the applicability of the method at a genome-wide scale. RPIseq (Muppirala, 2011) and Wang (2013) predicts RNA-protein interactions from sequence information only, using SVM and random forest classifiers on k-mer features in the former case and an extended Naive Bayes classifier considering correlation between features in the latter. CatRapid (Bellucci, 2011) uses the physicochemical properties of sequences to build interaction profiles to estimate the RNA-protein interaction propensity. These methods are trained on RNA-protein interactions obtained from 3D complexes available in PDB (Rose, 2015). Structural information is clearly more detailed than interaction maps obtained by sequencing approaches, but it is much harder to determine. Additionally, PDB complexes cover only individual interactions between fragments of proteins (usually one or two domains) and small fragments of RNA (with median length of 21 nucleotides in eukaryotic cells). In our setting, we are instead considering genome-wide interactions between proteins and transcripts. The more recent version CatRapid omics (Agostini, 2013) extends the prediction of the RNA-binding propensity at a genome-wide scale. In this perspective, it has a scope which is the most similar to the one we are targeting in this work. However, the computational limitations of the CatRapid omics web server prevent a genome-wide analysis of its performance[1].

In this work we present RNAcommender, a novel tool for genome-wide recommendation of protein targets. The main purpose of RNAcommender is to suggest candidate mRNA targets (transcripts) for unexplored RBPs, using interaction information available from high-throughput experiments performed on other proteins. RNAcommender basically works as a recommender system (Ricci, 2010), by propagating interaction information from known RBPs to novel ones. It takes as input a (incomplete) protein-mRNA interaction map and sequence information for both proteins and mRNAs, and attempts at completing the interaction map. For RBPs with few known targets (from low-throughput assays), this amounts at suggesting additional interactions. For completely novel RBPs (or even putative ones), it recommends the entire set of interactions from scratch. This de novo prediction task, known as *cold start recommendation* in recommender systems, is made possible by turning sequence information into appropriate features allowing to measure similarity between proteins (and between mRNAs) in terms of their binding capabilities. RNAcommender provides as output a ranking of candidate mRNA targets for each protein of interest.

We tested RNAcommender on a large dataset of human RBP-RNA interactions with high-throughput experimental evidence. For each test protein, we simulated both completion and de novo prediction by silencing most and all of the interaction information respectively. From its ability in successfully recovering the silenced interactions, RNAcommender appears to be a valid tool to identify potential targets for uncharacterised RNA-binding proteins.

[1] A comparison ran on a reduced setting, with 50 randomly sampled candidate targets per protein, proved favourable to RNAcommender, with an AUC ROC of 0.74 averaged over all RBPs as compared to an average AUC ROC of 0.60 achieved by CatRAPid omics. Similar results were obtained when comparing to the RPIseq web server (Muppirala, 2011), which obtained an average AUC ROC of 0.62. Details of the experimental comparison are reported in the supplementary material.

2 Materials and Methods

2.1 Dataset

The AURA 2 database (Aug. 5, 2015) (Dassi, 2014) includes a manually curated and comprehensive catalog of experimentally determined interactions between human RBPs and UTRs (untranslated regions in mRNAs). From this collection, we extracted data for all RBPs with high-throughput interaction evidence (in order to be able to validate RNAcommender predictions). This selection resulted in a set of 67 distinct RBPs interacting with 72,226 UTRs for a total of 502,178 interactions.

The available number of UTRs bound by an RBP ranges from 400 to 31,964, with a median of 4,503 and a mean of 7,495 (standard deviation: 7,711). The most selective RBPs interact with less than 1% of the possible targets, while the most general RBPs interact with more than 40% of the UTRs (the median is around 5-10%). The interaction information was encoded in an $n \times m$ matrix Y, where n and m are the number of RBPs and UTRs respectively: $Y_{ij} = 1$ if RBP i interacts with UTR j, and 0 otherwise.

2.2 RBP features

Features representing RBPs were built using domain information provided by Pfam (v. 28.0) (Finn, 2013), in order to capture similarities between protein structure, function and modularity at the same time (Lunde, 2007).

Each protein sequence was scanned against the HMM models of the Pfam-A v. 28.0 database, selecting all domains with e-value equal to or lower than 1.0. For each domain found in the RBP, the Fisher score of the matching protein subsequence was computed. The Fisher score is the derivative of the subsequence log-likelihood score with respect to each of the HMM model parameters (Jaakkola, 2000). Each protein was represented by the concatenation of the Fisher scores of its matching subsequences with respect to their correspondent Pfam models. When multiple subsequences of an RBP matched the same Pfam HMM model (i.e. they were identified as the same domain), their Fisher scores were averaged. When a Pfam domain was not detected in a protein a zero vector was used.

