TarMiner: automatic extraction of miRNA targets from literature

Rodothea-Myrsini Tsoupidi
IMIS, 'Athena' RC, Greece
romi@imis.athena-innovation.gr

Ilias Kanellos
IMIS, 'Athena' RC & ECE, NTUA, Greece
kanellos@dblab.ntua.gr

Thanasis Vergoulis
IMIS, 'Athena' RC, Greece
vergoulis@imis.athena-innovation.gr

Ioannis S. Vlachos
Univ. of Thessaly & UoA, Greece
ivlachos@lessr.eu

Artemis G. Hatzigeorgiou
Univ. of Thessaly, Greece
arhatzig@uth.gr

Theodore Dalamagas
IMIS, 'Athena' RC, Greece
dalamag@imis.athena-innovation.gr

ABSTRACT

MicroRNAs (miRNAs) are small RNA molecules that target particular genes and prohibit their expression. Since many important diseases are related to the expression or non-expression of particular genes, knowing the miRNAs that affect these genes can help in finding possible treatments. In the last decade, a large amount of experimental studies trying to reveal the targets of several miRNAs has been published. A handful of curated databases that collect miRNA targets from the literature have been developed to make this information more easily available. However, due to the large number of existing published articles, maintaining these databases up-to-date is a tedious task that requires important resources. In this work we introduce TarMiner, a pipeline for automatic extraction of miRNA targets that can facilitate the curation process of databases that maintain miRNA validated targets.

Keywords

bioinformatics, NLP, text mining

1. INTRODUCTION

MicroRNAs (*miRNAs*) are small non-protein coding RNA molecules that bind on the transcripts of particular genes, called *targets*, and inhibit their expression. Since this function associates them to the causes and the treatment of many diseases (e.g., various types of cancer [2, 18]), many efforts have been made to discover the targets of each miRNA molecule.

In recent years, many databases that collect verified miRNA targets have been developed [4, 13]. In almost all cases, their data are created and maintained up-to-date by human curators. This work is not trivial and requires important amount of time, since it consists of examining a large number of scientific publications and trying to identify excerpts that include useful information for the database. Thus, the administration of curated databases is a very tedious task.

TarMiner is a system aiming to facilitate the work of curators for databases that collect miRNA targets. In particular, given a set of publications, TarMiner automatically identifies their sentences that mention experimentally verified miRNA-to-gene interactions.

Our contribution can be summarised as follows:

- We introduce TarMiner, a tool for automatic extraction of verified miRNA-to-gene interactions from the text of scientific publications.
- We perform experiments on real data that evaluate the accuracy of the aforementioned tool. In particular, we measure precision, recall, and F-measure for several configurations of TarMiner.

In contrast to existing automatic methods to extract miRNA targets, TarMiner has the following advantages: (a) it uses the full text of the publications and not only their abstracts to retrieve many missing interactions, (b) it uses a classifier trained on curated data that considers many NLP features to achieve improved precision, and (c) it supports miRNA and gene name recognition for a large set of species.

Outline. Section 2 describes TarMiner's workflow, while Section 3 presents the results of TarMiner's evaluation. Section 4 summarises previous work that is related to TarMiner. Finally, Section 5 summarises the work.

2. TARMINER'S AUTOMATIC INTERACTION EXTRACTION

Given a publication, TarMiner tries to automatically retrieve any miRNA-to-gene interactions described in its sentences. Its workflow is presented in Figure 1. First, TarMiner extracts from the publication all the sentences that contain at least a miRNA and a gene name (e.g., 'mmu-mir-1' and 'FIGN', respectively). Then, using NLP, a feature vector is formed for each miRNA-gene pair found in the aforementioned sentences. Finally, a binary classifier based on the maximum entropy model [5, 6] classifies each miRNA-gene pair either as a miRNA-to-gene interaction or

SSDBM '15, June 29 - July 01, 2015, La Jolla, CA, USA

http://dx.doi.org/10.1145/2791347.2791366.

