Doctoral Dissertation

Computational Methods for Short Interfering RNAs

by

Bui Ngoc Thang

Supervisor: Professor Ho Tu Bao

School of Knowledge Science
Japan Advanced Institute of Science and Technology

March 2015

Abstract

In 2006, Fire and Mello received their Nobel Prize for their contributions to research on RNA interference (RNAi). Their work and that of others on discovery of RNAi have had an immense impact on biomedical research and will most likely lead to novel medical applications to design novel drugs for treating many kinds of diseases such as influenza A virus, HIV, Hepatitis B virus, cancer and so on. RNAi is the biological process in which short interfering strand RNA (siRNA) target and silence the target gene (mRNA). In RNAi, siRNAs can be synthesized and injected in to the cell to silence mRNAs, i.e, to control the diseases. However siRNAs can target and silence the same mRNA different efficacy and siRNAs can also silence unrelated mRNAs. Therefore synthesizing highly effective siRNAs to design novel drugs is one of the most crucial issues on RNAi research.

Research on siRNAs can be seen by consecutive generations each characterized by its typical problems. The first generation focuses on the problem of finding effective siRNA design rules where each effective siRNA design rule is composed by important characteristics of siRNAs influencing to their knockdown efficacy. In this generation, many effective siRNA design rules were found out by biological empirical processes and applying machine learning techniques. The second generation focuses on the problem of building predictive models to predict knockdown efficacy of siRNAs. Machine learning techniques have been alternatively and mostly employed to solve this problem. However, following limitations remain: most of siRNA design tools have low performance and many siRNAs generated by these effective siRNA design rules are inactive or ineffective. Performance of the proposed models is also still low and decreases when tested on independent datasets. As a result, finding solutions for the two above problems in order to generate highly effective siRNAs is still a great challenge. Due to those limitations, the next generation of methods for generating highly effective siRNAs has mostly not appeared.

Our research focuses on contribution to overcome the above–mentioned limitations in the first two generations. On the first problem, we proposed two effective siRNA design rules by developing a new descriptive method. This method not only detected characteristics of previous design rules but also discovered new positional characteristics to design effective siRNAs. On the second problem, we proposed computational methods to build better predictive models for predicting the siRNA knockdown efficacy. The key idea is not only focusing on learning algorithms but also exploiting results of the empirical processes to enrich the siRNA representation by incorporating siRNA design rule, and using labeled as well as scored datasets. Based on experimental evaluation, our proposed predictive models achieved better performance than all models recently reported in the literature.

Keywords: RNAi, siRNA, siRNA design rule, semi–supervised learning, bilinear tensor regression.

Acknowledgments

I wish to express my sincere gratitude to my principal advisor Professor Ho Tu Bao of Japan Advanced Institute of Science and Technology. He encouraged and teach me the way to become a researcher. He not only spent a lot of time to guide me in my research but also helped and teach me many problems in my life. Becoming his student is one of biggest chances for me.

I would like to thank the Ministry of Education and Training, Vietnam (MOET) for providing finance support, and I am grateful to Japan Advanced Institute of Science and Technology for providing me a good environment to study. I also wish to express my thanks to the University of Engineering and Technology, Vietnam National University, Hanoi for supporting necessary conditions for me to study.

I would like to show my gratitude to Professor Tatsuo Kanda who suggested useful information on RNAi. I have had a great opportunity to work with him in the consecutive research projects with my laboratory.

I also wish to express my thanks to doctor Le Si Vinh of the University of Engineering and Technology, Vietnam National University, Hanoi for his suggestions and continuous encouragements.

Finally, I dedicate this dissertation to my wife, Nguyen Thi Thanh Huyen, my son, Bui Nhat Huy, and my family members for their loves and encouragement during the time period of my PhD student

Contents

Chapter 1

Introduction

The first part of this chapter presents the overview of RNA interference. In the second part, the avoiding off–target effects of siRNAs and generating highly effective siRNAs are discussed in more detail following biological and computational biology approaches. The third part presents our problem formulation and contributions. The final one presents thesis organization.

1.1 RNA interference

RNA interference (RNAi) is a cellular process for sequence specific destruction of mRNA. Long double stranded RNA duplex or hairpin precursors are cleaved into short interfering RNAs (siRNAs) by the ribonuclease III enzyme Dicer. Guided by RNA induced silencing complex (RISC), the siRNAs bind to their complementary target mRNAs and induces degradation of mRNAs. Therefore, the translation process of the mRNA into protein will be prevented and infection by RNA viruses can be blocked. RNAi occurs in the process of post–transcriptional gene silencing (PTGS), which involves numerous cellular proteins besides the RNA. RNAi process is strongly conserved in eukaryotes and presumably serves as a protection against viruses and genetic instability arising from mobile genetic elements such as transposons.

In 2006, Fire and Mello received Nobel prize for their contributions to RNA interference (RNAi). Their contributions as well as other research groups' to discovery of RNAi have already had an immense impact on biomedical research and will most likely lead to novel medical applications in the future.

1.1.1 The mechanism and main components of RNAi

The basic processes involved have been determined in detail. In the first step of the RNAi process (Figure 1.1 B), the endonuclease Dicer cleaves the long dsRNA into siRNAs. In

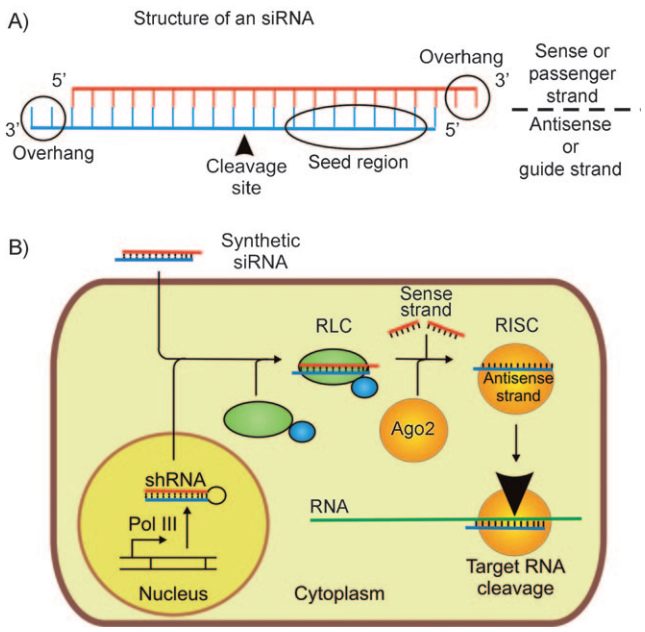


Figure 1.1: The RNAi scheme and short interfering RNAs (siRNAs). Source: [Kurreck , 2009]

the second step, siRNAs are unwinded into sense and antisense siRNA strands by the RNA induced silencing complex, which guides the antisense siRNAs to the complementary target RNAs. As a result, the target RNAs are cleaved at a specific site in the center of the siRNA (10 nucleotides from the 5' end) [Elbashir *et al.*, 2001]. The catalytic component that cleaves the target RNAs, has been identified as the protein designated Argonaut 2 (Ago2). By analyzing the crystal structure of Ago2, it shows Ago2 contains a domain which is similar to RNase H, that cleaves the RNA component of a DNA/RNA duplex [Liu *et al.*, 2004]. After cleavage, the target RNAs lack elements which are typically responsible for stabilizing mRNAs so that the cleaved mRNAs are rapidly degraded by RNases and the protein can not be synthesized from these mRNAs.

There are the three main components involve the RNAi process: siRNAs, enzyme Dicer, and RNA induced silencing complex (RISC). siRNAs are a short interfering double-stranded RNA (dsRNA) that are 21–23 nucleotides with phosphorylated 5' ends and hydroxylated 3' ends with two overhanging nucleotides (Figure 1.1 A). Dicer is an endoribonuclease in the RNase III family that cleaves double-stranded RNA into short double-stranded RNA fragments (siRNAs) and RISC is a multi–protein complex that incorporates one strand of siRNA or micro RNA (miRNA) to target messenger RNA (mRNA).

Figure 1.2: The co–suppression of petunia plants. The left plant is wild-type; the right plants contain transgenes that induce suppression of both transgene and endogenous gene expression. Source: Wikipedia

1.1.2 RNA interference in plants

In plants, the RNA silencing was discovered when searching for transgenic petunia flowers that were expected to be more purple. In 1990, R. Jorgensens laboratory up-regulated the activity of a gene for chalcone synthase (chsA), an enzyme involved in the production of anthocyanin pigments (Figure 1.2). As a result, some of the transgenic petunia plants harboring the chsA coding region under the control of a 35S promoter lost both endogene and transgene chalcone synthase activity, and thus many of the flowers were changed their color or developed white sectors [Napoli *et al.*, 1990]. The loss of cytosolic chsA mRNA was not associated with reduced transcription, as demonstrated by run on transcription tests in isolated nuclei [Van Blokland *et al.*, 1994]. The term "co–suppression" was employed to describe the loss of mRNAs of both the endo and the transgene.

During this period of time, other laboratories [Ingelbrecht *et al.*, 1994] also found that the transcribing-sense genes could down–regulate the expression of homologous endogenous genes. Subsequently, many similar events of co–suppression were reported in the literature. All cases of cosuppression resulted in the degradation of endogene and transgene RNAs after nuclear transcription had occurred [Kooter *et al.*, 1999]. Since post–transcriptional RNA degradation was observed in a wide range of transgenes expressing the plant, bacterial, or viral sequences, it was renamed post–transcriptional gene silencing (PTGS). PTGS can be started not only by sense genes but also by antisense genes, and biochemical evidence suggests that similar mechanisms might operate in both cases [Francesco *et al.*, 2001]. It is to point out that although the co–suppression phenomenon was originally observed in plants.

Around the same time, the observed alterations in the PTGS-related phenotypes led to found multiple site integrations, aberrant RNA formations, repeat structures of the transgenes, and so on. Later on, it became clear that the expression of the transgene led to the formation of dsRNA, which initiated PTGS.

Reports from several laboratories in the past few years had discovered that the loss in steady-state accumulation of the target mRNA is almost total if the designed transgene construct of the transgenic plant produces the nuclear transcript in the duplex conformation. This evidence points out that the production of dsRNA is required to initiate PTGS in plants. Based on this, plants carrying strongly transcribing transgenes in both the sense and antisense orientations are currently being produced that show strong PTGS features. These transgenic plants can silence endogene, invading viral RNA, or unwanted foreign genes in a sequence specific and heritable manner.

In summary, the sense and antisense components of the above–mentioned transgenes are separated only by an intron to increase the efficacy of PTGS [Chuang *et al.*, 2000, Smith *et al.*, 2000]. For example, Arabidopsis thaliana and Lycopersicon esculentum (tomato) plants were transformed with a transgene construct designed to generate self–complementary iaaM and ipt transcripts. iaaM and ipt are oncogenes of agro–bacteria that are responsible for crown gall formation in infected plants. The transgenic lines retained susceptibility to Agro–bacterium transformation but were highly refractory to tumorigenesis, providing functional resistance to crown gall disease by post–transcriptional degradation of the iaaM and ipt transcripts [Escobar *et al.*, 2001].

1.1.3 RNA interference in mammalian cells

This section uses materials taken from [Kurreck , 2009]. The first gene silencing technique was applied in eukaryotes such as plants, C. elegans or D. melanogaster but not in mammals because in the mammalian cells, long double strand RNA sequences (dsRNAs) cause unspecific interferon (IFN) response. It is the reason why the dsRNAs in these cells were considered as pathogens and the protein synthesis is inhibited by protein kinase R [Clemens *et al.*, 1997]. However, in 2001 Tuschl and colleagues shown that 21–nucleotide siRNAs could degrade mRNAs in the mammalian cells including human embryonic kidney (293) and HeLa cells [Elbashir *et al.*, 2001]. It led to create new opportunities for researchers to study gene function and gene–specific therapeutics. Phosphate groups at the 5' ends of the pre–synthesized siRNAs were modified by the kinase Clp1 after the synthetic siRNAs were injected into the cells [Weitzer *et al.*, 2007]. These siRNAs combined to RISC and inhibited mRNAs as above mentioned (Figure B).

Antisense oligonucleotides have been employed to inhibit the translation process from mRNAs to proteins. Furthermore, in 1998 Fire and Mello shown the important role of dsRNA, RNAi has been considered as expansion of the antisense oligonucleotide strategy.

Antisense and RNAi strategies have some common things such as the necessity of the binding site on mRNAs, the stability of RNAs based on chemical modification and the deliver to transport dsRNAs to the target RNAs. The RNAi can be employed the

antisense oligo–nucleotide strategy to speed up its progress [Corey *et al.*, 2007]. However, the two strategies have some differences: the antisense oligo–nucleotide strategy uses single antisense strand RNAs to target and cleave mRNAs in the nucleus of the cells. In contrast, RNAi employs double strand RNAs to target and inhibit mRNA in the cytoplasm. In RNAi, the primary protein of RISC is Ago2 meanwhile antisense use RNase H to activate.

After introducing siRNAs into the cells, mRNAs can be immediately inhibited. The inhibition process can occur during 48 hours. However, by chemical modification of siRNAs, this process is more stable and longer. In RNAi, siRNAs only degrade mRNAs at different levels but not knock them out. Therefore, RNAi is considered as a knockdown technology.

Some experiments shown that the inhibition of siNRAs to the target genes is stable over time period of approximately five days in vitro [Watanabe *et al.*, 2004] and in vivo [Christoph *et al.*, 2006]. In different species, siRNAs can inhibit mRNAs at different periods of time. For example, an siRNA targeted and degraded the apolipoprotein B in mice for only a few days and after the knockdown efficacy of siRNAs was decreased to 70%, whereas the knockdown in nonhuman primates was still stable to inhibit mRNAs efficiently after 11 days [Zimmermann *et al.*, 2006]. Therefore, the time period for inhibition ability of an siRNA can depend on numerous factors, such as the target organ, the target mRNA, and the species [Kurreck , 2009].

It has also been reported that RNAi not only a process to inhibit target genes but also can act as platforms for siRNA-mediated chromatin modifications [Buhler *et al.*, 2007]. This has been observed in some spieces such as yeast, plants, and fruit flies. However, the importance of RNAi in mammalian cells has not been clearly studied.

1.1.4 Applications of RNA Interference

Materials in this section about the common applications are also taken from [Kurreck , 2009].

Investigation of Gene Function

Eukaryotic model organisms and human genome sequencing techniques is one of the most developments in the last decades. However, analysis of gene function is still a challenge. Thus, it is considered as the most important problem. The discovery of RNAi led to make a big chance to analyze gene function. This led to the adoption in only a few years of RNAi as a standard method of molecular biological research that is employed in a very large number of biochemical laboratories.

The inhibition is based on the matching between mRNA and siRNA, so analysis of gene functions can be significantly faster. By selecting appropriate mRNAs to analyze their functions, isoforms of protein can be selectively turned off [Warnecke *et al.*, 2004]. The

main goal of a pharmaceutical project is to design and produce traditional drug, RNAi promises the evaluation of novel function for mRNAs [Chatterjiee–Kishore *et al.*, 2005].

Therapeutic Applications

The clinical development of antisense oligonucleotides [Crooke *et al.*, 2004] and ribozymes [Schubert *et al.*, 2004] was utilized in the therapeutic application of siRNAs. Therefore, the first RNAi treatments were started on humans just three and a half years after siRNAs were first employed in mammalian cells. The basic difference between antisense oligonucleotides and siRNAs are their size, and it is very expensive and difficult to synthesize these oligomers. Furthermore, sense and antisense strands of the siRNAs must be synthesized separately.

Eye Diseases

The eye is the organs of vision with low nuclease activity in which the active agent can be injected intravitreally easily. The only two oligonucleotides approved by the American Food and Drug Administration were employed to treat eye diseases. The first RNAi–based clinical studies were started at the end of 2004 that an siRNA targeted and degraded vascular endothelial growth factor (VEGF). The siRNA has been tested under the name Bevasiranib in a phase III trial by the company Opko Health.

By chemical modification of siRNA, Sirna Therapeutics initiated the first RNAi–based clinical studies. The siRNA was stabilized by unpaired deoxythymidine with a phosphorothioate bond and inverted a basic sugar residues on the ends of the antisense and sense strand, respectively.

In a further clinical study, the siRNA RTP801i-14 against the hypoxia-induced gene rtp801 was used for the treatment of age–related macular degeneration according to Quark Pharmaceuticals. This approach is possibly safer and more efficient than the anti-VEGF substances.

Viral Infections

The analysis of viral infections is one of the most important medical problems that researchers focus on. Viral infections such as HIV–1, HBV and HCV are continually increasing. Furthermore, variants of viruses such as the influenza virus H5N1, SARS is very difficult to be treated. New dangers from viruses must be expected when viruses move from animals to human and vice versa. Although, antiviral agents have been developed, there are only few approved drugs for the treatment of viral diseases. Therefore, new antiviral strategies is very necessary to be developed.

The complementary base pairing of a mRNA and the antisense strand siRNA allows to apply RNAi technology to any given variant of a virus or to new types of virus. This is one of great advantages of RNAi comparing with conventional approaches, which require time–consuming optimization of small–molecule substances. Since the first reports about the antiviral effects of siRNAs against respiratory syncytial virus (RSV) [Bitko *et al.*, 2001], other successful RNAi applications against most classes of medically relevant viruses, including HIV-1, HBV, HCV, SARS-coronavirus, influenza virus, polio virus, and coxsackie virus, have been published [Kurreck , 2009].

An important role in RNAi approaches against viruses is to find and target intended mRNAs. Because viral RNAs have significant secondary structure, it is difficult for siRNAs to inhibit them efficiently.

As above mentioned, the time period of active siRNA is over approximately 5 days, so viral escape becomes one of biggest problems for RNAi. For both the polio virus [Gitlin *et al.*, 2005] and HIV [Boden *et al.*, 2003], cases have been described in which viral replication can be inhibited efficiently at the beginning, but after the virus titer increases again, because of their mutants which can tolerance the inhibition of siRNAs.