Formally speaking, let $\mathcal{T} : \{t_1, \ldots, t_M\}$ be the set of domain types modeled in Pfam (i.e. RRM_1, KH_1, ...), and $D : \{d_1, \ldots, d_N\}$ be the set of domains associated to a protein p (e.g. protein FUS has an RMM_1 in position 287-365 and a zf-RanBP in position 422-453). We define $\Theta : D \to \mathcal{T}$ as the function mapping domains of p to the domain types of Pfam. Let $D_{t_j} = \{d_i : \Theta(d_i) = t_j\}$ be the set of domains of type t_j in protein p. Let $\mathbf{s}_{d_i}$ be the fisher score of domain d_i with respect to the HMM model of $\Theta(d_i)$. In general if $\Theta(d_i) \neq \Theta(d_j)$ then $\mathbf{s}_{d_i} \in \mathbb{R}^a, \mathbf{s}_{d_j} \in \mathbb{R}^b$ with $a \neq b$. We computed the averaged Fisher score with respect to a domain type t_j as:

$$\mathbf{s}_{t_j} = \begin{cases} \frac{1}{|D_{t_j}|} \sum_{d_i \in D_{t_j}} \mathbf{s}_{d_i} & \text{if } |D_{t_j}| > 0 \\ \mathbf{0} & \text{otherwise} \end{cases} \quad (1)$$

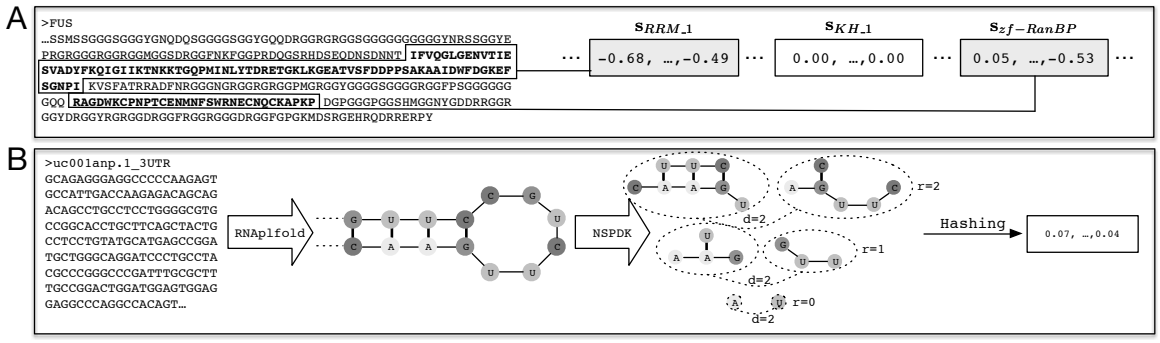

Fig. 1. (A) Each protein is represented by the concatenation of the Fisher scores (Jaakkola, 2000) of its domains with respect to their correspondent Pfam models. Missing Pfam domains are represented with a zero vector. (B) The RNA secondary structure is predicted using `RNAplfold` (Lorenz, 2011), then the feature representation is computed using the NSPDK approach that extends the notion of k-mers (with gaps) from the domain of strings to the domain of graphs.

We defined the Fisher score of a protein p as the concatenation of the Fisher scores with respect to all Pfam domains: $\mathbf{s}_p = [\mathbf{s}_{t_1}, \ldots, \mathbf{s}_{t_M}]$ (Figure 1A).

Finally, in order to control the dimensionality of the protein vectors, each RBP was represented in terms of its empirical kernel map, i.e. its similarity with respect to all other RBPs. The similarity between two proteins was computed as the normalised dot product between their Fisher score vector representations: $sim(p,q) = \langle \mathbf{s}_p, \mathbf{s}_q \rangle / \sqrt{||\mathbf{s}_p|| \cdot ||\mathbf{s}_q||}$.

2.3 RNA features

The self interacting structure of an RNA sequence is key to understanding post-transcriptional processes like protein binding. Unfortunately, there is still little experimental knowledge about the folding structure of full-length mRNAs, and although high-throughput protocols are now available (Sugimoto, 2015), one has still to rely on computational approaches to predict the structural properties of mRNA. In Lange (2012), different secondary structure prediction methods have been assessed yielding the conclusion that local folding can be more accurate than global approaches. They suggest a maximal span of 150 nucleotide to achieve a reasonable balance between maximizing the number of accurately predicted base pairs, while minimizing the effects of incorrect long range predictions. As recommended, we used `RNAplfold` (Lorenz, 2011) to estimate base pairs probabilities when interactions are constrained to lie within a user defined maximal span, which makes it suitable to scan long sequences. In order to consider only reliable predictions, we set the locality parameter to 150 nucleotides, we reduced the maximum span to 40 nucleotides and we set the average base pair probability cut-off to 0.4. Differently from sequence based approaches, here we built an explicit molecular graph using nucleotides as vertices and the predicted base pairs, together with the ribose-phosphate backbone, as edges (see Figure 1B). We then used the Neighborhood Subgraph Pair Decomposition Kernel (NSPDK) approach of Costa and De Grave (2010) to efficiently compute a sparse feature representation from the graph encoding. The NSPDK extends the notion of counting common k-*mers* (with gaps) from the domain of strings to the domain of graphs. Like it was done in Frasconi (2014), all distinct *neighborhood subgraphs* are given a unique numerical identifier using a fast hashing technique, obtaining in this way an explicit, although sparse, feature encoding[2]. Instead of considering short subsequences of length k (the k-mers), NSPDK considers small neighborhood graphs of maximal radius R, which are defined as the subgraphs induced by all the vertices within a given maximal distance R from a given node. To model the notion of 'gaps', i.e. the idea that two parts can match even if they differ in some positions, NSPDK considers pairs of neighborhood graphs at a maximal distance D as a single entity, in this way the matching operation can ignore all the nodes that are in an intermediate position between the two neighborhood graphs. As an example consider the feature marked as r=0, d=2 in Figure 1B, in this case the 'G' intermediate node is ignored and the feature can be matched to any pair of nodes with labels 'A' and 'U' that are at a relative distance of 2. The complete set of features is obtained considering all nodes in a graph as roots and all possible combinations of the values for the radius and the distance up to the user defined maximal values R and D.