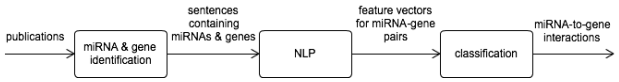

Figure 1: TarMiner's workflow.

as a non-interaction. The next sections describe this process in detail.

2.1 Identifying sentences that contain miRNA and gene names

TarMiner follows the assumption that only sentences which contain at least a miRNA and a gene name might describe miRNA-to-gene interactions. Therefore, identifying miRNA and gene names is a very important task for TarMiner.

2.1.1 miRNA name identification

A simple grammar is adequate for the identification of miRNA names. This is due to the fact that an official nomenclature with some standard variations is used in the literature. In particular, any miRNA name (with a very small number of exceptions) has the following structure: (species prefix)-mir-(miRNA suffix), where (species prefix) is a 3-gram[1] that encodes the species in which the miRNA appears, while (miRNA suffix) is an identifier that differentiates miRNAs from each other. Note that some miRNA names include other tokens (e.g., "let", "lsy", "lin") instead of "mir" (e.g. hsa-let-7a-5p).

In the literature, miRNA names are often used with small deviations. For instance:

- the species prefix is omitted
- the "mir" token is replaced by "miRNA", "microRNA", or "mirn".

TarMiner considers these name deviations; for instance, any of the tokens "miRNA", "microRNA", and "mir" are considered identical to "mir".

Besides, when an article refers to many miRNAs, their names are often presented in a more compact form. For example, the sentence "mir-100, mir-200, and mir-300" can also be written in the following ways:

- mir-100,-200, and -300
- mir100/200/300
- miRNAs -100, -200, and -300
- mir 100, 200 and 300

Considering the aforementioned issues, TarMiner uses the grammar shown in Figure 2 to identify miRNA names.

2.1.2 Gene name identification

There are many databases collecting information about known genes. Although these databases assign well-structured identifiers to the genes, these identifiers are not widely used in the literature. More commonly, genes are referred to by descriptive names that convey their function. The structure of these names makes it difficult to create a grammar that describes them. To complicate things, a publication

[1]Or, in some cases, a 4-gram.

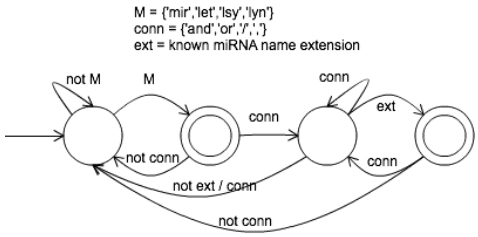

Figure 2: miRNA name recognition grammar.

may contain references to a gene transcript instead of references of the corresponding gene.

It is evident from the above that gene name identification can be a tedious task. To achieve the best possible results, TarMiner uses a large gene name dictionary which contains gene and transcript names and identifiers for many species. Note that many dictionary entries are synonyms. TarMiner uses the tool mygene[2] [15] and NCBI eUtils[3] to find such dependencies among gene or transcript names.

A final issue in recognising gene names is the use of abbreviations. In particular, biomedical terms that refer to diseases, cell tissues, or chemical compounds are often introduced in publications along with an abbreviation. In order to identify whether an abbreviation is used for a gene, TarMiner uses appropriate abbreviation definition extraction software [10].

2.2 Creating feature vectors using NLP

After identifying sentences that contain miRNA and gene names, TarMiner applies NLP on them. The result of this processing is the identification of miRNA-gene pairs and the production of a feature vector for each one of them.

2.2.1 NLP on sentences

Figure 3 summarises the NLP performed on the given sentences. First of all, any stop words are removed. Then, TarMiner performs POS tagging utilising the Stanford Tagger [12], which uses the log linear model to discover the part of speech of each word.

The next step consists of applying stemming to all words. TarMiner uses two stemmers: Morpha Stemmer [7] and Porter Stemmer [9]. The former uses words and their POS tags to extact their lemmas (i.e., to remove their inflections). The latter produces even simpler structures than the lemmas.

After stemming, TarMiner performs phrase chunking on the sentences. A phrase chunker, based on the maximum entropy statistical model, was developed. This software divides any given sentence into noun, verb, or adverbial phrases. The phrase chunker was developed using the maxent module of the NLTK library [1] and was trained on the CONLL 2000 corpus [11] using the MegaM algorithm [3].