Cancer

RNAi has been considered as a promising approach to treat cancer [Pai *et al.*, 2006]. The gene expression which leads to develop the tumor to create new blood vessels to supply the tumor can also be inhibited. In addition, metastasis of cancer is the real difficult problem even that primary tumors are surgically removed. RNAi can be employed to degrade resistance of tumors to treatment with chemotherapeutic agents or radiotherapy. If chemotherapeutic agents are used to treat cancer, tumors can be resistant through the expression of the multidrug resistance (MDR) gene. However, siRNAs can degrade the MDR expression in which tumor cells become vulnerable to chemotherapeutics [Wu *et al.*, 2003].

Many published research show that RNAi has employed to force the tumor growth process develop slow. For example, siRNAs against CD31 inhibit the growth of tumors in various xenograft mouse models [Santel *et al.*, 2006]. The siRNAs penetrate into the tumor endothelial cells as lipoplexes and block angiogenesis.

Further Clinical Trials

In a further clinical study, RNAi is being employed to treat acute kidney failure. It has been shown that the siRNA AKli-5 can inhibit the tumor suppressor p53 to prevent the cell damage [Komarov *et al.*, 1999]. However the period of time for the inhibition of the tumor suppressor was limited. The safety of AKli-5 is to be tested in a phase I trial in patients who have a high risk of kidney failure because of a major cardiovascular

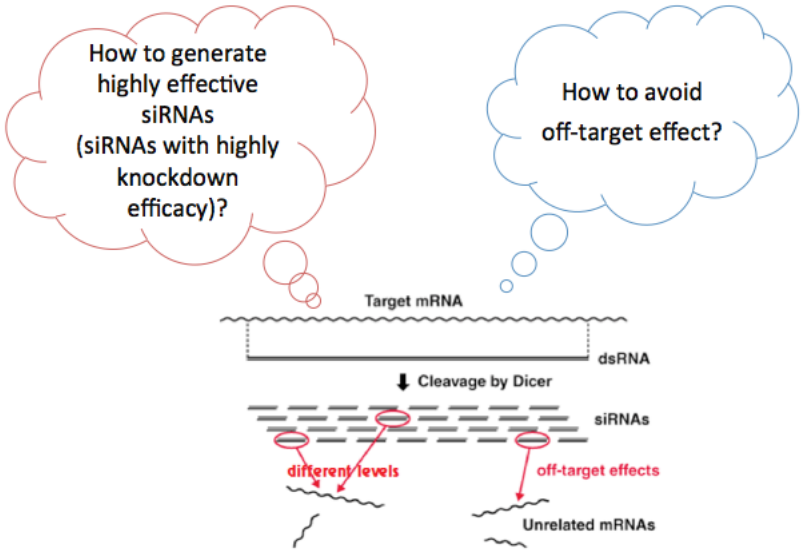


Figure 1.3: The two crucial problems in RNAi. The first problem is how siRNA can avoid off–target effects. The second one is how to generate highly effective siRNAs.

operation.

In January 2008, TransDerm Inc. started a clinical research to inhibit the autosomal-dominant genetic disease, Pachyonychia congenita. The siRNA was injected and it targeted and silenced the expression of the keratin mutation K6a [Smith *et al.*, 2008].

1.2 siRNA research

In mammalian cultured cells, RNAi is typically induced by the use of short interfering RNAs. siRNAs are generally 21 bp double-stranded RNA molecules with di–nucleotide overhang at 3' ends (Figure 1.1A). They can be introduced directly by transfection or electroporation, or generated within the cell from double strand RNA. Interestingly, siRNA sequences can be synthesized to silence target genes. In practice, good experimental design dictates that functional siRNAs to the same target should be used independently to ensure that the biological effect is due to silencing of the target gene. However, by empirical analysis, biologists reported that the efficiency of different siRNAs against the same target RNA varies drastically [Holen *et al.*, 2006], and siRNAs can also target unrelated mRNAs [Jackson *et al.*, 2003] that was called the off–target effects of siRNAs. In RNAi, synthesizing siRNAs that have highly knockdown efficacy and avoid off–targets effects, is a very important issue to design novel drugs. Therefore, the two following crucial problems (Figure 1.3) have to be significantly considered: (i) how siRNAs avoid off–target effects and (ii) how to generate highly effective siRNAs. These two problems are discussed in more detail in the remain of this section.

1.2.1 Avoiding off–target effects of siRNAs

Off–target effects occur when an siRNA is processed by the RNA–Induced Silencing Complex (RISC) and silences unintended targets (mRNAs). Off-target effects were first characterized in detail by Jackson and co–workers in 2003 [Jackson *et al.*, 2003]. Using microarray profiling as a method of detection, the authors identified modest, 1.5– to 3–fold changes in the expression of dozens to hundreds of genes following transfection of individual siRNA. The levels of complementarity between the sense or antisense strand of the siRNA and the off-targeted genes varied considerably, and the overall off–target profile was unique for each siRNA, suggesting a sequence specific phenomenon.

Initially, these modest changes in off–target gene expression led many to dismiss the event as inconsequential. Unfortunately, this optimism was recently dispelled by reports that off–target effects could induce measurable phenotypes. More recent investigations have shown that it is not the overall identity of an mRNA with the siRNA, but rather the perfect correspondence between parts of the 3'–UTR and the seed region (positions 2–7 or 2–8) of the antisense strand of the siRNA which determines whether gene expression is influenced (Figure 1.1 A). Du *et al.* [Du *et al.*, 2005] shown that it is not only the position of mismatched base pair, but also the identity of the nucleotides forming the mismatch effecting off-target effects of siRNAs. Thus, off–target effects depend on single nucleotide mismatched targets and mismatched positions. The perfect matching of base pairing in the central region of the target site was found to be critical for the silencing activities, and siRNA is highly sensitive to mismatches in this region. Certain nucleotide mutations at positions 5, 7, 8 and 11 were found to be tolerated fairly well and the expression of the fusion gene was repressed.

The last approach toward eliminating off–target effects is associated with siRNA design. Studies by Birmingham [Birmingham *et al.*, 2006], Lin [Lim *et al.*, 2005] and Jackson [Jackson *et al.*, 2006] revealed that off–targeted genes frequently contained matches between the seed region of the siRNA (positions 2–7) and sequences in the 3' UTR of the off-targeted gene. This means that off–target effects can be reduced by clever design of the siRNA. Thus this promising approach of siRNA that provides both potent (highly effective siRNAs) and specific (off–target effects) gene knockdown. Furthermore, the specificity of siRNAs can be reduced through the incorporation of modified nucleotides. It is comparatively easy to completely inactivate the sense strand by modifications so that the danger of off–target regulation can be reduced to a minimum. On the other hand, changes to the antisense strand are more challenging since knockdown efficacy of siRNAs to target the mRNAs must not be influenced.

Concerning this problem in computational biology approach, a few research groups have proposed scoring functions or models to predict off-target effects ability of siRNAs. The first scoring function was proposed by Alistair and his colleagues [Alistair *et al.*, 2008].

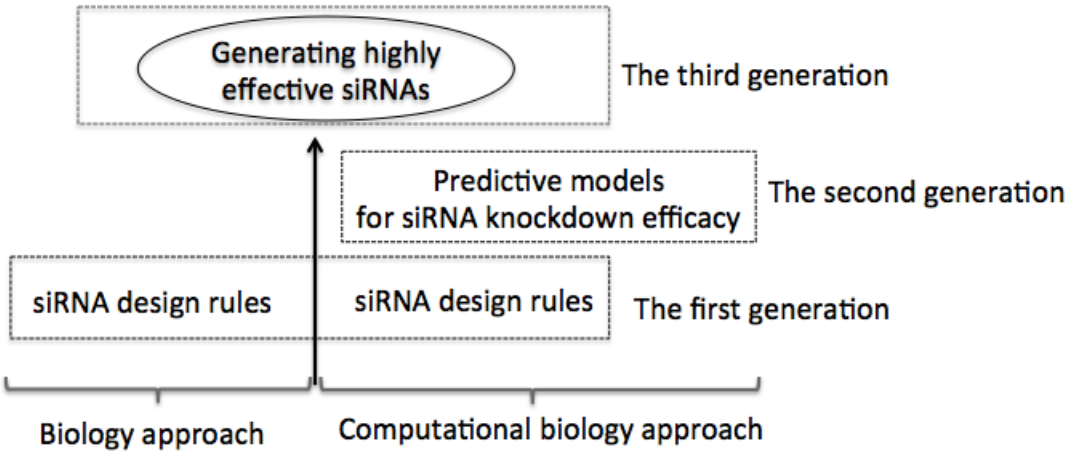


Figure 1.4: Finding highly effective siRNAs in biology and computational biology approaches.

They developed a new specificity scoring scheme based on the results by [Du *et al.*, 2005], which uses experimentally observed off–target effects at each siRNA position. Currently, in 2010, Karol and co–workers [Karol K. *et al.*, 2010] proposed the kernel method to analysis off–target. They developed a method based on sequence alignment kernel which measures sequence similarity based on shared occurrences of length subsequences, counted with up to mismatches. Although, the building a function for off–target effects of siRNAs is considered as an important problem, however features based on particular mismatch positions and mismatch regions between siRNA and mRNA may be insufficient information to build a good predictor and the evaluation of learned predictors is a difficult issue. Therefore, it becomes a challenge problem.

1.2.2 Generating highly effective siRNAs

As above mentioned, siRNAs can be synthesized and introduced into the cell in order to silence target genes. It leads to design many novel drugs based on siRNAs for treating many kinds of diseases. However, siRNAs can target and knockdown the same mRNA at different levels (different knockdown efficacy), therefore generating highly effective siRNAs is a crucial problem. In order to generate highly effective siRNAs, the common sense in biology and computational approaches is to find out important characteristics of siRNAs that influence their knockdown efficacy. As a results, siRNA design rules composing these characteristics have been reported to generate effective siRNAs (Figure 1.4). Besides that, when a number of synthesized siRNAs has become large, alternative machine learning techniques have also applied to build models for predicting knockdown efficacy of siRNAs. These techniques to build predictive models have been considered as the second generation when the first generation, tools for effective siRNA design rules, is based

Figure 1.5: The first problem: siRNA design rules were built in the biological approach.

on small datasets and consists of guidelines in contrast to a quantitative scoring scheme [Ichihara *et al.*, 2007, Mysara *et al.*, 2012 , Sciabola *et al.*, 2013]. Although many siRNA design rules were reported, these design rules have low performance and efficiency. In addition, the performance of the existing predictive models is also low and the population of siRNAs is very large. Therefore, generating highly effective siRNAs problem is still a challenge, so innovative techniques should be proposed to solve this problems. We consider these techniques as the third generation for generating highly effective siRNAs. As a result, problems respectively corresponding to the three above generations are described as follows

Problem 1: Finding effective siRNA design rules (the first generation)

Problem 2: Building predictive models for predicting siRNA knockdown efficacy (the second generation)

Problem 3: Generating highly effective siRNAs (the third generation)

The research context of the generating highly effective siRNAs is discussed in terms of the first and second generations in more detail in the biology and computational biology approaches as follows

Finding effective siRNA design rules (Problem 1) in biological approach

In 1998, Fire and Mello discovered the important role of dsRNAs in RNAi. Although, dsRNAs can be synthesized and injected into the cell in order for the antisense strand to bind the mRNA, the full–length antisense strand was never detected. This led to search for shorter form of the antisense strand, short interfering RNA (siRNA), derived from the dsRNA. In 2001, Elibalshir *et al.* [Elbashir *et al.*, 2001, Elbashir *et al.*, 2001, Elbashir *et al.*, 2002] found that siRNAs having 19–21 nt in length with 2 nt overhangs

at the 3' ends can silence mRNA efficiently when they introduced 19–21 nucleotide dsRNA (siRNAs) into mouse and human cells. Scherer *et al.* [Scherer *et al.*, 2003] reported that the thermodynamic properties (G/C content of siRNA) to target specific mRNA are important characteristics. Soon after these early works, many rational rules for effective siRNAs have been reported [Reynolds *et al.*, 2004, Uitei *et al.*, 2004, Amarzguioui *et al.*, 2004, Hsieh *et al.*, 2004, Jagla *et al.*, 2005] (Figure 1.5). Characteristics of these rules relate to thermodynamic properties, point–specific nucleotides and specific motif sequences. These siRNA design rules are described as follows

Reynolds *et al.* [Reynolds *et al.*, 2004] analyzed 180 siRNAs systematically and found the following eight criteria for improving siRNA selection:

- G/C content 30−52%
- At least 3 'A's or 'U's at positions 15−19
- Absence of internal repeats
- An 'A' at position 19
- An 'U' at position 3
- An 'A' at position 10
- A base other than 'G' or 'C' at position 19
- A base other than 'G' at position 13

Ui-Tei *et al.* [Uitei *et al.*, 2004] examined 72 siRNAs targeting six genes and discovered four rules for effective siRNA designs. They summarized the following characteristics:

- A 'A' or 'U' at position 19
- A 'G' or 'C' effective at position 1
- At least five 'U' or 'A' residues from positions 13–19
- No GC stretch more than 9 nt long

Amarzguioui and Prydz [Amarzguioui *et al.*, 2004] analyzed 46 siRNAs targeting single gene and reported the following six rules for effective siRNA designs based on their literature:

- $\Delta T_3 = T_3 - T_5$, the difference between the number of A/U residues in the three terminal positions at $3'$ and $5'$ ends of sense strand. $\Delta T_3 > 1$ is positively correlated with functional siRNA
- A 'G' or 'C' residue at position 1, positively correlated
- An 'U' residue at position 1, negatively correlated
- An 'A' residue at position 6, positively correlated
- An 'A' or 'U' at position 19, positively correlated
- A 'G' at position 19, negatively correlated.

Hsieh *et al.* [Hsieh *et al.*, 2004] implemented an experiment with 138 siRNAs targeting 22 genes and exploited the following characteristics:

- Nucleotide 'C' is negative at position 6
- Nucleotide 'C' or G is positive and A or U is negative at position 11
- Nucleotide 'A' is positive at position 13
- Nucleotide 'G' is positive at position 16
- Nucleotide 'U' is positive and nucleotide G is negative at position 19

Jagla *et al.* [Jagla *et al.*, 2005] tested 601 siRNAs targeting one exogenous and three endogenous genes and reported four rules in the following way:

- An 'A' or U positive at position 19
- An 'A' or 'U' positive at position 10
- A 'G' or 'C' positive at position 1
- More than three 'A/U's between positions 13 and 19

Although the positional nucleotide characteristics for siRNA design rules are considered as the most important factor to determine effective siRNAs, there are a number of very active siRNAs which do not correspond to the proposed criteria, while numerous other carefully designed siRNAs are inactive. Recently, even the hypothesis that the relative stability of the two ends has an influence on their efficiency has been called into question. Neither in an experimentally investigated set of different siRNAs nor in a comprehensive analysis of published siRNAs or siRNAs posted to databanks could a correlation between the terminal stability of the siRNA and its silencing activity be found. Other characteristics of the siRNA also possibly play a role. In addition, previous empirical analysis only based on small datasets and focused on specific genes. Therefore, these rules may be not enough information to design effective siRNAs.

Studies with antisense oligonucleotides have already shown that the accessibility of the binding region on the target RNA of oligonucleotides is a great importance for the efficiency of silencing. A correspondence between the accessibility for antisense oligonucleotides and siRNAs has been demonstrated [Kretschmer *et al.*, 2003]. In a more comprehensive analysis, the accessibility of target RNAs was predicted by an iterative bioinformatic approach and by experimental RNase H mapping [Overhoff *et al.*, 2005]. The results showed that siRNAs against predicted highly accessible areas were more efficient than those whose target sequence was inaccessible. The relative thermodynamic stability of the two ends of the siRNA proved, in contrast, not to be a suitable criterion for the prediction of the efficiency of an siRNA.

Besides the siRNA itself, the target RNA could also play an important role in silencing. This could help to explain why the expression of some targets is easily inhibited, while

the knockdown of others is more difficult. In a study with several thousand siRNAs, which were conceived for different genes according to the BIOPREDsi algorithm, 70% of the investigated kinase genes were easy silenced, while 6% of the genes could not be down–regulated by up to 10 different siRNAs [Krueger *et al.*, 2007].

Finding effective siRNA design rules (Problem 1) and building predictive models (Problem 2) in computational biology approach

In biological approach, research groups have to spend a lot of time and finance for each empirical test. For that reason, they can also not handle on large datasets. Therefore, it can be a reason the proposed methods in this research are insufficient to design effective siRNAs. Based on the aim of biological research that is to design effective siRNAs, research groups in computational biology research have applied machine learning techniques to build models for finding siRNA design rules and predict knockdown efficacy of siRNAs.

Concerning the finding siRNA design rules problem, Teramoto [Teramoto *et al.*, 2005] and co–workers used Support Vector Machine (SVM) to select effective siRNAs (Figure 1.6). They developed an algorithm for predicting siRNA functionality by using generalized string kernel (GSK) combined with the Libsvm program to extract sequence feature and to classify siRNAs into effective and ineffective classes by representing each siRNA as k–mer subsequences. Based on the coefficient vector of the model, they also detected the top 20 motifs that can be used to discriminate effective and ineffective siRNAs but they could not deduce a siRNA design rule. Ladunga and coworkers [Ladunga *et al.*, 2007] also used the SVMLight package with poly–nominal kernels to train over 2200 siRNA sequences. To represent siRNAs, they used 572 features relating to sequence, thermodynamic and accessibility characteristics. Shigeru Takasaki and his colleagues proposed prediction methods based on neural networks and decision trees for selecting effective siRNA from many possible candidates [Takasaki *et al.*, 2010, Takasaki *et al.*, 2013]. In the first method, the author used K-means algorithm to calculate variances and centers of each Radial Basis Function corresponding to K nodes on the hidden layer. The similarity of two sequences used is Euclidean distance. In the second one, a decision tree is divided into growing and pruning steps. The testing data were used to check the increasing miss–classification error in the tree pruning step. Moreover, he combined two methods to increase the performance of the predictor.