Here we follow the recommendations of Heyne (2012) and set both maximal values to 2. The feature space dimensionality in NSPDK can be controlled adjusting the co-domain of the hashing function that maps graphs to integers. Note that a small dimensionality implies a higher efficiency in storage and subsequent processing, but also a higher risk of *collisions*, i.e. of assigning the same feature identifier to subgraphs that are not isomorphic, which leads to greater noise in the encoding. However, Li and König (2010) have shown theoretical robustness guarantees when considering codes obtained from the lowest bits of each hashed value. For this reason here we considered only the 10 lowest bits, effectively limiting the feature space dimensionality of the RNA structure encoding to 1024 (Figure 1B).

2.4 The model

Our model is inspired by the matrix factorization (MF) approach to collaborative filtering (Koren, 2009) where RBPs and RNAs play the roles of "users" and "items", respectively. If used in its basic form, MF would map both RBPs and RNAs to a latent feature space where a large correlation (dot product) between latent vectors predicts an interacting RBP-RNA pair. In the absence of side information, both RBPs and RNAs would be represented by their indicator vectors. Learning consists of determining two low-rank matrices P and R such that the RBP-RNA interaction matrix, Y, can be approximated as $Y \approx PR^\top$. While the basic MF approach has proved effective to build recommendation systems for movies (Koren, 2009), it is not directly suitable in our case for two main reasons. First, as explained in the introduction, the cold-start problem is severe for test proteins (none or just few RNA targets may be available). This setting calls for explicit feature vectors for RNAs and, most importantly, for RBPs in order to perform recommendations. Second, the number of RBPs is much smaller than the one of RNAs, which makes it difficult to directly project both in the same latent space.

[2] This is similar to the fingerprint technique used in chemoinformatics for small molecules.

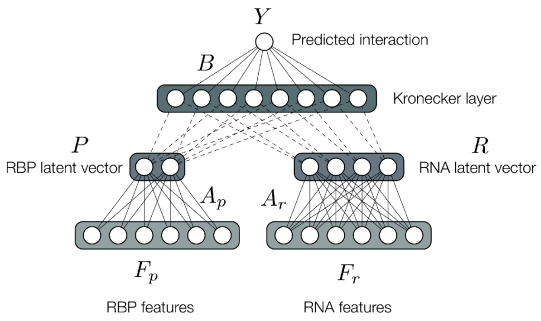

Fig. 2. Interpretation of the factorization model as a neural network.

The model used in this paper is based on tri-factorization, in a similar way as in Ding (2006), but without orthogonality constraints. A related form of tri-factorization has been also proposed for predicting multirelational dyadic data (Nickel, 2011). A major difference of our approach is the introduction of explicit features, mediated by latent projection matrices, and the use of non-linear mappings.

In our model, explicit features for RBPs and RNAs are computed as described in Section 2.2 and 2.3 respectively. These feature-based representations are (non-linearly) mapped into latent spaces of different sizes, where a third mapping associates them. The three factors (i.e. the parameters of the three mappings) are jointly tuned.

Formally speaking, let $F_p \in \mathbb{R}^{n \times l_p}$ and $F_r \in \mathbb{R}^{m \times l_r}$ be the explicit feature matrices associated with RBPs and RNAs, respectively. Let $A_p \in \mathbb{R}^{l_p \times k_p}$, $F_r \in \mathbb{R}^{l_r \times k_r}$, and $B \in \mathbb{R}^{k_p \times k_r}$ denote the three factors in the decomposition. The model is then defined by:

$$P = \sigma(F_p A_p) \in \mathbb{R}^{n \times k_p} \quad (2)$$

$$R = \sigma(F_r A_r) \in \mathbb{R}^{m \times k_r} \quad (3)$$

$$\hat{Y} = \sigma(PBR^\top) \in \mathbb{R}^{n \times m} \quad (4)$$

where σ is the logistic function. The model can also be interpreted as a feedforward neural network with a Kroneker layer (second-order units) as shown in Figure 2. One additional interpretation of the model is that RBP-RNA pairs are mapped into a non-linear feature space where the similarity (dot product) between two pairs is the product of the similarities between the corresponding RBP latent vectors and the RNA latent vectors. Interaction is then predicted by a linear classifier in this feature space. Preliminary results showed that deeper architectures, even with pretraining of the layers, increase the complexity and the training time of the model, without introducing significative improvements in the model performance. Additionally, worse performance was experienced when removing the Kronecker layer, showing the benefit of projecting proteins and RNAs into different latent spaces.