Finally, TarMiner performs dependency parsing on the sentences, which extracts grammatical relations among the words of sentences, such as subject-verb, or subject-object relations. The underlying parser we used is the Stanford Dependency Parser (part of the Stanford NLP library). For a given sentence, this parser outputs a directed graph with labels describing the association of words in it.

[2]http://mygene.info

[3]http://www.ncbi.nlm.nih.gov/books/NBK25501/

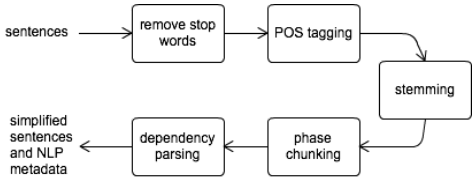

Figure 3: TarMiner's NLP process.

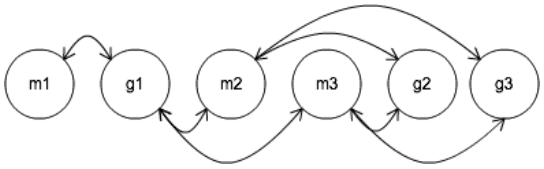

Figure 4: The miRNA-gene pairs extracted from sentence "... m1 ... g1 ... m2 ... m3 ... g2 ... g3".

2.2.2 *Pair extraction*

Many miRNA and gene names may be present in a single sentence. TarMiner extracts distinct pairs of miRNA and gene names from each such sentence. The pair extraction returns all possible pairs, with only one restriction regarding the order of miRNA and gene names in the sentence. In particular, TarMiner extracts miRNA-gene pairs residing in strictly neighbouring sentence blocks, where a block is a continuous part of the sentence containing only one type of names (i.e. miRNAs or genes).

Consider, for example, the sentence "... m1 ... g1 ... m2 ... m3 ... g2 ... g3", where m1-m3 are miRNA names, g1-g3 are gene names, and ... represents arbitrary text. The miRNA-gene pairs extracted from this sentence by TarMiner are represented as bidirected arrows in Figure 4. Due to the limitation mentioned earlier, pairs m1-g2 and m1-g3 are not extracted.

2.2.3 *Filtering out pairs*

One interesting finding of TarMiner's evaluation is that not considering some miRNA-gene pairs during training and classification may result in improved accuracy (see Section 3). For instance, pairs which are mentioned only few times in the publication can be filtered out to achieve both better precision and recall in miRNA-to-gene interactions identification. TarMiner supports this type of filtering. The threshold of the minimum number of sentences in which a pair should appear in a publication, denoted as *minSent*, is a configuration parameter of TarMiner.

2.2.4 *Producing the feature vectors*

For each miRNA-gene pair, TarMiner uses as features some characteristics of the sentences that contain it. These characteristics are extracted based on the results of the performed NLP (see Section 2.2.1). Moreover, for some of the features, TarMiner makes use of a set of words that are commonly used to describe interactions between miRNAs and genes. We refer to this set as the set of *interaction terms*.

TarMiner's features can be divided in the following categories:

- **Dictionary based.** This category consists of features that depend on the existence of a particular interaction term in the sentence.
- **Phrase chunk based.** This category consists of features that consider the last word in any noun phrases that involve miRNA and gene terms.
- **Dependency graph based.** This category has two subcategories. The first uses the shortest paths between gene and miRNA terms on a sentence's dependency graph. Words on these paths are extracted as features. The second category uses word pairs involving miRNA, gene, or interaction terms, extracted from the dependency graph. The extracted feature is the combination of words, as well as their dependency relation.
- **Word location based.** This category contains features that consider the location of words in sentences regardless of their grammatical properties. There are four subcategories: (a) features that consider word pairs preceding miRNA or gene names, (b) features that consider word pairs succeeding miRNA or gene names, (c) features considering single words preceding miRNA or gene names, and (d) features considering intermediate word sequences between miRNA and gene terms.
- **Mixed.** This category combines elements from the previously described categories. It contains four subcategories: (a) features that consider words in the shortest dependency graph paths that also belong to noun phrases of gene or miRNA names, (b) features considering word sequences of up to three words that contain an interaction term and are intermediate of miRNA and gene names, (c) features considering intermediate word sequences between miRNA and gene names containing an interaction term, which is substituted by its POS tag, and (d) features that consider intermediate word sequences between miRNA and gene names containing an interaction term, where all words but the interaction term are substituted by their POS tags.