However, these discriminative techniques are potentially unsuitable to detect hidden characteristics of data. The relationships of characteristics are not explicit and visualizable. Neural networks can not guarantee the solution and produce different results when training again with the same data. In their work, the meaning of clusters was not mentioned and Euclidean distance is also not good to assess similarity of each pair of siRNAs. Thus, the K-means algorithm in this case can be low efficiency. Moreover, the decision

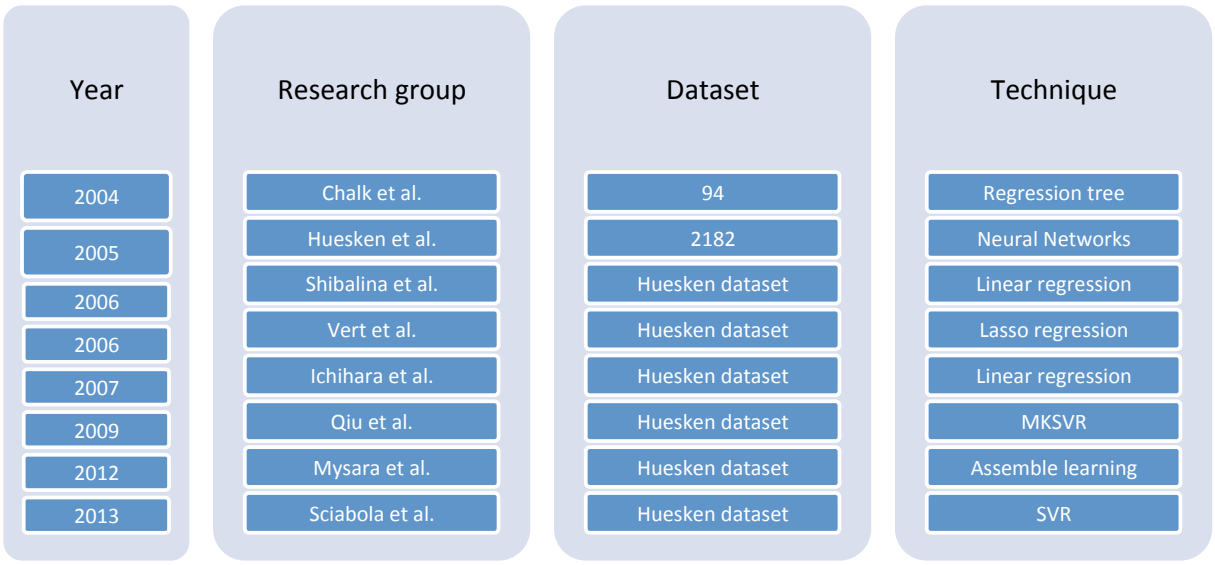


Figure 1.6: The first problem: siRNA design rules were built in the computational biology approach.

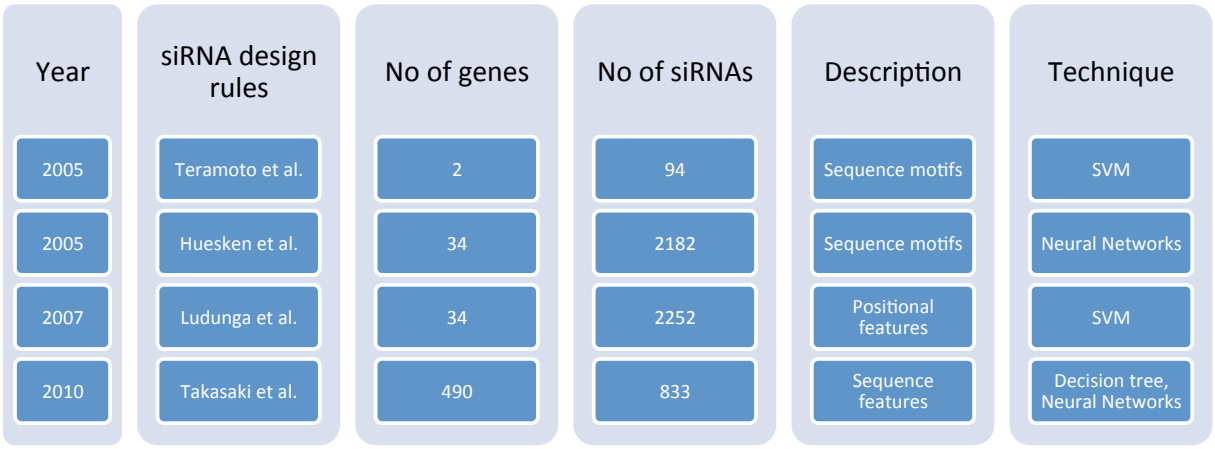


Figure 1.7: The second problem: predictive models were proposed in the computational biology approach.

tree method can not generalize the data well because of overfitting. it can also be unstable because small variations in data may result different trees or different design rules.

Concerning the building predictive models problem, many machine learning techniques have been applied to predict siRNA knockdown efficacy (Figure 1.7). Chalk *et al.* [Chalk *et al.*, 2004] used thermodynamic properties by using the regression tree tool in the BioJava software. According to them, the score of a siRNA candidate is incremented by one for each rule fulfilled giving a score range of (0,7). Huesken *et al.* [Huesken *et al.*, 2005] was proposed the predictive model in which motifs for effective and ineffective siRNA sequences were detected by a artificial neural network (ANN) which trained on 2,182 siRNAs and tested on 249 siRNAs. The BIOPREDsi scoring function was developed by the number of specificities and sensitivities for ANN.

Their dataset was widely used as benchmark to train and test other regression models [Shabalina *et al.*, 2006, Vert *et al.*, 2006, Ichihara *et al.*, 2007, Mysara *et al.*, 2012]. Most notably, Qui and colleagues used multiple support vector regression with numerical and RNA string kernels for siRNA efficacy prediction [Qiu *et al.*, 2009], and Sciabola *et al.* [Sciabola *et al.*, 2013] applied three-dimension structural information of siRNA to increase predictability of the regression model.

It is worth noting that most of those methods suffer from some drawbacks. Their correlations between predicted values and experimental values of the dependent variable ranging from 0.60 to 0.68 were considerably decreased when tested on independent datasets. It may be caused by the fact that the Huesken dataset is still too small to be representative of the siRNA population having about 4^{19} possible siRNAs. In addition, the performance of machine learning methods is heavily dependent on the choice of data representation (or features) on which they are applied. It is a reason why much of the actual effort in deploying machine learning algorithms goes into the design and learning of data transformations that result in a representation of the data that can support effective machine learning. In the previous models, siRNAs are encoded by binary, spectral, tetrahedron, and sequence representations. However, these representations are not good to represent siRNAs in oder to build a good model to predict accurate knockdown efficacy of siRNAs.

Alternatively, several works [Klingelhoefer *et al.*, 2009 , Chang *et al.*, 2012] used classification methods on siRNAs which were experimentally labeled in terms of knockdown efficacy. This siRNA dataset, hereafter called labeled siRNA dataset, was taken from the siRecord database [Ren *et al.*, 2006] consisting of siRNAs classified into four classes with labels 'very high', 'high', 'medium' and 'low'. The classification methods build classifiers from the labeled siRNA dataset to predict the class labels of unknown siRNAs.

1.3 Problem formulation and major contributions

As above–mentioned, generating highly effective siRNAs is one of the two important problems in RNAi to design novel drugs for treating many kinds of diseases. In the biological approach, biologists based on their experiments to detect siRNA design rules that have important characteristics effecting to knockdown efficacy of siRNAs. In the computational approach, machine learning techniques have applied to not only find siRNA design rules but also build predictive models to predict knockdown efficacy of siRNAs. However, they have some above–mentioned limitations:

(i) Design rules are insufficient to select effective siRNAs

(ii) Developed models have low performance and achieve not good results when tested on independent datasets.

In addition, existing design rules can generate many thousands of siRNA candidates and many generated siRNAs were inactive or ineffective. On the other hand, the population of siRNAs is about 4^{19}, so it is very difficult to generate all of siRNAs that a model can predict their knockdown efficacy. Therefore, in order to generate highly effective siRNAs, a promising way is to find out siRNA design rules with high confidence and build better predictive models. The aim of this way is to use these design rules to narrow searching space of effective siRNAs. Based on this searching space, the predictive models can predict siRNAs with highly effective or highly knockdown efficacy. Based on this idea and to overcome above drawbacks, we focus on the two following problems:

Solving Problem 1: Finding a new descriptive method to build rational effective siRNA design rules

To synthesize effective siRNAs, many research groups in both biology and computational biology approaches reported siRNA design rules by analyzing empirical processes and applying machine learning techniques (Figure 1.5, 1.6). However, the performances of siRNA design rules are still slow. Many candidates generated by these design rules were inactive or ineffective ones. In order to find out find out better siRNA design rules, important characteristics of siRNAs should be detected. Therefore, we proposed a new descriptive method to detect rational design rules for effective siRNAs that mostly have characteristics of previous design rules and contain new characteristics influencing knockdown efficacy of siRNAs. The method will detect descriptive rules by adapting the Apriori algorithm with automatic *min_support* values after transforming siRNAs to transactions. The detected descriptive rules were filtered, graphically represented and analyzed to generate design rules for effective siRNAs. This work was reported in the 5th Asian Conference On Intelligent Information and Database Systems.

Solving Problem 2: Developing better predictive models to predict knockdown efficacy of siRNAs

Although, many machine learning techniques have been applied to build models to predict knockdown efficacy of siRNAs (Figure 1.7), these models have some limitations as above–mentioned. In order to improve siRNA knockdown efficacy prediction, our framework is to enrich data representation and build better predictive models.

Concerning the enriching data representation, we transform siRNAs to enriched matrices (the second order tensors) such that on the enriched matrix space, background knowledge of siRNA design rules is incorporated to enrich siRNA representation and clustering property of ordinal labeled siRNAs is also preserved. To this end, transformation matrices are designed, incorporated background knowledge of the siRNA design rules.

Each transformation matrix is learned to transform siRNAs to a component representation (vector representation) of their enriched matrix. For each siRNA, the combination of component representations using these transformation matrices generate a enriched matrix representing this siRNA.

Concerning the building predictive model, because the performance of siRNA design rules are not so high, component representations capturing background knowledge of these design rules are weighted. We also assume that the linear model can predict knockdown efficacy of the enriched matrices well. Therefore, a bilinear tensor regression function is formed and learned by solving an optimization problem.

Based on this framework, the three variant predictive methods were proposed as follows

In the first method, the enriching data representation method and the predictive method were learned independently. We firstly developed data representation learning method to enrich siRNAs. The key idea to learn representations of siRNAs is not only focusing on learning algorithms but also exploiting results of the empirical process to enrich the siRNAs. In the proposed method, transformation matrices were learned by incorporating existing siRNA design rules with labeled siRNAs in the siRecord database. After learning transformation matrices and transforming siRNAs to enriched matrices, we developed a linear tensor regression method to build a better predictive model of the siRNA knockdown ability. In the objective function, the Frobenius norm was appropriately replaced by L2 regularization norm for an effective computation. This work was reported in Pacific–Asia Conference on Knowledge Discovery and Data Mining.

In the second method, the data representation learning and predictive model learning phases are combined together. It means that a siRNA representation learning method was integrated into another proposed bilinear tensor regression method to make more accurate the prediction and precise the siRNA representation. In this tensor regression method, transformation matrices to enrich siRNA representation and parameters of regression model are learned together by integrating scoring siRNAs, labeled siRNAs and siRNAs design rules discovered by empirical processes. In addition, a labeled dataset was used to supervise the parameter learning process of the model. This work was submitted to BMC Bioinformatics Journal and under revision process.

In the last one, learning transformation matrices and parameters of model are similar to that of the second method. However, in this method we used not only existing siRNA design rules but also automatically predictive rules found by the LUPC method [Ho *et al.*, 2003] to enrich siRNA representation.

The contributions of this work are summarized as follows

- Developed and employed computational methods to find effective siRNAs design rules. We firstly used the LUPC method proposed by Ho *et al.* to discover predictive

rules and then developed a descriptive method to detect siRNA design rules. Base on the proposed method, new characteristics that effect to knockdown efficacy of siRNAs were found.

- Developed siRNA representation learning methods that is incorporated important characteristics of siRNA design rules. siRNA sequences are transformed as enriched matrices by learning transformation matrices. Optimization methods to enrich siRNAs using siRNA design rules were proposed.
- Proposed tensor regression methods to predict siRNA efficacy. In the objective functions, L2 norm were used instead of Frobenius norm that allows to learn the set of model parameters effectively. By analyzing the developed models, we quantitatively determined positions on siRNAs where nucleotides can strongly influence inhibition ability of siRNAs and provided guidelines based on positional features for generating highly effective siRNAs. Our proposed models achieve better performance than current models. The proposed methods can be easily extended when a new siRNA design rule is found.
- Developed a predictor called BiLTR using C plus plus programming language.

1.4 Thesis organization

The dissertation will be organized as follows.

Chapter 1 introduces the overview of RNA interference (RNAi). In RNAi research, generating highly effective siRNAs is discussed in more detail in biological and computational biology approaches. Drawbacks of previous research focusing on this problem are pointed out. In this chapter, our objectives and contributions are also presented.

Chapter 2 presents our proposed computational methods in order to build rational siRNA design rules. The first method applied the LUPC algorithm proposed by Ho *et, al.* [Ho *et al.*, 2003] to detect siRNA design rules. The second one is our proposed method that transforms siRNA sequences to transactions and then apply an adaptive Apriori algorithm with automatic *min_support* to detect effective siRNA design rules.

Chapter 3 presents proposed learning methods for data representation to enrich siRNAs. siRNAs are encoded to binary matrices. These binary matrices are then transformed to enriched matrices by transformation matrices that are designed and integrated by existing design rules. The two learning methods were proposed to learn these transformation matrices. The key ideas of the first method are incorporation of siRNA design rules to transformation matrices and properties preservation of ordinal labeled dataset. The second one was integrated into a tensor regression model learning. siRNA design rules, labeled dataset, and scoring dataset were employed together to learn transformation matrices as well as parameters of regression model.

Chapter 4 discusses about our proposed methods "tensor regression models for siRNA efficacy prediction". Our work aims to develop methods for better prediction of the siRNA knockdown efficacy by exploiting both scored and labeled siRNA datasets and available design rules as well as predictive rules detected by the LUPC algorithm. In the developed methods, scoring siRNAs, labeled siRNAs and siRNA design rules are integrated to enrich the siRNA representation and learn prediction models. Experiment results when testing on the test set and the three independent datasets shown that the performance of the proposed models is better that that of current models.

Chapter 5 summarizes the main contributions and achievements. In this chapter, we also discuss about advantages and disadvantages of our methods as well as future works.

Chapter 2

Computational methods for detecting siRNA design rules

This chapter presents the two methods to discover predictive rules that used to design effective siRNA sequences. The first method applied the LUPC algorithm. The second one is our proposed predictive method that transforms siRNA sequences to transactions, and employed an adaptive Apriori algorithm with automatic min_support values to detect descriptive rules. Based on the detected descriptive rules, we designed rational design rules for effective siRNA sequences.

2.1 Introduction

In 2006, Fire and Mello received Nobel prize for their contributions to RNA interference (RNAi). Their contributions as well as other research groups' to discovery of RNAi have already had an immense impact on biomedical research and will most likely lead to novel medical applications in the future. RNAi is a powerful technique for post–transcriptional silencing of messenger RNA (mRNA). In RNAi, double strand RNA (dsRNA) sequences are introduced into cells and cleaved into short interfering RNA sequences (siRNAs). After that, each siRNA binds to its complementary target mRNA and induces its degradation. Therefore, the translation process of the mRNA into protein will be prevented and infection by RNA viruses can be blocked. On RNAi research, designing of effective siRNAs, which can silence mRNA sequences efficiently, is one of the most important challenges. Numerous biological works have been carried out in order to clarify rational design rules to generate effective siRNAs.

In the view point of biological approach, the first rational design rules for siRNAs were proposed by Elibalshir [Elbashir *et al.*, 2001, Elbashir *et al.*, 2001, Elbashir *et al.*, 2002]. They suggested that effective siRNAs having 19–21 nt in length with 2 nt overhangs at

3' end can silence mRNA efficiently. LiSa J Scherer et al. [Scherer *et al.*, 2003] reported that the thermodynamic properties to target specific mRNA are important characteristics. Soon after these early works, many rational rules [Reynolds *et al.*, 2004, Uitei *et al.*, 2004, Amarzguioui *et al.*, 2004, Hsieh *et al.*, 2004, Jagla *et al.*, 2005] for effective siRNAs have been reported. Characteristics of these rules relate to thermodynamic properties, point-specific nucleotides and specific motif sequences.

Although the positional nucleotide characteristics for siRNA design rules are considered as the most important factor to determine effective siRNAs, there exist inconsistencies among proposed design rules. Most of previous design rules have the same statements at position 1 and 19 on siRNAs but have some inconsistencies at other positions. This also implies that these rules might result in the generation of many candidate siRNAs and thus make it dificult to extract a few of them for synthesizing effective siRNAs. Furthermore, previous empirical analysis only based on small datasets and focused on specific genes. Therefore, these rules may be not enough information to design effective siRNAs.