The factorization model is trained using stochastic gradient descent to optimise the regularised mean squared error:

$$\min_{A_p, A_r, B} \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} (Y_{ij} - \hat{Y}_{ij})^2}{n \cdot m} + \lambda \cdot r(A_p, A_r, B) \quad (5)$$

where $Y \in \mathbb{R}^{n \times m}$ is the interaction matrix between n proteins and m RNAs, and the regulariser $r(A_p, A_r, B)$ is the normalised Frobenius norm of the model weights:

$$r(A_p, A_r, B) = \frac{||A_p||_F}{l_p \cdot k_p} + \frac{||A_r||_F}{l_r \cdot k_r} + \frac{||B||_F}{k_p \cdot k_r} \quad (6)$$

The normalization has the role of canceling out the dependency on the sizes of the different matrices.

3 Results and Discussion

RNAcommender performance was tested on a collection of human protein-RNA interactions extracted from the AURA 2 database (see Section 2.1). In order to provide an estimate of the quality of predictions for protein target completion (for proteins with low-throughput experiments only) and full de novo recommendations (for proteins with no interaction information), we simulated both scenarios using the set of proteins with high-throughput experimental evidence. We performed a set of leave-one-protein-out experiments, each time using the full interaction information for $n - 1$ proteins, and hiding for the left-out one most of the interaction information available in the completion setting and all of it in the de novo one. We then evaluated the consistency of the provided recommendations with the hidden interactions (complete results are available as Supplementary data).

All experiments were run on a machine mounting 12 Intel® Xeon® CPUs E5-2603 v3 @ 1.60GHz, and 64GB of RAM, running Linux Ubuntu 14.04 LTS. The computation of the features for 67 proteins required around 30 minutes (single-threaded computation). The computation of the features for the 72,226 UTR sequences required 2.5 hours splitting the computation over the 12 CPUs. Training the model (see Section 2.4) required around 130-140 seconds per training epoch in multi-threaded computation over 12 CPUs, where a training epoch is defined as a complete pass over the training dataset (that contains around 4.8 million examples). Multi-threading scaled the computation time in an almost linear fashion.

3.1 Protein target completion

In this section, we considered the scenario of the prediction of RNA targets for an RBP with little RNA interaction information available. This situation frequently occurs when all known interactions for the RBP were determined by low-throughput experiments. In order to assess the performance of RNAcommender in this setting, we considered RBPs with high-throughput experiments, masking the majority of their known interactions. For each RBP, we masked (during training) all known interactions except for 15 RNA targets (this value corresponds to the average number of known interactions annotated in the AURA 2 database for proteins without high-throughput evidence). To improve the reliability of the results, we repeated the sampling procedure 5 times for each RBP (mean and standard deviation are reported for all the results presented in this section).

In principle, since the test RBP has few known interactions in the training set, it is possible to perform recommendation even without using explicit features for RBPs and RNA targets (see Sections 2.2 and 2.3). Nevertheless, our experiments indicate that the inclusion of features clearly improved the recommendation in terms of diversity and serendipity. Diversity expresses the heterogeneity level of the recommendations, i.e. how different are the recommended RNAs when considering different RBPs, while serendipity is a measure of how surprising the correct recommendations are (Shani and Gunawardana, 2011). In this work, we actualise the concept of serendipity on RBP target predictions. First, we introduce the measure of the *popularity* of an RNA j, which corresponds to the percentage of RBPs in the dataset binding to it: $pop_j = (\sum_{i=1}^{n} Y_{ij})/n$, where n is the number of RBPs and Y is the interaction matrix defined in Section 2.1. The concept of serendipity is inversely related to the one of popularity. An RNA that interacts with few (or none) of the RBPs in the dataset is more surprising when recommended than a common RNA that is known to bind all the proteins in the dataset. For this reason we define the *serendipity* of an RNA j as $ser_j = 1 - pop_j$.

In this section, the results for three different incremental feature usage scenarios are shown: no explicit features (ID.ID), explicit features only for the RNAs (ID.FE), and explicit features for both RBPs and RNAs (FE.FE).

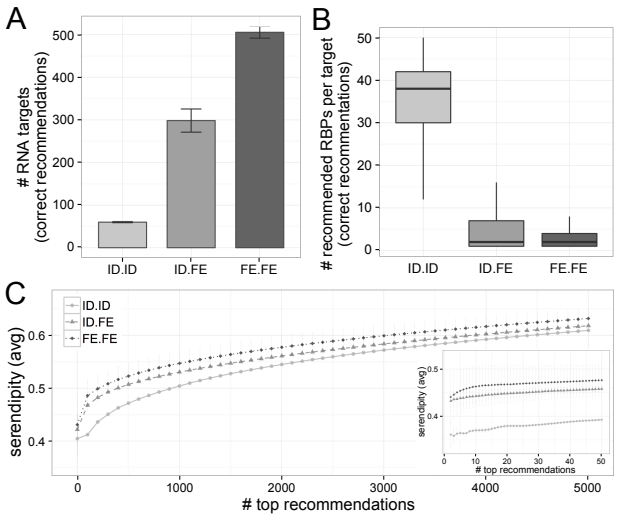

Fig. 3. (A) Number of different targets with a correct recommendation that are included in the top 50 target list of at least one protein. (B) Box plot of the number of recommended RBPs per RNA target. (C) Moving average of the serendipity of the RNA sequences along the rankings produced by the three feature settings.