In addition to the previously described features, TarMiner also makes combined use of them. Note that each of the previously mentioned features corresponds to an element of each candidate pair's feature vector. The value of each element is set to 1 if the feature holds for at least one sentence that contains the pair and 0 otherwise.

2.3 Classification

TarMiner's classifier gets a set of miRNA-gene pairs represented as feature vectors and judges whether there is an interaction between the miRNA and the gene. The decision is based on TarMiner's trained model (see Section 2.4 for details about training).

The output of TarMiner's classifier is the probability of each miRNA-gene pair to correspond to an actual interaction. Thus, given a pair m1-g1, if the probability is greater or equal to 0.5, then the classifier decides that there is an interaction between m1 and g1. Otherwise, the co-occurrence of the miRNA and the gene name in the same sentence is assumed random.

Characteristic	Value
Num. of publications	1,236
Num. of ver. miRNA targets	2,869

Table 1: Characteristics of the dataset used for Tarminer's evaluation

2.4 Training the classifier

TarMiner's binary maximum entropy classifier is trained on a set of feature vectors generated by sentences of publications contained in PMC database[4]. Each feature vector is marked as an interaction or non-interaction based on data from TarBase v7 [13], a human curated database that collects miRNA-to-gene interactions from life sciences publications. MegaM software [3] is executed with the previous input to perform the classifier's training. In fact, the training calculates the proper weight for each feature in order to achieve the optimal classification of the feature vectors.

Recall that, for each publication in the training set, TarMiner considers only pairs appearing in at least minSent sentences (see Section 2.2.3). Moreover, any publications that contain more than *maxInter* miRNA-to-gene interactions are not used for training (and testing) of TarMiner. This is because a large number of interactions are not expected to be described in a sentence of the article (e.g., they may be described in a figure or table). Considering these publications during training are going to falsify training. Furthermore, publications focusing on fewer interactions are expected to provide richer textual information. This is because, in case that few interactions are described, this may imply they are more meticulously presented, e.g. through conducting more experiments.

3. EVALUATION

TarMiner's workflow was developed by combining third-party software with custom Python code. For the needs of the evaluation, a dataset containing open-access scientific publications, along with all verified miRNA-to-gene interactions inside them, was constructed. Table 1 summarises the characteristics of this dataset. The text of the publications was derived by PMC[5], while the verified interactions by TarBase v7 database.

The aforementioned dataset was used to perform 5-fold cross validation for the experiments. Each time, it was divided in a training and a testing set. The former consisted of the 2/3 of the publications, while the latter of the rest. The values of precision, recall, and F-measure, presented later in this section, are the average of the values observed for the same measures observed for each repetition of the experiments.

Table 2 presents the values of the aforementioned accuracy measures calculated during TarMiner's evaluation. TarMiner was trained and tested using varying values of minSent (see Section2.2.3) and maxInter (see Section 2.4) filters. First of all, precision is high (around 0.8) for almost any of the examined TarMiner configurations. Recall is very high (around 0.75) for small values of maxInter, however, it decreases rapidly for larger values. F-measure also follows the same trend. The reason for this drop in recall and F-measure

[4]http://www.ncbi.nlm.nih.gov/pmc/about/intro/
[5]http://www.ncbi.nlm.nih.gov/pmc/

Precision

minSent / maxInter	1	5	10
1	0.8168	0.7901	0.7610
2	0.8614	0.8196	0.8197
3	0.8461	0.8151	0.8096
4	0.8157	0.8443	0.8081