In computational biological approach, some discriminative methods have been applied to find design rules and select effective siRNAs. Chalk *et al.* [Chalk *et al.*, 2004] reported thermodynamic properties by using the regression tree tool in BioJava software. According to them, the score of a siRNA candidate is incremented by one for each rule fulfilled giving a score range of (0,7). Teramoto Chalk *et al.* [Teramoto *et al.*, 2005] and Ladunga [Ladunga *et al.*, 2007] used Support Vector Machine (SVM) to select effective siRNAs. Teramoto adapted the string kernel with Libsvm program to classify siRNAs into effective and ineffective classes by representing each siRNA as k–mer subsequences. Ladunga used SVMLight package with polynominal kernels to train over 2200 siRNA sequences. Huesken *et al.* [Huesken *et al.*, 2005] discovered motifs for effective and ineffective siRNA sequences based on the significance of nucleotides by applying the artificial neural network to train more than 2,400 siRNAs targeting human as well as rodent genes. Shigeru Takasaki [Takasaki *et al.*, 2010] proposed prediction methods based on neural networks and decision trees for selecting effective siRNA from many possible candidates. In the first method, the author used K-means algorithm to calculate variances and centers of each Radial Basis Function corresponding to K nodes on the hidden layer. The similarity of two sequences used is Euclidean distance. In the second one, a decision tree is divided into growing and pruning steps. The testing data were used to check the increasing missclassification error in the tree pruning step. Moreover, he combined two methods to increase efficiency of the prediction.

However, these discriminative techniques are potentially unsuitable to detect hidden characteristics of data. The relationships of charateristics are not explicit and visualiable. Neural networks can not guarantee the solution and produce diffirent results when training again with the same data. In Takasaki work, the meaning of clusters were not mentioned

and Euclidean distance is also not good to assess similarity of each pair of siRNAs. Thus, K-means algorithm in this case can get bad results. Moreover, the decision tree method can not generalise the data well because of overfitting and it can be unstable because small variations in data may result different trees or the different design rules.

To overcome those above drawbacks, we present the two methods to detect siRNA design rules. The first method is LUPC (stands for Learning the Unbalanced Positive Class)[Ho *et al.*, 2003] that can learn minority or rare classes in large unbalanced datasets with high performance. The main features of the LUPC method are a combination of separate-and-conquer rule induction with association rule mining. The second one is a descriptive method. It is the promising way to find important characteristics from data and describe data explicitly. It also clarifies the relationships between characteristics of data. Therefore, a new descriptive method will be proposed to detect rational design rules for effective siRNAs that mostly have characteristics of previous design rules and contain new characteristics of effective siRNAs. The method will detect descriptive rules by apdating Apriori algorithm with automatic $min\_support$ values after transforming siRNAs to transactions. The detected descriptive rules are filtered, graphically represented and analyzed to generate design rules for effective siRNAs.

2.2 Methods

2.2.1 Learning prediction rules by LUPC

Denote $S = \{(S_1, C_1), (S_2, C_2) \ldots, (S_n, C_n)\}$, where S_i is a sequence of length $|S_i|$ over the alphabet $\Sigma = \{A, C, G, U\}$ or $\Sigma = \{amino \;\; acid\}$ and $C_i \in \{C_1, C_2, ..., C_c\}$ of the class labels. When there are only two classes we call one as positive denoted by Pos and the other as negative denoted by Neg, and thus the labeled set $S = Pos \cup Neg$. Assume we have a set US of sequences S_i without knowing their labels, and assume that $|S|$ is small but $|US| \gg |S|$. The problem is to find a minimal set of rules satisfying two conditions: (1) *Complete*: each sequence is recognized by at least one found rule, (2) *Consistent*: rule found for Pos do not match any negative sequences in Neg and vice versa.

Given parameters α $(0 < \alpha < 1)$ and β $(0 < \beta < 1)$, a subsequence P is an α-*coverage* for Pos if

$$\frac{|cover_{Pos}(P)|}{|Pos|} \geq \alpha,$$

and is a β-*discriminant* for Pos if

$$\frac{|cover_{Pos}(P)|}{|cover_S(P)|} \geq \beta,$$

where $cover_{Pos}(P)$ is the set of sequences in Pos that contains P and $cover_S(P) = cover_{Pos}(P) \cup cover_{Neg}(P)$. In the Pos class, if P is both α-coverage and β-discriminant,

Algorithm LUPC

Input: Labeled sequences in Pos and Neg, and parameters $minalpha$, $minbeta$. **Output**: $\alpha\beta$-strong DMOPS motifs for Pos.

Rule $(Pos, Neg, minalpha, minbeta, \gamma)$
$MotifSet = \phi$
$\alpha, \beta \leftarrow$ **Initialize**$(Pos, minalpha, minbeta)$
while $Pos \neq \phi$ & $(\alpha, \beta) \neq (minalpha, minbeta)$ **do**
  $NewMotif \leftarrow$ **Motif**$(Pos, Neg, \alpha, \beta, \gamma)$
  if $NewMotif \neq \phi$ **then**
    $Pos \leftarrow Pos \setminus Cover^{+}(NewMotif)$
    $MotifSet \leftarrow MotifSet \cup NewMotif$
  else
    Reduce(α, β)
  endif
  $MotifSet \leftarrow$ **PostProcess**$(MotifSet)$
endwhile
return$(MotifSet)$

Figure 2.1: Algorithm for Sequential Learning the Unbalanced Positive Class

P is called $\alpha\beta$-strong for Pos. Similar concepts can be defined for Neg. A subsequence will be a rule when it satisfies both α-*coverage* and β-*discriminant* thresholds.

Note that if sequence P_1 is a subsequence of a sequence P_2, then we have $cover(P_2) \subseteq cover(P_1)$, i.e., the coverage of P_1 is larger and the discrimination ability of P_1 is smaller than those of P_2. Given an α-coverage pattern P, the most informative pattern related to P in terms of coverage is the longest α-coverage pattern containing P. Alternatively, given a β-discriminant pattern P, the most informative pattern related to P in terms of discrimination is the shortest β-discriminant pattern contained in P.

Given two sets of positive sequences Pos and negative sequences Neg, Algorithm LUPC will find a minimal set of prediction rules satisfying condition Complete and Consistent. $Motif(Pos, Neg, \alpha, \beta, \gamma)$ is an exhaustive search procedure that expands a subsequence one position to the left or to the right, starting with length's subsequence is 1.

In procedure finding an $\alpha\beta$-strong motif, the subroutine *Adjacentaa* searches for letters that can be added to $S(i)$ if making $S(i+1)$ satisfies α and β. The subroutine *StopCond* checks if *Adjacentaa* is successful. If 'no', it returns an empty new motif. If 'yes', the subroutine $CandMotifs$ ranks $S(i+1)$ by their number of occurrences in Pos if

Procedure: Finding an $\alpha\beta$-strong motif

Motif $(Pos, Neg, \alpha, \beta, \gamma)$

$CandMotifSet = \phi$
Adjacentaa$(Pos, Neg, \alpha, \beta)$
while StopCond$(Pos, Neg, \alpha, \beta, \gamma)$ **do**
  CandMotifs$(Pos, Neg, \alpha, \beta, \gamma)$
end while
$Motif \leftarrow FirstCandMotifinCandMotifSet$
$return(Motif)$

Figure 2.2: Algorithm for Sequential Learning the Unbalanced Positive Class

there are more than one amino acid that make $S(i+1)$ satisfying both α and β, and $cover(S(i+1)) > \gamma$.

The subroutine $CandMotifs$ may require a lot of checks on Neg to see if a generated motif candidate is $\alpha\beta$-strong. However, thanks to the property "*given a threshold α, a pattern P, many motif candidates are quickly rejected if they are found to match the condition $cover_{Neg}(P) \geq ((1-\alpha)/\alpha) \times cover_{Pos}(P)$ during the scan of Neg*". It is easy to count $cover_{Pos}(P)$ for each motif candidate P as Pos is small, and we need only to accumulate the count of $cover_{Neg}(R)$ when scanning Neg until either we can reject the motif candidate as the constraint holds or we completely go throughout Neg and find the motif has satisfied accuracy.

2.2.2 A descriptive method to detect siRNA design rules

To generate design rules for effective siRNAs, our method is described as following steps (Figure 2.3)

1. Transform siRNA sequences in original dataset to transactions.
2. Apply an adaptive Apriori algorithm with automatic *min_support* values to detect descriptive rules for effective and ineffective siRNAs.
3. Filter descriptive rules and generate design rules for effective siRNAs.

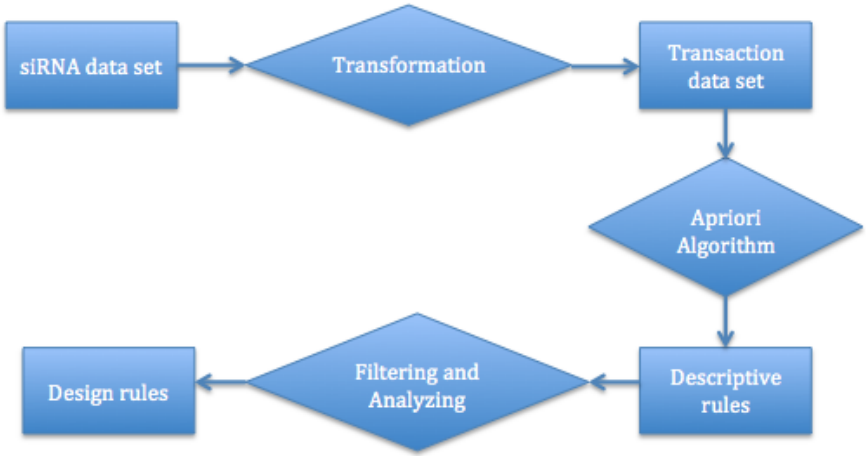


Figure 2.3: Main steps of the descriptive method to detect siRNA design rules.

Transforming siRNA sequences to transactions

Let n denote length of siRNA sequences. In order to transform siRNAs $v_1v_2 \dots v_n$ to a transaction, these siRNA sequence is considered as a set of pairs $\{(1, v_1), (2, v_2), \dots, (n, v_n)\}$ where each pair (p, v) indicates the nucleotide v at position p $(1 \leq p \leq n)$. A function is built to map each pair (p, v) to a positive integer number. Each function value is considered as an item in the transaction dataset. The nucleotides 'A', 'C' , 'G' and 'U' are respectively encoded by 1, 2, 3 and 4 values, respectively. The function is defined as follows

$$f : \{1, \dots, n\}\{1, 2, 3, 4\} \rightarrow \mathbb{N}$$

$$f(p, v) = 4(p - 1) + v$$

Hence, each siRNA sequence $v_1v_2 \dots v_n$ is transformed to a set of items $\{ f(1, v_1), f(2, v_2) , \dots, f(n, v_n) \}$. It is easy to see that the function f is a bijective map with the determination region belonging to $[1, \dots, 4n]$. If function f receives the value x, p and v correspond to $x \ mod \ 4$ and $(x - v) \ div \ 4 + 1$. A k-itemset $(1 \leq k \leq n)$ is a set of items $\{f(p_1, v_1), f(p_2, v_2), \dots, f(p_k, v_k)\}$ on this new feature space. The siRNA design rule detection problem now can be considered as the finding frequent itemsets whose frequencies are not less than $min\_support$ value.

An adaptive Apriori algorithm to detect descriptive rules

In this section, $min\_support$ value is defined to determine $(k + 1)$-frequent itemset joined by two k-frequent itemsets. P denotes a set of transactions having items in k-frequent itemset $\{f(p_1, v_1), f(p_2, v_2), \dots, f(p_{k-1}, v_{k-1}), f(p_k, v_k)\}$ and Q denotes a set of transactions having items in k-frequent itemset $\{f(p_1, v_1), f(p_2, v_2), \dots, f(p_{k-1}, v_{k-1}), f(p'_k, v'_k)\}$.

In the set P, we consider items at the position p'_k on transactions. There are four

Algorithm 1 The Adaptive Apriori algorithm to detect descriptive rules for effective siRNA

Input: siRNA sequences in the C_i class, $i = 1, 2$.
Output: Set S_i of descriptive rules for siRNAs in the C_i class.
for $s = 1 \rightarrow |C_i|$ **do**
  Transform siRNA sequences s to transactions using f function.
end for
k=1;
L_k= {2-itemset}, count frequencies of 2-itemsets.
repeat
  k=k+1
  $S_i = S_i \cup L_k$
  for each pair of k-itemsets in L_k **do**
    Generate $(k+1)$-itemset t.
    Compute $min\_support$ by using equation (2.3)
    if frequence of $t \geq min\_support$ **then**
      $L_{k+1} = L_{k+1} \cup \{t\}$
    end if
  end for
until ((k$\geq$ n) $\vee$ ($L_{k+1} = \emptyset$))

items $f(p'_k, 1)$, $f(p'_k, 2)$, $f(p'_k, 3)$ and $f(p'_k, 4)$ at this position. Thus, at this position, when the set P is divided into four subsets by applying the Dirichlet principle, there is at least one subset A of P such that its cardinality satisfies the following inequation:

$$|A| \geq \lceil \frac{|P|}{4} \rceil + 1 \tag{2.1}$$

The subset A of P that its transactions have $f(p'_k, v'_k)$ item is considered. It is clear that these transactions have $(k+1)$ items $f(p_1, v_1), f(p_2, v_2), \ldots, f(p_k, v_k)$ and $f(p'_k, v'_k)$ and frequency of this $(k+1)$-itemset is cardinality of A. In case the cardinality of A satisfies the above inequation, we call this $(k+1)$-itemset to be frequent.

We analyze the same way for Q set when considering items at the position p_k on transactions. $(k+1)$-itemset is frequent if cardinality of subset A satisfies the following inequation:

$$|A| \geq \lceil \frac{|Q|}{4} \rceil + 1 \tag{2.2}$$

From equations (2.1) and (2.2): $(k+1)$-itemset joined two above k-frequent itemsets is frequent if its frequency satisfies the equation (1) or (2). Thus, frequence of $(k+1)$-itemset satisfies the following inequation:

$$|A| \geq \min\{\lceil \frac{|P|}{4} \rceil + 1, \lceil \frac{|Q|}{4} \rceil + 1\}$$

$$\Leftrightarrow |A| \geq \lceil \frac{\min\{|P|, |Q|\}}{4} \rceil + 1$$

Therefore, *min_support* is defined as the right formulation of inequation.

$$min\_support = \lceil \frac{\min\{|P|, |Q|\}}{4} \rceil + 1 \tag{2.3}$$

The adaptive Apriori algorithm is described in Algorithm 1 where C_1 and C_2 denote effective and ineffective siRNA classes, respectively. S_1 and S_2 denote sets of frequent itemsets in class C_1 and C_2. L_k denotes a set of k-frequent itemsets. Unlike traditional Apriori algorithm, the candidate generation and frequent itemset mining steps are combined into one step. In this step, *min_support* value is computed using equation (2.3) and the $(k+1)$-itemset joined by the two k-frequent itemsets will be checked whether it is frequent or not. It is also applied to detect descriptive rules for ineffective siRNA sequences in the class C_2.

Filtering descriptive rules and generating design rules

The adaptive Apriori algorithm can result many redundant frequent itemsets. It means that there exist rules that generalize these rules. Therefore, redundant rules should be eliminated from S_1 and S_2. On the other hand, we want to detect frequent itemsets in S_1 and S_2 which have high confident. Thus, we find itemsets in the S_1 and S_2 sets such that their confidences equal to 1. The Algorithm 2 shows the filtering descriptive rules in S_1. The descriptive rules in the S_2 set are also filtered by the same way. After filtering descriptive rules in S_1 and S_2, filtered descriptive rules are graphically represented as sequence logos by using the Weblogo tool. On sequence logos, the height of a nucleotide at each position represents its contribution to design rule for siRNAs. Therefore, design rules are generated by choosing nucleotides in decreasing order of their height at each position on sequence logos.

2.3 Experimental evaluation

The first method was applied to detect predictive rules for effective siRNAs by using the 'very high' class. Parameters of the LUPC method are set up: the cover for the 'very high' class is set $\frac{2314}{2655}$ with the accuracy of $\frac{2615}{2956}$. As a result, 114 predictive rules were detected when tested on the 'very high' dataset. To evaluate these rules, we incorporated them into transformation matrices to enrich siRNA representation (*see Chapter 4*) and evaluate the performance of a predictive model.

The second method is applied to generate two design rules for effective siRNAs with 19 nt and 21 nt in length. Our rules are also assessed as previous rules. The experimental

Algorithm 2 Filtering descriptive rules for effective siRNAs

Input: Descriptive rules in S_1, siRNAs in C_1 and C_2.
Output: Filtered descriptive rules.
// eliminate inconfident rules in S_1
for each descriptive rule t in S_1 **do**
  for each siRNA s in C_2 **do**
    if (s contains t) **then**
      $S_1 = S_1 \setminus \{t\}$
    end if
  end for
end for
// eliminate redundant rules in S_1
for each descriptive rule t in S_1 **do**
  for each descriptive rule r in S_1 **do**
    if ($r! = t$) & (r contains t) **then**
      $S_1 = S_1 \setminus \{t\}$
    end if
  end for
end for

evaluation of this method is described as follows

In our experiment, we used two dataset collected from the siRecord database. The first dataset are siRNAs with size of 19 nt that consists of 2470 effective siRNAs labeled 'very high' class and 1261 ineffective siRNAs labeled 'low' class in the siRecord database. The second one contains 1461 effective and 538 ineffective siRNA sequences with 21 nt in length. In our method, *min_support* values are automatically defined, however, it can be decreased to zero. Therefore, low bound of *min_support* was set by 10. The programs was coded in C++ on Dev-cpp environment. The processor speed of computer is 2.52 GHz and the memory is 4 GB.