In the absence of explicit features the proteins and the RNAs were represented by defining $F_p = \mathbb{I}_n$ and $F_r = \mathbb{I}_m$, respectively, where $\mathbb{I}_r$ stands for the r-th dimensional identity matrix.

All parameters for our model were selected with a 10-fold cross-validation procedure. We obtained $k_p = 5$ and $k_r = 50$ for the latent space sizes. The difference in the optimal sizes for the two spaces was expected because of the different cardinality of the sets of RBPs and RNAs in the dataset. For the stochastic gradient descent we set the learning rate $\eta = 1.0$. Regarding the parameter that controls the regularization of the weights of the model, different values were selected according to the feature usage: $\lambda = 10^{-2}$ for ID.ID, and $\lambda = 10^{-4}$ for ID.FE and FE.FE. In addition to regularisation, we used early stopping in order to avoid overfitting. We trained our model for 25 epochs for ID.ID, and 14 epochs for for ID.FE and FE.FE, where an epoch consists of a complete iteration over the entire training dataset. From these cross-validated parameters, we note that the introduction of explicit features diminishes the importance of the regularization of the model weights, and improves the convergence speed.

RNAcommender outputs a ranking (ranging from 0 to 1) of RNA targets for the test protein. As an overall measure of the quality of this ranking, we computed the Area Under the ROC curve (AUC ROC) for the three different feature settings. When averaged over the left-out proteins, we obtained very similar scores (0.76 for ID.ID and FE.FE, 0.77 for ID.FE; detailed results are available as the Supplementary material). However, the informativeness of these predictions in terms of coverage and serendipity is rather different, as will be detailed in the following.

It is common, when evaluating recommender systems, to focus the attention on the top recommendations because they are the ones that the user will most likely take into consideration. For each test protein and for each feature setting we considered the top 50 recommended targets. Figure 3A shows the number of different targets with a correct recommendation that are included in the top 50 target list of at least one protein. It is clear that the use of explicit features significantly increased the number of correctly recommended RNA targets. From 60 in the ID.ID case, to 298 in the ID.FE case and 506 in the FE.FE case. This increase in the overall number of different RNA targets in the top 50 recommendations implies a higher diversity and, indirectly, a higher serendipity because the targets for different proteins tend to be more diverse. Figure 3B represents box plot of the number of recommended RBPs per RNA target. Clearly the absence of explicit features (ID.ID) tended to produce less differentiated recommendations, since on average an RNA was recommended to 32 out of 67 proteins (with a median value of 38 RBPs). On the other hand the introduction of explicit features (ID.FE and FE.FE) produced very diverse recommendations, where an RNA was on average recommended to 6 proteins in the ID.FE case and to 3 proteins in the FE.FE case (for both cases the median value was 2).

One could wonder whether the previous analysis is influenced by the decision to focus on the first 50 predictions. Figure 3C shows the cumulative moving average of the serendipity of the recommended RNA targets along the top 5000 rankings produced by the three feature settings. Values were averaged over all samplings of all the 67 test RBPs. In all the cases, the serendipity increased along the rankings. However, the introduction of explicit features (ID.FE and FE.FE) improved the serendipity of the recommendation, by suggesting more specific targets for each RBP, especially when focusing on the top recommendations.

The same phenomenon of increased coverage and serendipity was observed when switching from single predicted targets to functional enrichments of sets of predicted targets. Gene Ontology enrichments became more specific and diverse as soon as explicit features were introduced in the model, while the ID.ID scenario provided the same set of repeated enrichments for each RBP analysed (Supplementary Figure S1).

3.2 De novo recommendation of protein targets

After testing our approach on the target completion task, we evaluated the ability of RNAcommender to suggest RNA targets for completely unexplored proteins. Performance were again computed in a leave-one-protein-out fashion, this time hiding all the interaction information for the left-out protein. In this setting, where no interaction-based propagation is possible, predictions need to be driven solely by feature similarity with training proteins. This implies that no recommendation is possible for proteins with null similarity with all other proteins in the dataset. This reduced the number of feasible leave-one-out experiments on our dataset from 67 to 49. In this setting we trained our model using the same parameters obtained through 10-fold cross-validation in the case in which both RBPs and RNAs were represented by explicit features (see Section 3.1).

Table 1 shows the results of the predictions for each leave-one-protein-out experiment. Each row reports the identity of the left-out RBP, the number of bound RNA targets (over the total of 72,226), the cumulative similarity (defined by the sum of the similarities with all other proteins in the dataset), the fraction of correct targets in the top 50 predictions and when considering a number of predictions equal to the number of true targets ($nTargets$) of the protein, and finally the AUC ROC computed over the entire set of candidate targets. Boldface numbers indicate a statistically significant enrichment in the number of correct targets in the top predictions with respect to an equally sized random sample, as computed by a Fisher test with $\alpha = 0.05$. Out of 49 leave-one-out experiments, the enrichment was statistically significant in 37 and 46 cases when considering precision at 50 and precision at $nTargets$ respectively (in most cases with a p-value many orders of magnitude smaller than the significance threshold).