Recall

minSent / maxInter	1	5	10
1	0.7698	0.6103	0.5069
2	0.7618	0.5784	0.5184
3	0.7346	0.5941	0.5444
4	0.7547	0.5991	0.5317

F-measure

minSent / maxInter	1	5	10
1	0.7924	0.6883	0.6081
2	0.8080	0.6780	0.6345
3	0.7863	0.6871	0.6506
4	0.7834	0.7004	0.6411

Table 2: TarMiner's precision, recall, and F-measure for varying thresholds of minSent and maxInter filters.

is that TarMiner focuses on interactions mentioned in sentences containing miRNA and gene names. However, this is not a convenient way to provide information about a large number of miRNA targets. In many cases, this information is contained in supplementary figures or tables. Identifying such interactions is beyond the scope of TarMiner, since its focus is on information expressed in natural language. Another case is that, often, the authors introduce custom notation to refer to the set of genes they examine. As a result, TarMiner may fail to identify these genes in the text and will fail to retrieve their interactions.

Moreover, it is evident that, in some cases, larger values of minSent threshold achieve improved values of precision. This is because, often, miRNA-to-gene interactions that are verified by the experiments described in a publication are mentioned many times in the publication's text. Increasing the minSent threshold means that the miRNA-to-gene pairs used for training and testing have larger probability to correspond to an actual interaction. Therefore, the classifier is more accurately trained.

4. RELATED WORK

In recent years, a multitude of databases that record information about miRNAs and their targeted genes and diseases have been developed. In most cases, thorough manual curation is used to initialise and update these databases. Much of the effort in the curation process is related to the extraction of useful information that is included in the text of scientific publications. This is why some databases have integrated automated techniques in their workflows to assist curators.

Regarding experimentally verified, as well as algorithmically predicted targets, databases such as *MiRecords* [16] and *MiRDB* [14] have been developed. Most of the validated targets recorded in these databases are captured through human curation of publications (MiRDB also performs trivial text mining, however, only to identify the importance of each miRNA molecule). *MiRTarBase* [4] and *TarBase* [13]

are databases that focus on experimentally verified miRNA-to-gene interactions. Both are based on curators surveying scientific articles. TarBase also exploits text mining on the articles' titles and abstracts to find those that are considered more likely to present findings on miRNA targets to assist its curators in finding interesting material.

MirSel [8] is a text-mining based repository collecting miRNA targeted genes and proteins. Its goal is to supplement other repositories. Text mining is executed on publication abstracts. The extraction of relationships is based on the co-occurrence of miRNA and gene names, as well as on particular dictionary terms that describe associations of biomolecules.

Finally, another work that utilises text mining techniques, however to identify miRNA association to cancer types, is MiRCancer [17]. It utilises common sentence structures that are used for describing the expression of miRNA molecules in cancer. These structures are used to create a rule set against which sentences of publication abstracts are matched. Matching sentences are then extracted and manually curated. Only relationships that are verified from these sentences are stored in the database. After proper adaptation, MirCancer's text mining could be used to identify miRNA-to-gene interactions.

Note that none of the existing automatic methods (i.e., MirSel, MiRCancer, and the text mining phase of TarBase) use full text of publications and none of them support 8 species. Moreover, they utilise rules on how the co-occurrence of a miRNA name, a gene name and a term indicating an association between them in the same sentence, suggest a miRNA-to-gene interaction. This approach may result in many false interactions since the meaning of a sentence may be altered by its context. On the other hand, TarMiner uses a classifier trained on curated data that considers many NLP features and, thus, is able to filter out such false positives.

5. CONCLUSION

To conclude, we presented TarMiner, an automated tool for identifying verified miRNA-to-gene interactions presented in scientific texts. Our approach consists of (a) applying NLP to construct feature vectors for each contained miRNA-gene pair, and (b) classifying each pair as interaction or non-interaction based on the binary maximum entropy model. TarMiner's classifier is trained using data from PMC and TarBase v7 databases.

We evaluated TarMiner applying 5-fold cross validation. The experiments revealed satisfying precision and recall for publications containing up to ten interactions. To the best of our knowledge, no prior work used a trained classifier for the task of miRNA-to-gene interaction recognition and no prior method used curated data for evaluating its performance in terms of correctly identified targets.