In process to generate design rule for effective siRNAs with 19 nt in length by using the first dataset, 153 and 5 filtered descriptive rules for effective and ineffective siRNA sequences are detected, respectively. Figure 2.4 and Figure 2.5 show the two sequence logos for two above types of filtered rules. The above two sequence logos are analysed to generate a rational design rule for highly effective siRNAs with 19 nt in length. Our rule shows that effective siRNA sequences with 19 nt in length have the sixteen following characteristics: A 'G/C' and absence of 'A/U' at position 1 (1), An 'A' and absence of 'U' at position 2 (2), an 'A' at position 3 (3), absence of 'A' at position 4 (4), An 'A' and absence of 'C' at position 6(5), an 'A/G' at position 7 (6), a 'C' at position 9 (7), an 'U' at position 10 (8), an 'A/G/U' at position 11 (9), an 'A/C/U' at position 13 (10), an 'A/G' at position 14 (11), an 'A/U' and absence of 'C' at position 15 (12), an 'A/G/U' at position 16 (13), An 'A/U' and absence of 'G/C' at position 17 (14), An 'A/U' and

Figure 2.4: Sequence logo of design rules for effective siRNA with 19 nt in length

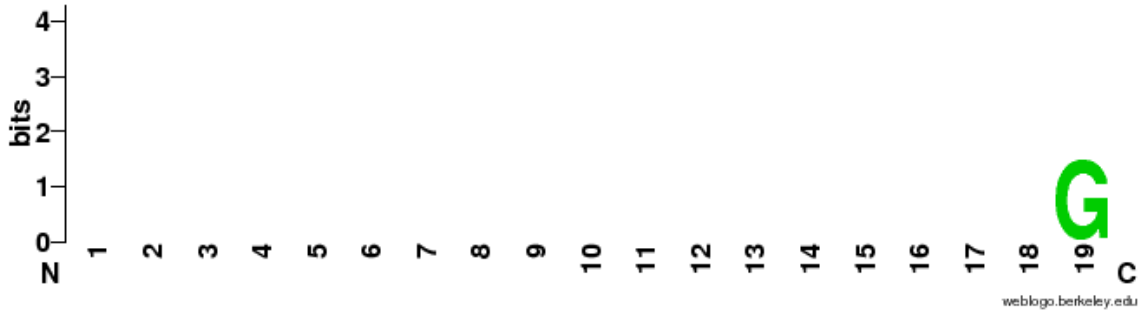


Figure 2.5: Sequence logo of design rules for ineffective siRNA with 19 nt in length

absence of 'G/C' at position 18 (15), An 'A' and absence of 'G' at position 19 (16). This rule is called DR19 and represented on Table 2.1.

When applying our method on the second dataset, 332 filtered descriptive rules for effective siRNAs with 21 nt in length are detected. However, the set of filtered descriptive rule for ineffective siRNAs is the empty set. It may be caused by the imbalance of the dataset. The sequence logo of filtered descriptive rules for effective siRNAs is shown in Figure 2.2. The rational design rule for effective siRNA with 21 nt in length has twenty following characteristics: an 'A/G' at position 1 (1), an 'A' at position 2 (2), a 'G/C' at position 3 (3), an 'A/C' and absence of 'U' at position 4 (4), an 'U/A' and absence of 'C' at position 5 (5), an 'G/C/U' at position 6 (6), absence of 'A' at position 7 (2), an 'A/U' at position 8 (8), a 'G/U' and absence of 'A' at position 9 (9), an 'U' at position 10 (10), an 'U' at position 11 (11), absence of 'U' at position 13 (12), absent 'C' at position 14 (13), absence of 'C' at position 15 (14), an 'A' and absence of 'U' at position 16 (15), an 'A/G' and absence of 'C' at position 17 (16), an 'U' and absence of 'C' at postion 18 (17), an 'A' at position 19 (18), 'A/U' and absence of 'G/C' at position 20 (19), 'A/U' and absence of 'G' at position 21 (20). The rule is called DR21 and represented on Table 2.2.

The DR19 is in a good agreement with previous design rules at some characteristics as follows.

- A 'G/C' at position 1 but nucleotide 'G' is more important than 'C' at this position

	Position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
Reynolds	effective			A							U			A/C/U						A/U
Ui-Tei	effective	G/C																		A/U
	ineffective	A/U																		G/C
Amarzguioui	effective	G/C					A							U						A/U
	ineffective	U									U									G
Jagla	effective	G/C									A/U									A/U
Hsieh	effective											G/C		A			G			U
	ineffective						C					A/U								G
Our rule	effective	G/C	A	A			A	A/G		C	U	A/G/U		A/C/U	A/G	A/U	G/A/U	A/U	U/A	A
	ineffective				A		G													G

Table 2.1: Rational rules for effective siRNA with 19 nt in length

Figure 2.6: Sequence logo of design rules for effective siRNA with 21 nt in length

- An 'A' at position 3 as Reynolds' rule
- An 'A' at postion 6 as Amarzguioui's rule. Absence of 'C' at this position as Hsieh's rule
- An 'U' at position 10 as Reynolds' rule and Jagla's rule
- Absence of 'G' at position 13 as Reynolds' rule
- Absence of 'G' at position 13 as Reynolds' rule
- A 'G' at position 16 as Hsieh's rule
- An 'A' at position 19
- Absence of 'G' at position 19 as Reynolds' rule, Amarzguioui's rule and Hsieh's rule

Interestingly, DR19 contains new characteristics that makes it satisfy other important characteristics of previous rules such as thermodynamic propertities or GC content ranging from 30% to 52% (In our case, GC content ranges from 36% to 52%); at least three 'A/U' at positions from 15 to 19; at least five 'A/U' at positions from 13 to 19 (characterizing for effective silencing, efficience siRNAs entry into RISC and ability of siRNA duplex to unwind); no GC stretch more than 9 nucleotides. In addition, new characteristics in the seed region ranging from position 2 to position 7 may play an important role

	Position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
Huesken	effective	A/G		C								U	A			A/U	A			A/U	A/U	A/U
	ineffective	no G		A								C				C						
Our rule	effective	A/G	A	G/C	C/A, no U	U/A, no C	G	G/C, no A	A/G	G/U, no A	U	U		G, no U	A, no C	U/A, no C	A, no U	G/A, no C	U	A	U/A	A/U
	ineffective																					

Table 2.2: Rational rules for effective siRNA with 21 nt in length

to avoid off-target effects of siRNA that is also one of challenging proplems in RNAi [Pei *et al.*, 2006]. Moreover, characteristics of DR19 in the region (9-11) can make siRNAs recognize and cleave target mRNAs [Reynolds *et al.*, 2004]. Therefore, DR19 not only integrates characteristics of previous design rules but also provides new characteristics for effective siRNAs.

The DR21 is compared to Huesken's motifs which is generated by using neural network. These two rules have the same conclusion at following points:

- An 'A/G' at position 1
- A 'G/C' at position 3
- An 'A' at position 8
- An 'U' at position 11
- An 'A/U' at position 15, no 'C' at this position
- An 'A' at position 16
- An 'A' at position 19
- An 'A/U' at position 20
- An 'A/U' at position 21

The DR21 rule does not give any conclusion at positions 12 as Huesken's rule because in DR21, contributions of different nucleotides at this position are similar to together. Thus, no nucleotide has more significant than the other. Another different point between these two rules is that DR21 rule contains new characteristics in the seed region (4-9) as the DR21 to avoid off-target effects of siRNAs. Moreover, Huesken's rule does also not include characteristics of two nucleotides overhang at the 3' end although these nucleotides can improve effective silencing.

2.4 Conclusion

We first applied the LUPC method to detect siRNA design rules which those accuracies are greater than 90%. The detected predictive rules was used to enrich siRNA representation of data for knockdown efficacy problem that is discussed in Chapter 4. Another method was proposed to detect siRNA design rule that was based on a descriptive approach to find two design rules for effective siRNAs with 19 nt and 21 nt in length. The found design rules not only contain the important characteristics of previous design rules but also have new characteristics to design siRNA effectively. In addition, we also define automatic $min\_support$ values for adatptive Apriori algorithm to detect descriptive rule efficiently.

Chapter 3

Learning methods for siRNA representation enrichment

This chapter presents data representation learning methods. siRNA sequences are represented as enriched matrices by learning transformation matrices that are incorporated background knowledge of siRNA design rules. The two methods are proposed to enrich data representation. The first method is discussed in this chapter. The second one is integrated in the tensor regression model learning so it will be presented in Chapter 4.

3.1 Introduction

In 2006, Fire and Mello received their Nobel Prize for their contributions to research on RNA interference (RNAi) that is the biological process in which RNA molecules inhibit gene expression, typically by causing the destruction of specific mRNA molecules. Their work and that of others on discovery of RNAi have had an immense impact on biomedical research and will most likely lead to novel medical applications. On RNAi research, designing of siRNAs (short interfering RNAs) with high efficacy is one of the most crucial RNAi issues. Highly effective siRNAs can be used to design drugs for viral-mediated diseases such as Influenza A virus, HIV, Hepatitis B virus, RSV viruses, cancer disease and so on. As a result, siRNA silencing is considered one of the most promising techniques in future therapy. Finding highly effective siRNAs among thousands of potential siRNAs for mRNAs remains a great challenge.

Since nearly a decade, machine learning techniques have alternatively been applied to predict knockdown efficacy of siRNAs. The first predictive model was proposed by Huesken *et al.* in which motifs for effective and ineffective siRNA sequences were detected basing on the significance of nucleotides by using a neural network to train 2,182 scoring siRNAs (the high score the higher knockdown efficacy) and test on 249 siRNAs

[Huesken *et al.*, 2005]. This data set was consequently used to build other predictive models [Ichihara *et al.*, 2007], [Shabalina *et al.*, 2006], [Vert *et al.*, 2006]. Recently, Qui *et al.* used multiple support vector regression with RNA string kernel for siRNA efficacy prediction [Qiu *et al.*, 2009], and Sciabola *et al.* applied three dimension structural information of siRNA to increase predictability of the regression model [Sciabola *et al.*, 2013].

However, most of those methods suffer from some drawbacks. Their correlations between predicted values and experimental values of dependent variable ranging from 0.60 to 0.68 were considerably decreased when tested on independent data sets. It can be caused by the fact that the performance of machine learning methods is heavily dependent on the choice of data representation (or features) on which they are applied. It is a reason why much of the actual effort in deploying machine learning algorithms goes into the design and learning of data transformations that result in a representation of the data that can support effective machine learning. In the previous models, siRNAs are encoded by binary, spectral, tetrahedron, and sequence representations. However, these representations are insufficient to represent siRNAs in order to build a good model for predicting siRNA efficacy. The binary representation is dummy one to indicate whether or not a particular nucleotide residues at a position on the siRNA sequence. Therefore, measures based on this representation are unsuitable. The spectral representation shows frequencies of k–mer on the siRNAs so it leads to lack of information to represent data. The tetrahedron representation maps the four nucleotides to the four vertices of regular tetrahedron. This presentation correlates to nucleotide properties such as base pairing, purines and pyrimidines groups. However, mathematical properties on this representation do not exist in base sequence. In addition, Huesken dataset may not be representative of the siRNA population having about 4^{19} siRNAs and the sample size is small. Besides the scoring siRNA dataset, the labeled siRNA datasets, e.g. siRecord database [Ren *et al.*, 2006] with labels such as 'very high", 'high', 'medium', 'low' for the knockdown ability were also exploited by classification methods.

Our work aims to develop data representation learning methods for siRNAs in order to build better prediction models of the siRNA knockdown ability. The key idea is not only focusing on learning algorithms but also exploiting results of the empirical process to enrich the data. In the first method, we first encode siRNAs as binary matrix as traditional representation. To transform siRNA to enriched representation, transformation matrices are designed and learned by incorporating siRNA design rules and using labeled siRNAs in siRecord database. The second siRNA representation learning method is integrated in the the tensor regression model learning to make more precise and accurate representation. Therefore, this method will be presented in detail in chapter 4. The contributions of this work are summarized as follows

3.2 The siRNA representation learning method

The problem of siRNA representation learning method using siRNA design rules is formulated as follows

- **Given:** Two sets of labeled siRNAs of length n, and a set of K siRNA design rules.
- **Find:** Transformation matrices that can transform siRNA sequences to enriched matrices.

The proposed method consists of three steps. The first step is to encode siRNAs as encoding matrices. The second is to design and learn transformation matrices. The final one is to use transformation matrices to enrich siRNAs as enriched matrices and learn model parameters of the bilinear tensor regression to predict the score of siRNAs using transformed matrices. The steps of the method are summarized in Table 3.1.

Table 3.1: siRNA representation learning method

1. To encode each siRNA sequence as an encoding matrix X representing the nucleotides A, C, G, and U at n positions in the sequence. Thus, siRNA sequences are represented as $n \times 4$ encoding matrices.
2. To learn transformation matrices $T_k, k = 1, ..., K$, each characterizes the knockdown ability of nucleotides A, C, G, and U at n positions in the siRNA sequence regarding the kth design rule. Each T_k is learned from the set of labeled siRNAs and the kth design rule. This incorporation of each design rule with siRNAs leads to solve a newly formulated optimization problem.
3. To transform siRNA (encoding matrices) to enriched matrices by K transformation matrices.

Step 1 of the method can be easily done where each siRNA sequence with n nucleotides in length is encoded as a binary encoding matrix of size $n \times 4$. In fact, four nucleotides A, C, G, or U are encoded by encoding vectors (1,0,0,0), (0,1,0,0), (0,0,1,0) and (0,0,0,1), respectively. If a nucleotide from A, C, G, and U appears at the jth position in a siRNA sequence, $j = 1, ..., n$, its encoding vector will be used to encode the jth row of the encoding matrix.

Step 2 is to learn transformation matrices T_k regarding the kth design rule, $k = 1, ..., K$. T_k has size of $4 \times n$ where the rows correspond to nucleotides A, C, G, and U and the columns correspond to n positions on sequences. T_k are learned one by one from the set of siRNAs and the kth design rule, thus we use T instead of T_k for simplification. Each cell $T[i, j], i = 1, ..., 4, j = 1, ..., n$, represents the knockdown ability of nucleotide i at position j regarding the kth design rule. Each cell $T[i, j]$ to be learned have to satisfy

Sequence	Encoding matrix X	Transformation matrix T	Transformed data vector $x = T \circ X$
AUGCU	1 0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 0 0 0 1	$\overline{0.5}$ 0.7 0.32 0.2 0.5 0.3 0.1 0.6 $\overline{0.6}$ 0.3 0.1 0.1 $\overline{0.08}$ 0.1 0.1 0.1 $\overline{0.1}$ 0 0.1 $\overline{0.1}$	(0.5, 0.1, 0.08, 0.6, 0.1)

Position	Knockdown ability	Nucleotides	Mapping to T	Constraints on T
19	Effective	A,U	$T[1,19]$, $T[4,19]$	$T[3,19]-T[1,19]<0$ $T[3,19]-T[4,19]<0$
	Ineffective	C	$T[2,19]$	$T[2,19]-T[1,19]<0$ $T[2,19]-T[3,19]<0$ $T[2,19]-T[4,19]<0$

Figure 3.1: The left table shows an example of encoding matrix, transformation matrix, and transformed vector (the values $\overline{0.5}, \overline{0.1}$ etc. are taken to the transformed vector). The right table is an example of incorporating the condition of a design rule at position 19 to a transformation matrix T by designing constraints.

a number of constraints. First, they are basic and normalization constraints on elements of T

$$T[i,j] \geq 0, \qquad i = 1, ..., 4; \quad j = 1, 2, \ldots, n \tag{3.1}$$

$$\sum_{i=1}^{4} T[i,j] = 1, \quad j = 1, \ldots, n \tag{3.2}$$

The second kind of constraints related to design rules. Each design rule propositionally describes the occurrence or absence of nucleotides at different positions of effective siRNA sequences. Therefore, if a design rule shows the occurrence (absence) of some nucleotides on jth position, then their corresponding values in the matrix T would be greater (smaller) than other values at column j. For example, the design rule in the right table in Figure 3.1 illustrates that at position 19, nucleotides A/U are effective and nucleotide C is ineffective. It means that knockdown ability of nucleotides A/U are bigger than that of nucleotides G/C and knockdown ability of nucleotide C is smaller than that of the other nucleotides. Thus, values $T[1,19], T[2,19], T[3,19]$ and $T[4,19]$ show the knockdown ability of nucleotides A, C, G and U at position 19, respectively. Therefore, five constraints at column 19 of T are formed. Generally, we denote the set of R trick inequality constraints on T by the design rule under consideration by

$$\{g_r(T) < 0\}_{r=1}^{R} \tag{3.3}$$

The third kind of constraints relating to preservation of the siRNA classes after being transformed by using transformation matrices T_k, it means that siRNAs belonging to the same class should be more similar to each other than siRNAs belonging to the other class.

Let vector x_l of size $1 \times n$ denote the transformed vector of the lth siRNA sequence using the transformation matrix T. The jth element of x_l is the element of T at column j and the row corresponds to the jth nucleotide in the siRNA sequence. To compute x_l, new column-wise inner product is defined as follows

$$x_l = T \circ X_l = (\langle X_l[1,.], T[.,1]\rangle, \langle X_l[2,.], T[.,2]\rangle, \ldots, \langle X_l[n,.], T[.,n]\rangle) \tag{3.4}$$

where $X_l[j,.]$ and $T[.,j]$ are the jth row vector and the jth column of the matrix X_l and T, respectively, and $\langle x, y\rangle$ denotes the inner product of vectors x and y.

The left table in Figure 3.1 shows an example of encoding matrix X, transformation matrix T and transformed vector x of the given sequence AUGCU. The rows of X represent encoding vectors of nucleotides in the sequence. Given transformation matrix T of size 4×5. The AUGCU sequence is represented by the vector

$x = (T[1,1],\ T[4,1],\ T[3,3],\ T[2,4],\ T[4,5]) = (0.5, 0.1, 0.08, 0.6, 0.1)$

Therefore, transformed data can be computed by the column-wise inner product $x = T \circ X$.