Cold start recommendation is driven by similarity, among RBPs and among targets. For this reason, higher performance was associated with RBPs with higher values of cumulative similarity (Figure 4A) (p-value 0.024, by Wilcoxon Rank Sum test). Cumulative similarity with the proteins in the training set should be taken into account when predicting an unexplored RBP, because this factor influences the quality of the recommendation. In other words, a higher cumulative

Table 1. Evaluation of the recommendations of RNAcommender in a de novo setting. Test RBPs are sorted according to the precision at 50 (descending), and the number of targets (ascending). Boldface numbers indicate precisions which are significantly better than what would be obtained with an equally sized random sample according to a Fisher test ($\alpha = 0.05$).

RBP	nTargets	cumSim	Pre@50	Pre@nTargets	AUCROC
TAF15	4462	1.69	**1.00**	**0.49**	0.90
FXR2	10460	1.85	**1.00**	**0.60**	0.87
LIN28B	15063	0.33	**1.00**	**0.64**	0.86
HNRNPD	15786	1.10	**1.00**	**0.41**	0.61
FMR1_iso1	16923	2.04	**1.00**	**0.66**	0.86
FMR1_iso7	18228	2.04	**1.00**	**0.58**	0.77
TIA1	19453	1.40	**1.00**	**0.73**	0.89
TIAL1	25616	1.03	**1.00**	**0.76**	0.88
AGO1	31964	0.59	**0.98**	**0.72**	0.82
EWSR1	6214	1.62	**0.96**	**0.58**	0.91
MSI1	10801	1.02	**0.96**	**0.47**	0.80
LIN28A	12821	0.33	**0.96**	**0.64**	0.88
EIF4A3	21759	0.05	**0.96**	**0.46**	0.65
RBM47	18653	−0.12	**0.92**	**0.58**	0.79
HNRNPF	4503	1.34	**0.90**	**0.30**	0.79
FUS	7577	1.74	**0.86**	**0.53**	0.87
AGO2	20761	0.40	**0.86**	**0.69**	0.85
ELAVL1	25715	1.34	**0.86**	**0.58**	0.72
DDX21	9424	0.05	**0.84**	**0.32**	0.67
ZC3H7B	12439	0.20	**0.82**	**0.51**	0.82
PCBP2	3749	0.31	**0.72**	**0.28**	0.78
FXR1	3358	1.50	**0.70**	**0.49**	0.93
YTHDF1	6648	0.26	**0.70**	**0.37**	0.81
HNRNPC	4799	0.88	**0.62**	**0.38**	0.85
RBM10	9968	0.10	**0.62**	**0.18**	0.72
HNRNPH1	4858	1.36	**0.56**	**0.23**	0.72
RBPMS	4706	0.03	**0.44**	**0.36**	0.86
IGF2BP2	9265	1.00	**0.42**	**0.40**	0.81
IGF2BP3	11429	1.15	**0.38**	**0.39**	0.75
IGF2BP1	9389	1.15	**0.30**	**0.37**	0.79
HNRNPA1	632	0.85	**0.28**	**0.18**	0.77
RBFOX2	850	0.55	**0.28**	**0.15**	0.77
HNRNPA2B1	2201	1.34	**0.28**	**0.22**	0.82
PUM2	3581	0.95	**0.18**	**0.21**	0.76
CELF1	940	0.27	**0.14**	**0.06**	0.72
QKI	1008	0.09	**0.14**	**0.12**	0.82
TARDBP	1332	0.06	**0.14**	**0.14**	0.80
STAU1	3520	0.42	0.10	**0.08**	0.48
YTHDF2	2108	0.26	0.04	**0.19**	0.85
AGO4	400	0.48	0.02	**0.04**	0.83
TARBP2	460	0.32	0.02	**0.05**	0.75
PUM1	3788	0.95	0.02	**0.12**	0.53
EIF3B	421	0.15	0.00	0.01	0.60
EIF3G	597	0.76	0.00	0.00	0.55
DGCR8	1600	0.27	0.00	**0.06**	0.63
PABPC1	2322	−0.11	0.00	**0.01**	0.39
U2AF2	2202	0.11	0.00	**0.05**	0.52
ADAR1	2210	0.02	0.00	**0.08**	0.70
RC3H1	2950	0.20	0.00	0.04	0.43

similarity for the test protein results in a more reliable recommendation on average. Note however, that the system learns how to weight and combine similiarities with respect to training proteins in providing recommmendations. Indeed, a simple approach predicting for a test protein all targets of its nearest neighbour in the training set provides substantially lower results (Supplementary Table T1). The average AUC ROC of the nearest-neighbour predictor is 0.66, againts a value of 0.75.

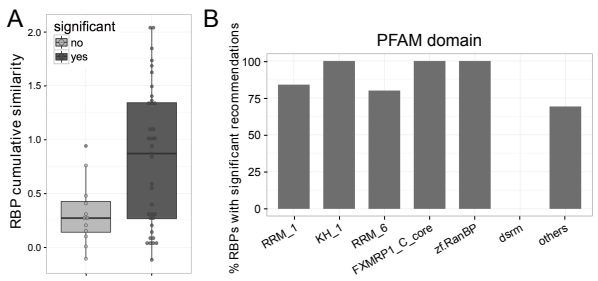

Fig. 4. (A) Box plot of the cumulative similarity of the RBPs grouped by significance. (B) Percentage of proteins with statistically significant predictions grouped by protein domain. The six most common domains are reported, plus “others” containing all remaining ones.