Acknowledgements. This work was performed in the framework of MEDA project within GSRT's KRIPIS action, funded by Greece and the European Regional Development Fund of the European Union under the O.P. Competitiveness and Entrepreneurship, NSRF 2007-2013 and the Regional Operational Program of ATTIKI.

6. REFERENCES

[1] S. Bird, E. Klein, and E. Loper. *Natural language processing with Python.* " O'Reilly Media, Inc.", 2009.

[2] R.W. Carthew. Gene regulation by micrornas. *Curr. Opin. Genet. Dev.*, 16(2):203–208, 2006.

[3] H. Daumé III. Megam: Maximum entropy model optimization package. *ACL Data and Code Repository, ADCR2008C003*, 50, 2008.

[4] S. Hsu, Y. Tseng, S. Shrestha, Y. Lin, A. Khaleel, C. Chou, C. Chu, H. Huang, C. Lin, S. Ho, et al. mirtarbase update 2014: an information resource for experimentally validated mirna-target interactions. *Nucleic Acids Res.*, 42(D1):D78–D85, 2014.

[5] E.T. Jaynes. Information theory and statistical mechanics. *Phys. Rev.*, 106(4):620, 1957.

[6] E.T. Jaynes. Information theory and statistical mechanics. ii. *Phys. Rev.*, 108(2):171, 1957.

[7] G. Minnen, J. Carroll, and D. Pearce. Applied morphological processing of english. *Natural Language Engineering*, 7(03):207–223, 2001.

[8] H. Naeem, R. Küffner, G. Csaba, and R. Zimmer. mirsel: automated extraction of associations between micrornas and genes from the biomedical literature. *BMC Bioinformatics*, 11(1):135, 2010.

[9] M.F. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

[10] A.S. Schwartz and M.A. Hearst. A simple algorithm for identifying abbreviation definitions in biomedical text. In *Proceedings of the 8th Pacific Symposium on Biocomputing*, pages 451–462, 2003.

[11] E.F. Tjong Kim Sang and S. Buchholz. Introduction to the conll-2000 shared task: Chunking. In *Proceedings of the 2nd workshop on Learning language in logic and the 4th conference on Computational natural language learning-Volume 7*, pages 127–132. Association for Computational Linguistics, 2000.

[12] K. Toutanova, D. Klein, C.D. Manning, and Y. Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 173–180. Association for Computational Linguistics, 2003.

[13] I.S. Vlachos, M.D. Paraskevopoulou, D. Karagkouni, G. Georgakilas, T. Vergoulis, I. Kanellos, I. Anastasopoulos, S. Maniou, K. Karathanou, D. Kalfakakou, et al. Diana-tarbase v7. 0: indexing more than half a million experimentally supported mirna: mrna interactions. *Nucleic Acids Res.*, 43(D1):D153–D159, 2015.

[14] X. Wang. mirdb: a microrna target prediction and functional annotation database with a wiki interface. *Rna*, 14(6):1012–1017, 2008.

[15] C. Wu, I. MacLeod, and A.I. Su. Biogps and mygene. info: organizing online, gene-centric information. *Nucleic Acids Res.*, page gks1114, 2012.

[16] F. Xiao, Z. Zuo, G. Cai, S. Kang, X. Gao, and T. Li. mirecords: an integrated resource for microrna–target interactions. *Nucleic Acids Res.*, 37(suppl 1):D105–D110, 2009.

[17] B. Xie, Q. Ding, H. Han, and D. Wu. mircancer: a microrna–cancer association database constructed by text mining on literature. *Bioinformatics*, page btt014, 2013.

[18] L. Zhang, S. Volinia, T. Bonome, G.A. Calin, J. Greshock, N. Yang, C. Liu, A. Giannakakis, P. Alexiou, K. Hasegawa, et al. Genomic and epigenetic alterations deregulate microrna expression in human epithelial ovarian cancer. *P. Natl. A. Sci.-Biol.*, 105(19):7004–7009, 2008.