The problem of transformation matrix learning is now formulated as finding T under constraints (1), (2) and (3) so that the similarity of transformed vectors x_l in the same class is minimum and the dissimilarity of x_l in different classes is maximum. The learning problem then leads to solve the optimization problem with the following objective function

$$Min \sum_{p,q\in N_1} d^2(x_p, x_q) + \sum_{p,q\in N_2} d^2(x_p, x_q) - \sum_{\substack{p\in N_1 \\ q\in N_2}} d^2(x_p, x_q) \tag{3.5}$$

$$\text{Subject to } \ T[i,j] \geq 0,\ \sum_{i=1}^{4} T[i,j] = 1, g_r(T) < 0,\ i = 1, ..., 4; j = 1, ..., n; r = 1, .., R.$$

In the objective function, the two first components are the sum of similarity of sequence pairs belonging to the same class and the last one is similarity of sequence pairs belonging to two different classes; $d(x, y)$ is the similarity measure between x and y (in this work we use Euclidean distance and L_2 norm); N_1 and N_2 are the two index sets of high and low efficacy siRNAs, respectively. Constraints $g_i(T)$ can also help to avoid the trivial solution of the objective function.

This optimization problem is solved by the following Lagrangian form

$$\begin{aligned}
E &= \sum_{p,q\in N_1} d^2(x_p, x_q) + \sum_{p,q\in N_2} d^2(x_p, x_q) - \sum_{\substack{p\in N_1 \\ q\in N_2}} d^2(x_p, x_q) + \sum_{j=1}^{n} \lambda_j \left(\sum_{i=1}^{4} T[i,j] - 1 \right) + \sum_{r=1}^{R} \mu_r g_r(T) \\
&= \sum_{\substack{p\in N_1 \\ q\in N_1}} \| x_p - x_q \|_2^2 + \sum_{\substack{p\in N_2 \\ q\in N_2}} \| x_p - x_q \|_2^2 - \sum_{\substack{p\in N_1 \\ q\in N_2}} \| x_p - x_q \|_2^2 + \sum_{j=1}^{n} \lambda_j \left(\sum_{i=1}^{4} T[i,j] - 1 \right) + \sum_{r=1}^{R} \mu_r g_r(T) \\
&= \sum_{p,q\in N_1} \sum_{j=1}^{n} (\langle X_p[j,.], T[.,j]\rangle - \langle X_q[j,.], T[.,j]\rangle)^2 + \sum_{p,q\in N_2} \sum_{j=1}^{n} (\langle X_p[j,.], T[.,j]\rangle - \langle X_q[j,.], T[.,j]\rangle)^2 \\
&\quad + \sum_{j=1}^{n} \lambda_j \left(\sum_{i=1}^{4} T[i,j] - 1 \right) + \sum_{r=1}^{R} \mu_r g_r(T) - \sum_{\substack{p\in N_1 \\ q\in N_2}} \sum_{j=1}^{n} (\langle X_p[j,.], T[.,j]\rangle - \langle X_q[j,.], T[.,j]\rangle)^2
\end{aligned}$$

where $\mu_r, r = 1, ..., R$ and $\lambda_j, j = 1, \ldots, n$ are Lagrangian multipliers. To solve the minimization problem, an iterative method is applied. For each pair of (i, j), $T[i, j]$ is solved while keeping the other elements of T. The Karush-Kuhn-Tucker conditions are

- Stationarity: $\frac{\partial E}{\partial T[i,j]} = 0, i = 1, ..., 4$ and $j = 1, \ldots, n$.
- Primal feasibility: $T[i, j] \geq 0, \sum_{i=1}^{4} T[i, j] = 1, g_r(T) < 0, i = 1, ..., 4;$ $j = 1, ..., n; r = 1, ..., R.$
- Dual feasibility: $\mu_r \geq 0, r = 1, \ldots, R.$
- Complementary slackness: $\mu_r g_r(T) = 0, r = 1, \ldots, R.$

From the last three conditions, we have $\mu_r = 0, r = 1, \ldots, R$. Therefore, the stationarity condition can be derived as follows

$$\begin{aligned}
\frac{\partial E}{\partial T[i,j]} &= 2 \sum_{p,q \in N_1} (\langle X_p[j,.], T[.,j] \rangle - \langle X_q[j,.], T[.,j] \rangle)(X_p[j,i] - X_q[j,i]) \\
&\quad +2 \sum_{p,q \in N_2} (\langle X_p[j,.], T[.,j] \rangle - \langle X_q[j,.], T[.,j] \rangle)(X_p[j,i] - X_q[j,i]) \\
&\quad -2 \sum_{p \in N_1, q \in N_2} (\langle X_p[j,.], T[.,j] \rangle - \langle X_q[j,.], T[.,j] \rangle)(X_p[j,i] - X_q[j,i]) + \lambda_j = 0
\end{aligned}$$

Set $Z_{p,q} = (X_p - X_q)^T$ and A_{ij} is the vector resulting from the column j of matrix A by removing the element $A[i, j]$. Therefore, the above formulation is derived as follows

$$\begin{aligned}
\frac{\partial E}{\partial T[i,j]} &= 2(\sum_{p,q \in N_1} \langle (Z_{p,q})_{ij}, T_{ij} \rangle Z_{p,q}[i,j] + \sum_{p,q \in N_2} \langle (Z_{p,q})_{ij}, T_{ij} \rangle Z_{p,q}[i,j] \\
&\quad - \sum_{p \in N_1, q \in N_2} \langle (Z_{p,q})_{ij}, T_{ij} \rangle Z_{p,q}[i,j]) \\
&\quad +2T[i,j] \left(\sum_{p,q \in N_1} Z^2_{p,q}[i,j] + \sum_{p,q \in N_2} Z^2_{p,q}[i,j] - \sum_{p \in N_1, q \in N_2} Z^2_{p,q}[i,j] \right) + \lambda_j = 0
\end{aligned}$$

We define the following equations

$$S(i,j) = \sum_{p,q \in N_1} Z^2_{p,q}[i,j] + \sum_{p,q \in N_2} Z^2_{p,q}[i,j] - \sum_{p \in N_1, q \in N_2} Z^2_{p,q}[i,j] \tag{3.6}$$

$$\begin{aligned}
B(i,j) &= \sum_{p,q \in N_1} \langle (Z_{p,q})_{ij}, T_{ij} \rangle Z_{p,q}[i,j] + \sum_{p,q \in N_2} \langle (Z_{p,q})_{ij}, T_{ij} \rangle Z_{p,q}[i,j] \\
&\quad - \sum_{p \in N_1, q \in N_2} \langle (Z_{p,q})_{ij}, T_{ij} \rangle Z_{p,q}[i,j].
\end{aligned} \tag{3.7}$$

Substitute (3.6) and (3.7) to $\frac{\partial E}{\partial T[i,j]}$, we have

$$T[i,j] = \frac{\frac{-\lambda_j}{2} - B(i,j)}{S(i,j)} \tag{3.8}$$

At a column j, T has to satisfy

$$\sum_{i_1=1}^{4} T(i_1,j) = 1 \quad \Leftrightarrow \sum_{i_1=1}^{4} \frac{\frac{-\lambda_j}{2} - B(i_1,j)}{S(i_1,j)} = 1 \Rightarrow \frac{-\lambda_j}{2} = \frac{1 + \sum_{i_1=1}^{4} \frac{B(i_1,j)}{S(i_1,j)}}{\sum_{i_1=1}^{4} \frac{1}{S(i_1,j)}} \tag{3.9}$$

Substitute (3.9) to (3.8), equation (3.8) can be derived as

$$T[i,j] \quad = \frac{\frac{1+\sum_{i_1=1}^{4} \frac{B(i_1,j)}{S(i_1,j)}}{\sum_{i_1=1}^{4} \frac{1}{S(i_1,j)}} - B(i,j)}{S(i,j)} = \frac{1+\sum_{i_1 \neq i} \frac{B(i_1,j)-B(i,j)}{S(i_1,j)}}{\sum_{i_1=1}^{4} \frac{S(i,j)}{S(i_1,j)}} \tag{3.10}$$

In this task, K design rules are used to learn K transformation matrices. The main steps are summarized in Algorithm 1. For each siRNA design rule, the algorithm will update each element of the transformation matrix according to equation (3.10). In each iterative step, the transformation matrix without trick inequality constraints is updated to reach the global optimal solution. If updated elements in a column satisfy the trick inequality constraints characterizing the condition at the corresponding position of the rule, that column will be updated to the target solution. The transformation matrix is updated until meeting the convergence criteria. $\| \ . \ \|_{Fro}$ is the Frobenious norm of a matrix.

The time complexity of the algorithm is $\mathrm{O}\big(KnN^2 t_{Max}\big)$ where N is the maximum number of N_1 and N_2. In particularly, the K transformation matrices are learned and each transformation matrices have $4 \times n$ elements. For each element of the kth transformation matrix has to compute equation (3.6) and (3.7). Therefore, the complexity of each element is $\mathrm{O}\big(N^2\big)$. It leads to the time complexity of the kth transformation matrix with size of $4 \times n$ in t_{Max} iterative steps is $\mathrm{O}\big(nN^2 t_{Max}\big)$. As a result, the time complexity to learn K transformation matrices is $\mathrm{O}\big(KnN^2 t_{Max}\big)$. When N_1 or N_2 is a large number, the learning method is time consuming. Therefore, to decrease the time complexity of the learning method and also describe the meaning of components in the formulation (3.8), we define the following sigma function and proposed the two following corollaries.

sigma function definition: Given a set of integer numbers $X = \{1,2,3,4\}$ a set of nucleotides $Y = \{A,C,G,U\}$. Sigma function σ is defined is a map from X to Y as follows

$$\sigma : \ X \to Y$$

Algorithm 3 Transformation matrices learning

Input: A data set $S = \{(s_l, y_l)\}_1^N$ where s_l are siRNA sequences and y_l are their labels, a set DR of K design rules, the length n of siRNA sequences.
Output: K transformation matrices $T_1, T_2, \ldots, T_K$.
Encoding siRNA sequences in S.
for $rule_k$ in DR **do**
  Form the set of constraints C_k based on $rule_k$
  Initialize the transformation matrix T_k satisfying C_k.
  $t = 0$ { Iterative step}
  repeat
    $t \leftarrow t + 1$
    for $j = 1$ to n **do**
      $v = T_k^{(t-1)}[., j]$ { A temporary vector}
      for $i = 1$ to 4 **do**
        Compute $v[i]$ using equation (3.10)
      end for
      if (v satisfies the constraints at the position j in C_k) **then**
        $T_k^{(t)}[., j] \leftarrow v$
      end if
    end for
  until ($\frac{\|T_k^{(t)} - T_k^{(t-1)}\|_{Fro}}{\|T_k^{(t-1)}\|_{Fro}} \leq \epsilon$) or ($t > t_{Max}$)
end for

$$\sigma(i) = \begin{cases} A & \text{if } i = 1 \\ C & \text{if } i = 2 \\ G & \text{if } i = 3 \\ U & \text{if } i = 4 \end{cases}$$

Corollary 1: *Let $F_1(Nu, j)$ and $F_2(Nu, j)$ denote the frequencies of nucleotide Nu at position j in siRNA sequences in the effective and ineffective class, respectively. Let N_1 and N_2 denote the number of siRNAs in the effective and ineffective class, respectively. Equation* 3.6 *is equivalent to the following equation.*

$$S(i,j) = \big(N_1 - F_1(\sigma(i), j)\big)\big(2F_1(\sigma(i), j) - F_2(\sigma(i), j)\big) + \big(N_2 - F_2(\sigma(i), j)\big)\big(2F_2(\sigma(i), j) - F_1(\sigma(i), j)\big)$$

Proof: We know that $X_l[j, i]$ is the element at the jth row and the i column of the encoding matrix X_l corresponding to the siRNA sequence s_l. $X_l[j, i]$ equals to 1 if only if the siRNA sequence s_l has nucleotide $\sigma(i)$ at position j.

For each pair (i, j) $i = 1, \ldots, 4$ and $j = 1, \ldots, n$ and pair of two siRNAs (s_p, s_q), we have $Z_{p,q}[i, j] = X_p[j, i] - X_q[j, i]$. Because $X_p[j, i], X_q[j, i] \in \{0, 1\}$ so $Z_{p,q}[i, j]$ belongs to $\{-1, 0, 1\}$. We also see that

$$Z_{p,q}[i,j] = \begin{cases} -1 & \text{if } X_p[j,i] = 0 \text{ and } X_q[j,i] = 1 \\ 0 & \text{if } (X_p[j,i] = 0 \text{ and } X_q[j,i] = 0) \text{ or } (X_p[j,i] = 1 \text{ and } X_q[j,i] = 1) \\ 1 & \text{if } X_p[j,i] = 1 \text{ and } X_q[j,i] = 0 \end{cases}$$

Therefore, $Z^2_{p,q}[i,j] = 1$ means that the pair of siRNA sequences (s_p, s_q) such that the nucleotide $\sigma(i)$ residues at the position j in the sequence s_p ($X_p[j,i] = 1$) and a nucleotide except the nucleotide $\sigma(i)$ residues at the position j in the sequence s_q ($X_q[j,i] = 0$), vise versa. Let $Cond1$ denote this condition.

Now, we consider to the first component of equation (3.6), $\sum_{p,q\in N_1} Z^2_{p,q}[i,j]$ is the number of pairs (s_p, s_q) in the effective class satisfying the condition $Cond1$. In the effective class, $F_1(\sigma(i), j)$ siRNAs contain the nucleotide $\sigma(i)$ at position j and $N_1 - F_1(\sigma(i), j)$ siRNAs that have a nucleotide except the nucleotide $\sigma(i)$ at position j. Therefore, we have the following equation

$$\sum_{p,q\in N_1} Z^2_{p,q}[i,j] = 2F_1(\sigma(i), j)\big(N_1 - F_1(\sigma(i), j)\big) \tag{3.11}$$

Similar arguments, we have following equations

$$\sum_{p,q\in N_2} Z^2_{p,q}[i,j] = 2F_2(\sigma(i), j)\big(N_2 - F_2(\sigma(i), j)\big) \tag{3.12}$$

$$\sum_{p\in N_1, q\in N_2} Z^2_{p,q}[i,j] = F_1(\sigma(i), j)\big(N_2 - F_2(\sigma(i), j)\big) + F_2(\sigma(i), j)\big(N_1 - F_1(\sigma(i), j)\big) \tag{3.13}$$

From (3.11),(3.12) and (3.13) equations, equation (3.6) can be derived as follows

$$\begin{aligned} S(i,j) &= \sum_{p,q\in N_1} Z^2_{p,q}[i,j] + \sum_{p,q\in N_2} Z^2_{p,q}[i,j] - \sum_{p\in N_1, q\in N_2} Z^2_{p,q}[i,j] \\ &= 2F_1(\sigma(i), j)\big(N_1 - F_1(\sigma(i), j)\big) + 2F_2(\sigma(i), j)\big(N_2 - F_2(\sigma(i), j)\big) \\ &\quad -F_1(\sigma(i), j)\big(N_2 - F_2(\sigma(i), j)\big) - F_2(\sigma(i), j)\big(N_1 - F_1(\sigma(i), j)\big) \\ &= \big(N_1 - F_1(\sigma(i), j)\big)\big(2F_1(\sigma(i), j) - F_2(\sigma(i), j)\big) + \big(N_2 - F_2(\sigma(i), j)\big)\big(2F_2(\sigma(i), j) - F_1(\sigma(i), j)\big) \end{aligned}$$

■

The equation in **Corollary 1** describes the relationship between expectations of the nucleotide $\sigma(i)$ at the jth position in the two classes in the original dataset. It means that we should consider the expectation of nucleotides when we want to design these nucleotides at a particular position on the sequence that can effect to knockdown efficacy.

Corollary 2: *Let* $E_1(j) = \frac{1}{N_1}\sum_{i=1}^{4} F_1\big(\sigma(i), j\big)T[i,j]$, *and* $E_2(j) = \frac{1}{N_2}\sum_{i=1}^{4} F_2\big(\sigma(i), j\big)T[i,j]$ *denote the expectations of the random variable at position* j *in the enriched vector using*

transformation matrix T ($T \circ X$) in the effective and ineffective classes. Equation (3.7) *is equivalent to the following equation.*

$$\begin{aligned} B(i,j) =& \Big(N_1E_1 - T[i][j]F_1\big(\sigma(i),j\big)\Big)\Big(F_2(\sigma(i),j) - 2F_1(\sigma(i),j)\Big) \\ &+ \Big(N_2E_2 - T[i][j]F_2\big(\sigma(i),j\big)\Big)\Big(F_1(\sigma(i),j) - 2F_2(\sigma(i),j)\Big) \end{aligned}$$

Proof:

Considering the first component of equation (3.7): $\sum_{p,q\in N_1}\langle (Z_{p,q})_{ij}, T_{ij}\rangle Z_{p,q}[i,j]$. It is inferred that $Z_{p,q}[i,j] = 1$ or $Z_{p,q}[i,j] = -1$.