For the majority of the proteins RNAcommender outperforms the nearest-neighbour approach, with the exception of the strongly related proteins in dataset, e.g. IGFBP1, IGFBP2 and IGFBP3 or LIN28A and LIN28B.

We next classified the RBPs of the dataset according to their domain composition. We considered the six most frequent domains in the RBPs of the dataset, grouping all the left-out RBPs under the category named "others". As expected, the most frequent domains were RNA binding. Figure 4B shows the percentage of significant recommendations, grouping the test proteins according to the criteria just explained, and Supplementary Figure S2 reports the average AUC ROC values. For nearly all the most frequent domains, the percentage of test proteins with significant recommendations was above 75%. The only exception was represented by proteins with the dsrm domain (double stranded RNA binding motif): ADAR1, DGCR8, STAU1, and TARBP2. None of these proteins was significant in terms of top 50 recommendations. These RBPs were also characterised by fairly low values of cumulative similarity (Table 1).

When classifying RBPs according to their Gene Ontology annotation, we observed excellent performance for RBPs located in the "polysome" (CC), acting as "negative regulators of translation" or involved in "mRNA transport" (BP) and, with "mRNA binding" function (MF) (see Supplementary Figure S3-5). Worse performance was associated with translation initiation factors (EIF3B, EIF3G, PABPC1) and, again, double stranded RNA binding proteins. Considering the modularity of molecular complexes operating in post-transcriptional regulation, recommendations are expected to be difficult for RBPs whose RNA interaction is highly mediated by other protein components. For example, EIF3B and EIF3G, for which no correct target was identified in the first 50 recommendations (see Table 1), are both components of eIF3, the largest eukaryotic initiation factor, which is made up of 13 subunits (des Georges, 2015); the majority of these components do not directly interact with mRNA or participate in the selection of the bound target. The same consideration can be applied to double stranded RNA binding proteins, possibly providing a further explanation for their low recommendation performance.

Until now, we focused the attention on the top recommendations because they are reasonably the most relevant for the researcher. In order to measure the overall quality of the ranking imposed by our recommendations, Table 1 also reports the value of the AUC ROC computed over the entire set of candidate RNA targets. High values of AUC ROC are often correlated with high significance of the Fisher test (e.g. TAF15, EWSR1), and AUC ROC values close to 0.5 are always correlated with a lack of significance. However, for some of the proteins where the Fisher test was not significant, the value of the AUC ROC was substantially higher than the one of a random ranking (e.g. AGO4, TARBP2). Even if a reasonably good ranking function is learned, when the number of true targets is very small it can be hard to rank them in the

first 50 predictions. In fact, we noticed a significant fraction of correct targets in the precision at $nTargets$ for both AGO4 and TARBP2.

Finally, we compared the quality of de novo recommendations with the one of target completion as reported in the previous section. We analysed the feature setting FE.FE reported in Section 3.1 and the case presented in this section. Note that the only difference between these two settings is the number of available interactions for the left-out protein (15 for the first case, 0 for the other). We compared the recommendation performance in terms of precision at 50 and AUC ROC. Since in Section 3.1 we repeated the sampling of the 15 positive interactions 5 times for each test protein, we aggregated the performance measures by taking the median value. The average precision at 50 was 0.51 in the case of target completion and 0.53 in the de novo one. An even smaller difference was registered for the mean AUC ROC value, i.e. 0.761 against 0.754. Both performance measures are strongly correlated between the two settings (Spearman's rank correlation of 0.97 and 0.98 respectively). A Wilcoxon signed-rank test confirmed that differences are not statistically significant, with p-values of 0.98 and 0.09 for precision and AUC ROC respectively. These results showed that training a model including interaction information from low-throughput experimental techniques did not improve the recommendation performance in a significant manner. We did not test the performance for an increased number of interactions in the training set, since only the scenarios with no (novel proteins), few (low-throughput experiments) or many (high-throughput experiments) known interactions are meaningful in the RNA-protein interaction prediction problem. Note that, as shown in Maticzka (2014), even in the case of high-throughput data availability, the development of predictive *in-silico* models is of interest as these models can better compensate for experimental noise, such as false negatives due to tissue dependent expression or false positives due to accidental cross-linking effects.

4 Conclusion

In this work we proposed RNAcommender, a tool for RNA-protein interaction recommendation. By representing RNAs and proteins with features extracted from RNA secondary structure and protein domains and combining them with existing interaction information, we enabled the recommendation of targets for RBPs with little or no experimental evidence of interaction. We validated RNAcommender on a large dataset of human RBP-RNA interactions, showing an overall good ranking performance (average AUC ROC of 0.75) and a significant enrichment in correct targets in the top 50 predictions for 75% of the tested RBPs. Considering the successful results obtained by RNAcommender we also recommended RNA targets to 25 RBPs with low-throughput evidence (from Dassi (2014)) and 18 completely unknown proteins (from Gerstberger (2014)). The recommendations are available as supplementary data. Surely, the complexity of RNA regulation requires further efforts to optimise the predictions, but our tool can be a valid companion to assist experimental research, especially for the majority of RBPs whose interactors have not yet been experimentally identified.