In case $Z_{p,q}[i,j] = 1$, we have: $X_p[j,i] = 1$ and $X_q[j,i] = 0 \Leftrightarrow (X_p)_{ij} = (0,0,0)$ and $(X_q)_{ij}$ receives one of the three vectors: (1,0,0); (0,1,0); (0,0,1). As a result, $(Z_{p,q})_{ij}$ receives one of the three vectors: (-1,0,0); (0,-1,0); (0,0,-1). Therefore, existing $i_1 \neq i$ such that $X_q[j,i_1] = 1$ and $\langle (Z_{p,q})_{ij}, T_{ij}\rangle Z_{p,q}[i,j] = -T[i_1,j]$. For each siRNA s_p having the nucleotide $\sigma(i)$ at the jth position (i.e $X_p[j,i] = 1$), we have

$$\sum_{q\in N_1}\langle (Z_{p,q})_{ij}, T_{ij}\rangle Z_{p,q}[i,j] = -\sum_{\substack{i_1\neq i \\ i_1\in N_1}} F_1(\sigma(i_1),j) * T[i_1,j] = N_1 * E_1 - F_1(\sigma(i),j) * T[i,j]$$

In fact, we have $F_1(\sigma(i),j)$ siRNAs having the nucleotide $\sigma(i)$ at the jth position, therefore we have the following equation

$$\sum_{p,q\in N_1}\langle (Z_{p,q})_{ij}, T_{ij}\rangle Z_{p,q}[i,j] = -F_1(\sigma(i),j)\sum_{\substack{i_1\neq i \\ i_1\in N_1}} F_1(\sigma(i_1),j) * T[i_1,j] = -F_1(\sigma(i),j)\Big(N_1 * E_1 - F_1(\sigma(i),j) * T[i,j]\Big)$$

In case $Z_{p,q}[i,j] = -1$: by similar arguments, we also have:

$$\sum_{p,q\in N_1}\langle (Z_{p,q})_{ij}, T_{ij}\rangle Z_{p,q}[i,j] = -F_1(\sigma(i),j)\sum_{\substack{i_1\neq i \\ i_1\in N_1}} F_1(\sigma(i_1),j) * T[i_1,j] = -F_1(\sigma(i),j)\Big(N_1 * E_1 - F_1(\sigma(i),j) * T[i,j]\Big)$$

Therefore, the following equation is derived for the both cases:

$$\sum_{p,q\in N_1}\langle (Z_{p,q})_{ij}, T_{ij}\rangle Z_{p,q}[i,j] = -2F_1(\sigma(i),j)\sum_{\substack{i_1\neq i \\ i_1\in N_1}} F_1(\sigma(i_1),j) * T[i_1,j] = -2F_1(\sigma(i),j)\Big(N_1 * E_1 - F_1(\sigma(i),j) * T[i,j]\Big) \tag{3.14}$$

The second component of equation (3.7): $\sum_{p,q\in N_2}\langle (Z_{p,q})_{ij}, T_{ij}\rangle Z_{p,q}[i,j]$ can be derived as follows

$$\sum_{p,q\in N_2}\langle (Z_{p,q})_{ij}, T_{ij}\rangle Z_{p,q}[i,j] = -2F_2(\sigma(i),j)\sum_{\substack{i_1\neq i \\ i_1\in N_2}} F_2(\sigma(i_1),j) * T[i_1,j] = -2F_2(\sigma(i),j)\Big(N_2 * E_2 - F_2(\sigma(i),j) * T[i,j]\Big) \tag{3.15}$$

The third component of equation (3.7): $\sum_{p\in N_1, q\in N_2}\langle (Z_{p,q})_{ij}, T_{ij}\rangle Z_{p,q}[i,j]$ can be derived as follows

$$\begin{aligned}\sum_{\substack{p\in N_1\\ q\in N_2}}\langle (Z_{p,q})_{ij}, T_{ij}\rangle Z_{p,q}[i,j] &= -F_1(\sigma(i),j)\sum_{\substack{i_1\neq i\\ i_1\in N_2}} F_2(\sigma(i_1),j)*T[i_1,j] - F_2(\sigma(i),j)\sum_{\substack{i_1\neq i\\ i_1\in N_1}} F_1(\sigma(i_1),j)*T[i_1,j]\\ &= -F_1(\sigma(i),j)\Big(N_2*E_2 - F_2(\sigma(i),j)*T[i,j]\Big) - F_2(\sigma(i),j)\Big(N_1*E_1 - F_1(\sigma(i),j)*T[i,j]\Big) \quad (3.16)\end{aligned}$$

From (3.14),(3.15), and (3.16) equations, equation (3.7) can be derived as follows

$$\begin{aligned}B(i,j) &= \sum_{p,q\in N_1}\langle (Z_{p,q})_{ij}, T_{ij}\rangle Z_{p,q}[i,j] + \sum_{p,q\in N_2}\langle (Z_{p,q})_{ij}, T_{ij}\rangle Z_{p,q}[i,j] - \sum_{p\in N_1, q\in N_2}\langle (Z_{p,q})_{ij}, T_{ij}\rangle Z_{p,q}[i,j]\\ &= -2F_1(\sigma(i),j)\Big(N_1*E_1 - F_1(\sigma(i),j)*T[i,j]\Big) - 2F_2(\sigma(i),j)\Big(N_2*E_2 - F_2(\sigma(i),j)*T[i,j]\Big)\\ &\quad +F_1(\sigma(i),j)\Big(N_2*E_2 - F_2(\sigma(i),j)*T[i,j]\Big) + F_2(\sigma(i),j)\Big(N_1*E_1 - F_1(\sigma(i),j)*T[i,j]\Big)\\ &= \Big(N_1E_1 - T[i][j]F_1\big(\sigma(i),j\big)\Big)\Big(F_2(\sigma(i),j) - 2F_1(\sigma(i),j)\Big)\\ &\quad +\Big(N_2E_2 - T[i][j]F_2\big(\sigma(i),j\big)\Big)\Big(F_1(\sigma(i),j) - 2F_2(\sigma(i),j)\Big)\end{aligned}$$

■

The equation in **Corollary 2** also describes the relationship between expectations of the enriched value corresponding to $\sigma(i)$ nucleotide at the jth position in the two classes after transforming to new feature space. The **Corollary 1** and **Corollary 2** have proofs that explains the derivation of formulations (3.6) and (3.7), respectively. We see that equations in **Corollary 1** and **Corollary 2** to compute $S(i,j)$ and $B(i,j)$ depend on frequencies of nucleotides at position in the sequence. These frequencies can be computed one time with time complexity of $\mathrm{O}\big(n(N_1+N_2)\big)$. After that, time complexity to compute $S(i,j)$ and $B(i,j)$ is constant in algorithm (3). Therefore, time complexity of $\mathrm{O}\big(KnN^2t_{Max}\big)$ reduces to $\mathrm{O}\big(n(N_1+N_2)+Knt_{Max}\big)$.

The third step of the method can be easily done where encoding matrices X of siRNAs are transformed to enriched matrices $(T_1\circ X, T_2\circ X, \ldots, T_1\circ X)^T$ by using K transformation matrices $T_1,\ T_2,\ \ldots,\ T_K$.

3.3 Experiment

In this representation learning method, we use the labeled dataset collected from siRecord database [Ren *et al.*, 2006]. This data set has 2470 siRNA sequences in 'very high' class and 2514 siRNA sequences in 'low' and 'medium' classes. Each siRNA sequence has 19 nucleotides. Seven design rules used to enrich representation of siRNAs are Reynolds rule, Uitei rule, Amarzguioui rule, Jalag rule, Hsieh rule, Takasaki rule and Huesken rule [Reynolds *et al.*, 2004, Uitei *et al.*, 2004, Amarzguioui *et al.*, 2004, Hsieh *et al.*, 2004,

0	0	0	0	0	0	0	0	0	1	0	0	0.33	0	0	0	0	0	0.5
0	0	0	0	0	0	0	0	0	0	0	0	0.33	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0.01	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0	0.33	0	0	0	0	0	0.5

Figure 3.2: The constraint matrix of the Reynold rule.

Jagla *et al.*, 2005, Takasaki *et al.*, 2010, Huesken *et al.*, 2005]. Constraints based on each design rule was created as a matrix of size 4×19. This matrix satisfies the following conditions.

- if the rule does not mention the designing of nucleotides at the jth position, values in the j row of the matrix are 0.
- if a nucleotide $\sigma(i)$ at the jth position is stated as ineffective in the rule, the cell (i, j) receives the lowest value in the jth row.
- if a nucleotide $\sigma(i)$ at the jth position is stated as effective in the rule and the others are not mentioned, the cell (i, j) receives the highest value in the jth row.
- if a nucleotide $\sigma(i)$ at the jth position is not stated in the rule, the cell (i, j) receives a value that is larger than the lowest value and less than the largest value.

For example, the Reynolds rule [Reynolds *et al.*, 2004] consists of the following positional characteristics

- An 'A' at position 19
- An 'U' at position 3
- An 'A' at position 10
- A base other than 'G' or 'C' at position 19
- A base other than 'G' at position 13

The constraint matrix based on this rule was created as figure 3.2. At position 19, the rule states that "An 'A' at position 19" and "A base other than 'G' or 'C' at position 19", therefore at this position, values in the constraint matrix corresponding to nucleotides 'A' and 'U' are the highest value and values corresponding to 'G' and 'C' are 0 (not mentioned in the rule).

The convergence criteria in Algorithm 3 are set up as following: threshold ϵ for transformation matrices is $2.5E^{-8}$ and the maximum iterative step is 5000.

The confidence of enriched representation based on transformation matrices is discussed in more detail in Chapter 4. In Chapter 4, we also introduce the second method for learning siRNA representation.

3.4 Conclusion

The data representation learning problem is one of the most crucial issues in the machine learning research. To overcome drawback of previous research on the prediction of siRNA knockdown efficacy, we have proposed data representation learning methods in order to enrich siRNAs. In the first method, siRNAs in the original space are transformed to enriched matrices by transformation matrices that are learned by integrating siRNA design rules. We also proposed two corollaries to decrease the time complexity of the method. The second method is presented in Chapter 4.

Chapter 4

Tensor regression methods for siRNA efficacy prediction

This chapter presents two tensor regression methods. To transform siRNAs to enriched matrices, the first method firstly used the siRNA representation algorithm in chapter 3 to learn transformation matrices. In this method, the two sets of siRNA design rules that were used to enrich siRNA representations are siRNA design rules detected by the LUPC method and empirical analysis, respectively. The second method used a labeled siRNA dataset to supervise the learning process of parameters of tensor regression model. In this method, transformation matrices and parameters of model learning were learned simultaneously to make more accurate and precise data representation. The final part of this chapter presents experimental evaluations of the proposed methods.

4.1 Introduction

RNA interference (RNAi) is a cellular process in which RNA molecules inhibit gene expressions, typically by causing the destruction of mRNA molecules. Long double stranded RNA duplex or hairpin precursors are cleaved into short interfering RNAs (siRNAs) by the ribonuclease III enzyme Dicer. The siRNAs are sequences of 19–23 nucleotides (nt) in length with 2 nt overhangs at the $3'$ ends. Guided by RNA induced silencing complex (RISC), siRNAs bind to their complementary target mRNAs and induce their degradation.

In 2006, Fire and Mello received the Nobel Prize for their contributions to research on RNA interference. Their work and those of others on discovery of RNAi have had an immense impact on biomedical research and will most likely lead to novel medical applications [Elbashir *et al.*, 2001, Elbashir *et al.*, 2002, Harborth *et al.*, 2003, Braasch *et al.*, 2003, Chiu *et al.*, 2003, Warnecke *et al.*, 2004]. In RNAi research, designing of siRNAs with

high efficacy is one of the most crucial RNAi issues. Highly effective siRNAs can be used to design drugs for viral-mediated diseases such as influenza A virus, HIV, hepatitis B virus, RSV viruses, cancer disease and so on. As a result, siRNA silencing is considered one of the most promising techniques in future therapy. However, finding highly effective siRNAs from a huge amount of potential siRNAs remains a great challenge.

Various siRNA design rules have been found by empirical processes since 1998. The first rational siRNA design rule was detected by [Elbashir *et al.*, 2001]. They suggested that siRNAs having 19–21 nt in length with 2 nt overhangs at the $3'$ ends can efficiently silence mRNAs. [Scherer *et al.*, 2003] reported that the thermodynamic properties to target specific mRNAs are important characteristics . Soon after these works, many rational design rules for effective siRNAs have been found, typically those in [Reynolds *et al.*, 2004], [Uitei *et al.*, 2004], [Amarzguioui *et al.*, 2004], [Hsieh *et al.*, 2004], [Jagla *et al.*, 2005], [Pei *et al.*, 2006]. For example, Reynolds *et al.* [Reynolds *et al.*, 2004] analyzed 180 siRNAs and found eight criteria for improving siRNA selection: (1) G/C content 30−52%, (2) at least 3 As or Us at positions from 15 to 19, (3) absence of internal repeats, (4) an A at position 19, (5) an A at position 3, (6) a U at position 10, (7) a base other than G or C at position 19, (8) a base other than G at position 13.

However, about 65% of siRNAs produced by design tools based on the above-mentioned design rules have failed when experimentally tested, says, they were 90% in inhibition and nearly 20% of them were found to be inactive [Ren *et al.*, 2006]. One reason is that the previous empirical analyses were only based on small datasets and focused on siRNAs for specific genes. Therefore, each of these rules is poor to individually design highly effective siRNAs.

For nearly a decade, machine learning techniques have alternatively been applied to predict knockdown efficacy of siRNAs. The first predictive model was proposed by [Huesken *et al.*, 2005] in which motifs for effective and ineffective siRNAs were detected based on the significance of nucleotides by using a neural network to train 2,182 scored siRNAs (i.e., siRNAs whose knockdown efficacy (score) was experimentally measured) and test on 249 siRNAs. This dataset was widely used to train and test other regression models [Ichihara *et al.*, 2007, Shabalina *et al.*, 2006, Vert *et al.*, 2006, Gong *et al.*, 2006, Mysara *et al.*, 2012]. Most notably, Qui *et al.* [Qiu *et al.*, 2009] used multiple support vector regression with RNA string kernel for siRNA efficacy prediction, and Sciabola *et al.,* [Sciabola *et al.*, 2013] applied three-dimension structural information of siRNA to increase predictability of the regression model.

It is worth noting that most of those methods suffer from some drawbacks. Their correlations between predicted values and experimental values of the dependent variable ranging from 0.60 to 0.68 were considerably decreased when tested on independent datasets. It may be caused by the fact that the Huesken dataset is still too small to be representative

of the siRNA population having about 4^{19} possible siRNAs. Another reason mentioned in Chapter 3 is that the performance of machine learning methods is heavily dependent on the choice of data representation (or features) on which they are applied. However, these representations of previous works are not good to represent siRNAs in oder to build a good model. Alternatively, several works [Klingelhoefer *et al.*, 2009 , Chang *et al.*, 2012] used classification methods on siRNAs which were experimentally labeled in terms of knockdown efficacy. This siRNA dataset, hereafter called labeled siRNA dataset, was taken from the siRecord database [Ren *et al.*, 2006] consisting of siRNAs classified into four classes with labels 'very high', 'high', 'medium' and 'low'. The classification methods build classifiers from the labeled siRNA dataset to predict the class labels of unknown siRNAs.

Our work aims to develop methods for better prediction of the siRNA knockdown efficacy by exploiting both scored and labeled siRNA datasets and siRNA design rules detected by our proposed method and empirical analysis. In the first method, transformation matrices were learned by incorporating siRNA design rules with labeled siRNAs in the siRecord database. We then used the transformation matrices to transform siRNAs as enriched matrices and do prediction with them by bilinear tensor regression where in the regularization term of the objective function, the Frobenius norm was appropriately replaced by L_2 norm for an effective computation. In the second method, scored siRNAs, labeled siRNAs and siRNA design rules were employed together to learn transformation matrices and parameters of tensor regression models. To obtain more precise data representation, the transformation matrices and parameters were iteratively and simultaneously learned. The labeled dataset was also used to supervise the learning phase. In the objective function, the Frobenius norm was also appropriately replaced by L_2 regularization norm for an effective computation. The experiments shown that the proposed methods achieve better results than most existing models.

The contributions of this work can be summarized as follows

1. Develop novel learning methods to predict the siRNA efficacy by (i) enriching siRNA sequences (solving an optimization problem formulated with design rules, the scored and labeled datasets), and (ii) appropriately learning a bilinear tensor regression model (using L_2 norm instead of Frobenius to effectively learn the set of regression model parameters) for predicting siRNAs.
2. Quantitatively determine positions on siRNAs where nucleotides can strongly influence inhibition ability of siRNAs.
3. Provide guidelines based on positional features for generating highly effective siRNAs.

In the second method, we developed a predictor called SSTR_1 (stands for Bilinear

Tensor Regression) using C++ programming language on X–Code environment. $SSTR_1$ is experimentally compared with published models on the Huesken dataset and three independent datasets commonly used by the research community. The results show that the performance of the $SSTR_1$ predictor is more stable and higher than that of other models.

4.2 Methods

4.2.1 Bilinear tensor regression method

Given a siRNA data set $D = \{(s_l, y_l)\}_1^N$ where s_l is the lth siRNA sequence of size n and $y_l \in \mathbb{R}$ is the knockdown efficacy score of s_l, K transformation matrices learned by siRNA representation method that is presented in chapter 3. Let X_l denotes the encoding matrix of s_l. Each encoding matrix X is transformed to enriched matrix by K transformation matrices, $(T_1 \circ X, T_2 \circ X, \ldots, T_K \circ X)$. $R(X) = (T_1 \circ X, T_2 \circ X, \ldots, T_K \circ X)^T$ denotes the second order tensor of size $K \times n$.