In future work, we will address the integration of the interactions suggested by RNAcommender with other tools to provide a more robust and comprehensive analytical pipeline; in particular we plan to localise the interaction sites within the RNAs leveraging GraphProt models (Maticzka, 2014), and to integrate interaction predictions within the PTRcombiner system (Corrado, 2014) in order to improve the identification of groups of RBPs binding similar sets of targets.

Acknowledgements

The authors want to thank Björn Grüning for the useful help in the creation of the Galaxy tool of RNAcommender.

Funding

FC is funded by the Federal Ministry of Education and Research (BMBF grant 031 6165A e:Bio RNAsys) and by the German Research Foundation (DFG grant BA 2168/3-3).

Conflict of Interest: none declared.

References

Agostini, F. *et al.* (2013). catRAPID omics: a web server for large-scale prediction of protein–RNA interactions. *Bioinformatics*, **29**(22), 2928–2930.

Beckmann, B. *et al.* (2015). The RNA-binding proteomes from yeast to man harbour conserved enigmRBPs. *Nature communications*, **6**.

Bellucci, M. *et al.* (2011). Predicting protein associations with long noncoding RNAs. *Nature Methods*, **8**(6), 444–445.

Corrado, G. *et al.* (2014). PTRcombiner: mining combinatorial regulation of gene expression from post-transcriptional interaction maps. *BMC genomics*, **15**(1), 304.

Costa, F. and De Grave, K. (2010). Fast neighborhood subgraph pairwise distance kernel. In *Proc. of ICML*, pages 255–262.

Dassi, E. *et al.* (2014). AURA 2: empowering discovery of post-transcriptional networks. *Translation*, **2**(1), e27738.

des Georges, A. *et al.* (2015). Structure of mammalian eIF3 in the context of the 43S preinitiation complex. *Nature*.

Ding, C. *et al.* (2006). Orthogonal nonnegative matrix t-factorizations for clustering. In *Proc. of ACM SIGKDD*, pages 126–135. ACM.

Jaakkola, T. *et al.* (2000). A discriminative framework for detecting remote protein homologies. *J. Comp. Biol.*, **7**(1–2), 95–114.

Finn, R. D. *et al.* (2013). Pfam: the protein families database. *Nucleic acids research*, page gkt1223.

Frasconi, P. *et al.* (2014). klog: A language for logical and relational learning with kernels. *Artif. Intell.*, **217**, 117–143.

Gerstberger, S. *et al.* (2014). A census of human RNA-binding proteins. *Nature Reviews Genetics*.

Haussler, D. (1999). Convolution kernels on discrete structures. Technical Report 99-10, UCSC-CRL.

Heyne, S. *et al.* (2012). GraphClust: alignment-free structural clustering of local RNA secondary structures. *Bioinformatics*, **28**(12), i224–32.

Hogan, D. *et al.* (2008). Diverse RNA-binding proteins interact with functionally related sets of RNAs, suggesting an extensive regulatory system. *PLoS Biol*, **6**(10), e255.

König, J. *et al.* (2012). Protein–RNA interactions: new genomic technologies and perspectives. *Nature Review Genetics*, **13**(2), 77–83.

Koren, Y. *et al.* (2009). Matrix factorization techniques for recommender systems. *Computer*, (8), 30–37.

Lange, S. *et al.* (2012). Global or local? Predicting secondary structure and accessibility in mRNAs. *Nucleic Acids Res.*, **40**(12), 5215–5226.

Li, P. and König, C. (2010). b-bit minwise hashing. *Proceedings of the 19th International Conference on World Wide Web*, pages 671–680.

Lorenz, R. *et al.* (2011). ViennaRNA package 2.0. *Algorithms Mol Biol.*, **6**(1), 26.

Lunde, B. M. *et al.* (2007). RNA-binding proteins: modular design for efficient function. *Nature reviews Molecular cell biology*, **8**(6), 479–490.

Maticzka, D. *et al.* (2014). GraphProt: modeling binding preferences of rna-binding proteins. *Genome Biol*, **15**(1), R17.

McHugh, C. A. *et al.* (2014). Methods for comprehensive experimental identification of rna–protein interactions. *Genome Biol*, **15**, 203.

Muppirala, U. K. *et al.* (2011). Predicting RNA-protein interactions using only sequence information. *BMC bioinformatics*, **12**(1), 489.

Nickel, M. *et al.* (2011). A three-way model for collective learning on multi-relational data. In *Proc. of ICML*, pages 809–816.

Pancaldi, V. and Bähler, J. (2011). In silico characterization and prediction of global protein–mRNA interactions in yeast. *Nucleic acids research*, **39**(14), 5826–5836.

Ricci, F. *et al.* (2010). *Recommender Systems Handbook*. Springer-Verlag New York.

Rose, P. W. *et al.* (2015). The RCSB protein data bank: views of structural biology for basic and applied research and education. *Nucleic acids research*, **43**(D1), D345–D356.

Shani, G. and Gunawardana, A. (2011). Evaluating recommendation systems. In *Recommender systems handbook*, pages 257–297. Springer.

Sugimoto, Y. *et al.* (2015). hiCLIP reveals the in vivo atlas of mRNA secondary structures recognized by Staufen 1. *Nature*, **519**(7544), 491–494.

Wang, Y. *et al.* (2013). De novo prediction of RNA–protein interactions from sequence information. *Molecular BioSystems*, **9**(1), 133–142.