The regression model can be defined as the following bilinear form

$$f(x) = \alpha R(X) \beta \tag{4.1}$$

where $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_K)$ is a weight vector of the K representations of X and $\beta = (\beta_1, \beta_2, \ldots, \beta_n)^T$ is a parameter vector of the model, and $\alpha R(X)$ component is the linear combination of representations $T_1 \circ X, T_2 \circ X, \ldots, T_K \circ X$. It also shows the relationship among elements on each column of the second order tensor or each dimension of $T_k \circ X, k = 1, 2, \ldots, K$. Equation (11) can be derived as follows

$$f(X) \;=\; \alpha R(X) \beta = \left(\beta \otimes \alpha^T\right)^T vec(R(X)) = \left(\beta^T \otimes \alpha\right) vec(R(X)) \tag{4.2}$$

where $A \otimes B$ is the Kronecker product of two matrices A and B, and $vec(A)$ is the vectorization of matrix A. The weight vector α and the parameter vector β are learned by minimizing the following regularized risk function

$$L(\alpha, \beta) \;=\; \sum_{l=1}^{N} (y_l - \alpha R(X_l) \beta)^2 + \lambda \parallel \beta^T \otimes \alpha \parallel_{Fro}^2 \tag{4.3}$$

where λ is the turning parameter to tradeoff between bias and variance, and $\parallel \beta^T \otimes \alpha \parallel_{Fro}$ is the Frobenius norm of the first order tensor $\beta^T \otimes \alpha$. $L(\alpha, \beta)$ can be derived as follows

$$
\begin{aligned}
L(\alpha,\beta) &= \sum_{l=1}^{N}\left(y_l-\alpha R(X_l)\beta\right)^2+\lambda\sum_{k=1}^{K}\sum_{j=1}^{n}\left(\alpha_k\beta_j\right)^2=\sum_{l=1}^{N}\left(y_l-\alpha R(X_l)\beta\right)^2+\lambda\sum_{k=1}^{K}\alpha_k^2\sum_{j=1}^{n}\beta_j^2 \\
&= \sum_{l=1}^{N}\left(y_l-\alpha R(X_l)\beta\right)^2+\lambda\sum_{k=1}^{K}\alpha_k^2\parallel\beta\parallel_2^2=\sum_{l=1}^{N}\left(y_l-\alpha R(X_l)\beta\right)^2+\lambda\parallel\alpha\parallel_2^2\parallel\beta\parallel_2^2 \quad (4.4)
\end{aligned}
$$

The risk function with Frobenius norm is converted to equation (14) with L_2 norm. In order to solve this optimization problem, an alternative iteration method is used. At each iteration, the parameter vector β is effectively solved by keeping the weight vector α and vice versa.

$$
\begin{aligned}
\tfrac{\partial L(\alpha,\beta)}{\partial\alpha} &= -2\sum_{l=1}^{N}\left(y_l-\alpha R(X_l)\beta\right)\left(R(X_l)\beta\right)^T+2\lambda\alpha\parallel\beta\parallel_2^2=0 \\
\Leftrightarrow \quad & \sum_{l=1}^{N}\alpha\left(R(X_l)\beta\right)\left(R(X_l)\beta\right)^T-\sum_{l=1}^{N}y_l\left(R(X_l)\beta\right)^T+\lambda\alpha\parallel\beta\parallel_2^2=0
\end{aligned}
$$

$$
\Rightarrow \quad \alpha = \sum_{l=1}^{N}y_l\left(R(X_l)\beta\right)^T\left(\sum_{l=1}^{N}\left(R(X_l)\beta\right)\left(R(X_l)\beta\right)^T+\lambda\parallel\beta\parallel_2^2 I\right)^{-1} \quad (4.5)
$$

$$
\begin{aligned}
\tfrac{\partial L(\alpha,\beta)}{\partial\beta} &= -2\sum_{l=1}^{N}\left(y_l-\alpha R(X_l)\beta\right)\left(\alpha R(X_l)\right)^T+2\lambda\beta\parallel\alpha\parallel_2^2=0 \\
\Leftrightarrow \quad & \sum_{l=1}^{N}\alpha R(X_l)\beta\left(\alpha R(X_l)\right)^T-\sum_{l=1}^{N}y_l\left(\alpha R(X_l)\right)^T+\lambda\beta\parallel\alpha\parallel_2^2=0 \\
\Leftrightarrow \quad & \sum_{l=1}^{N}\left(\left(\alpha R(X_l)\right)^T\otimes\left(\alpha R(X_l)\right)\right)\beta-\sum_{l=1}^{N}y_l\left(\alpha R(X_l)\right)^T+\lambda\beta\parallel\alpha\parallel_2^2=0 \\
\Rightarrow \quad \beta &= \left(\sum_{l=1}^{N}\left(\left(\alpha R(X_l)\right)^T\otimes\left(\alpha R(X_l)\right)\right)+\lambda\parallel\alpha\parallel_2^2 I\right)^{-1}\sum_{l=1}^{N}y_l\left(\alpha R(X_l)\right)^T \quad (4.6)
\end{aligned}
$$

Our proposed tensor regression model learning is summarized in Algorithm 4. In this algorithm, siRNA sequences are firstly represented as encoding matrices. The encoding matrices are then transformed to tensors by using K transformation matrices. After that, the weight vector α and the coefficient vector β are updated until meeting the convergence criteria, where t_{Max} denotes the maximum iterative step to update α and β, and ϵ_1 and ϵ_2 are thresholds for vectors α and β.

Algorithm 4 Tensor Regression Model Learning

Input: A data set $S = \{(s_i, y_i)\}_1^N$ where s_i are scoring siRNA sequences and $y_i \in \mathbb{R}$. K transformation matrices $R_1, R_2, \ldots, R_k$, and the length n of siRNA sequence.
Output: Weight vector $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_k)$ and parameter vector $\beta = (\beta_1, \beta_2, \ldots, \beta_n)$ that minimize the regularized risk function
– Represent siRNA sequences in S as enconding matrices.
– Transform encoding matrices to tensors using K transformation matrices.
– Initialize α and β randomly.
– $t = 0$ { Iterative step}
repeat
$t \leftarrow t + 1$
Compute $\alpha^{(t)}$ using equation (15)
Compute $\beta^{(t)}$ using equation (16)
until $((\frac{\|\alpha^{(t)} - \alpha^{(t-1)}\|_2}{\|\alpha^{(t-1)}\|_2} \leq \epsilon_1)$ and $(\frac{\|\beta^{(t)} - \beta^{(t-1)}\|_2}{\|\beta^{(t-1)}\|_2} \leq \epsilon_2))$ or $(t > t_{Max})$

4.2.2 Semi–supervised tensor regression method

We formulate the problem of siRNA knockdown efficacy prediction as follows

- **Given:** Two sets of labeled siRNA and scored siRNA sequences of length n, and a set of K siRNA design rules.
- **Find:** A function that predicts the knockdown efficacy of given siRNAs.

Our proposed method consists of three major steps that are described in Table 4.1.

Table 4.1: Method for siRNA knockdown efficacy prediction

1. To encode each siRNA sequence as an encoding matrix X representing the nucleotides A, C, G, and U at n positions in the sequence. Thus, siRNA sequences are represented as $n \times 4$ encoding matrices.
2. To transform siRNA encoding matrices by K transformation matrices T_k into enriched matrices, $k = 1, \ldots, K$. Each transformation matrix characterizes the knockdown ability of nucleotides A, C, G, and U at n positions in the siRNA sequence regarding the kth design rule. Each T_k captures background knowledge of the kth design rule. The enriched matrices of size $K \times n$ are considered as second order tensor representations of the siRNA sequences.
3. To build and learn a bilinear tensor regression model. In this step, K transformation matrices as well as parameters of the model are learned together with the labeled and scored siRNAs and available siRNA design rules. The final model is used to predict the knockdown ability of new siRNAs.

Step 1 of the method is done where each siRNA sequence with n nucleotides in length is encoded as a binary encoding matrix of size $n \times 4$. In fact, four nucleotides A, C, G,

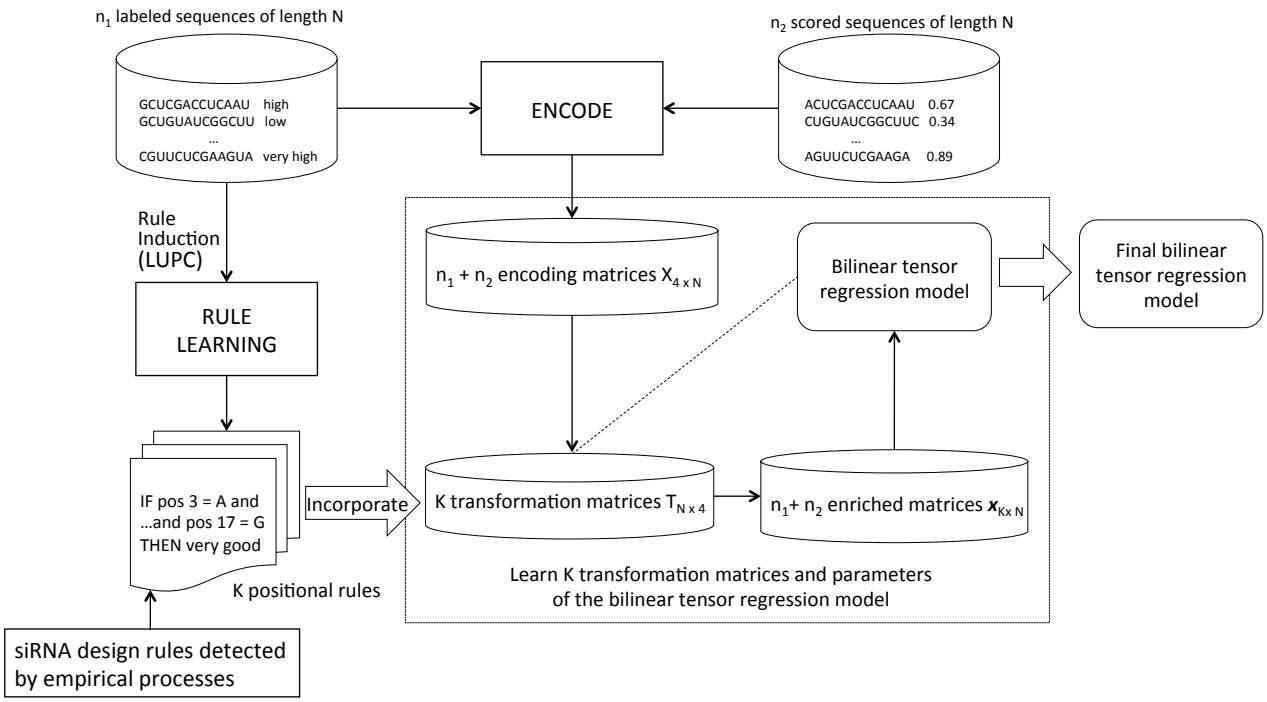


Figure 4.1: The semi–supervised tensor regression model to predict knockdown efficacy of siRNAs. n_2 enriched matrices were used to supervise the learning phase of transformation matrices and parameters of the model.

or U are encoded by encoding vectors (1, 0, 0, 0), (0, 1, 0, 0), (0, 0, 1, 0) and (0, 0, 0, 1), respectively. If a nucleotide from A, C, G, and U appears at the jth position in a siRNA sequence, $j = 1, ..., n$, its encoding vector will be used to encode the jth row of the encoding matrix.

Step 2 is to transform the encoding matrices by transformation matrices T_k regarding the kth design rule, $k = 1, ..., K$. T_k has size of $4 \times n$ where the rows correspond to nucleotides A, C, G, and U, and the columns correspond to n positions on sequences. T_k are learned from the kth design rule. Each cell $T_k[i, j]$, $i = 1, ..., 4$, $j = 1, \ldots, n$, represents the knockdown ability of nucleotide i at position j regarding the kth design rule. Each cell $T_k[i, j]$ to be learned has to satisfy a number of constraints. The first type of constraints is basic constraints on elements of T_k

$$T_k[i, j] \geq 0, \ i = 1, ..., 4; \quad j = 1, 2, \ldots, n \tag{4.7}$$

The second type of constraints relates to the design rules. Each design rule propositionally describes the occurrence or absence of nucleotides at different positions on effective siRNA sequences. Therefore, if a design rule shows the occurrence (or absence) of some nucleotides on jth position, then their corresponding values in the matrix T_k would be greater (or smaller) than other values at column j. For example, the design rule in the right table in Figure 3.1 illustrates that at position 19, nucleotides A/U are effective and

nucleotide C is ineffective. It means that the knockdown efficacy of nucleotides A/U are larger than that of nucleotides G/C and knockdown efficacy of nucleotide C is smaller than that of the other nucleotides. Thus, values $T[1,19], T[2,19], T[3,19]$ and $T[4,19]$ show the knockdown efficacy of nucleotides A, C, G and U at position 19, respectively. Therefore, five constraints at column 19 of T are formed. Generally, we denote the set of M_k trick inequality constraints on T_k by the design rule under consideration by

$$\{g_m(T_k) < 0\}_{m=1}^{M_k} \tag{4.8}$$

Let vector $x_l^{(k)}$ of size $1 \times n$ denote the transformed vector of the lth siRNA sequence using the transformation matrix T_k. The jth element of x_l is the element of T_k at column j and the row corresponds to the jth nucleotide in the siRNA sequence. To compute $x_l^{(k)}$, a new column-wise inner product is defined as follows

$$x_l^{(k)} = T_k \circ X_l = (X_l[1,.]T_k[.,1], X_l[2,.]T_k[.,2], \ldots, X_l[n,.]T_k[.,n]) \tag{4.9}$$

where $X_l[j,.]$ and $T[.,j]$ are the jth row vector and the jth column of the matrix X_l and T, respectively, and xy is the inner product of vectors x and y.

The left table in Figure 3.1 shows an example of encoding matrix X, transformation matrix T and transformed vector x of the given sequence AUGCU. The rows of X represent encoding vectors of nucleotides in the sequence. Given transformation matrix T of size 4×5. The sequence AUGCU is represented by the vector $x = (T[1,1], T[4,1], T[3,3], T[2,4], T[4,5]) = (0.5,\ 0.1,\ 0.08,\ 0.6,\ 0.1)$. Therefore, the transformed data can be computed by the column-wise inner product $x = T \circ X_l$.

The third type of constraints relates to preservation of the siRNA classes after being transformed by using transformation matrices T_k. It means that siRNAs belonging to the same class should be more similar to each other than siRNAs belonging to the other class. This constraint is formulated as the following minimization problem

$$\min \sum_{p,q \in N_1} d^2(x_p^{(k)}, x_q^{(k)}) + \sum_{p,q \in N_2} d^2(x_p^{(k)}, x_q^{(k)}) - \sum_{\substack{p \in N_1 \\ q \in N_2}} d^2(x_p^{(k)}, x_q^{(k)}) \tag{4.10}$$

In this objective function, the first two components are the sum of similarities of sequence pairs belonging to the same class and the last one is the sum of similarities of sequence pairs belonging to two different classes; $d(x,y)$ is the similarity measure between x and y (in this work we use Euclidean distance and L_2 norm); N_1 and N_2 are the two index sets of 'very high' and 'low' labeled siRNAs, respectively.

In step 3 of the method, each encoding matrix X_l is transformed to K representations $(x_l^{(1)}, x_l^{(2)}, \ldots, x_l^{(K)})$ or $(T_1 \circ X_l, T_2 \circ \ X_l, \ldots, T_K \circ \ X_l)$ by K transformation matrices. Denote $R(X_l) = (T_1 \circ X_l, T_2 \circ \ X_l, \ldots, T_K \circ X_l)^T$ be the second order tensor of size $K \times n$. The regression model can be defined as the following bilinear form

$$f(x) = \alpha R(X_l)\beta \tag{4.11}$$

where $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_K)$ is a weight vector of the K representations of X_l and $\beta = (\beta_1, \beta_2, \ldots, \beta_n)^T$ is a parameter vector of the model, and $\alpha R(X_l)$ component is the linear combination of representations $T_1 \circ X_l, T_2 \circ X_l, \ldots, T_K \circ X_l$. It also shows the relationship among elements on each column of the second order tensor or each dimension of $T_k \ \circ \ X_l, \ k \ = \ 1, 2, \ldots, K$. Equation (4.11) can be derived as follows

$$f(X_l) = \alpha R(X_l)\beta = \left(\beta \otimes \alpha^T\right)^T vec(R(X_l)) = \left(\beta^T \otimes \alpha\right) vec(R(X_l))$$

where $A \otimes B$ is the Kronecker product of two matrices A and B, and $vec(A)$ is the vectorization of matrix A.

The fourth type of constraints related to the knockdown efficacy of labeled siRNAs. The labeled siRNAs are used to supervise the learning phase of transformation matrices and the parameters of the model. siRNAs were classified by ordinal labels ('very high' and 'low' labels), therefore, the learned model to predict knockdown efficacy of siRNAs have to preserve ordinal property of the class for siRNAs in the labeled dataset. In particular, the knockdown efficacy of each siRNA sequence in the 'very high' class has to greater than that of siRNAs in the 'low' class. Therefore, let X_p denote the encoding matrix of the pth sequence in the 'very high' class and X_q denote the encoding matrix of the qth sequence in the 'low' class. We have the following constraints

$$(f(X_q) - f(X_p)) \leq 0 \Leftrightarrow \alpha(R(X_q) - R(X_p))\beta \leq 0, \ p \in N_1, q \in N_2 \tag{4.12}$$

Therefore, the regularized risk function satisfies the constraints (4.12) is formulated as follows

$$L(\alpha, \beta) = \sum_{l=1}^{N} (y_l - \alpha R(X_l)\beta)^2 + \lambda_1 \parallel \beta^T \otimes \alpha \parallel_{Fro}^2 + 2\lambda_2 \sum_{\substack{p \in N_1 \\ q \in N_2}} \alpha(R(X_q) - R(X_p))\beta \tag{4.13}$$

where $\lambda_1, \ \lambda_2$ are the turning parameters, and $\parallel \beta^T \otimes \alpha \parallel_{Fro}$ is the Frobenius norm of the first order tensor $\beta^T \otimes \alpha$. X_l and y_l are encoding matrix of the lth sequence and its knockdown efficacy in the scoring siRNA dataset, and N is the size of the scoring siRNA

sequences. The regularization term in equation (4.13) is derived as follows

$$\| \beta^T \otimes \alpha \|^2_{Fro} = \sum_{k=1}^{K}\sum_{j=1}^{n}(\alpha_k\beta_j)^2 = \sum_{k=1}^{K}\alpha_k^2 \sum_{j=1}^{n}\beta_j^2 = \sum_{k=1}^{K}\alpha_k^2 \parallel \beta \parallel_2^2 = \parallel \alpha \parallel_2^2 \parallel \beta \parallel_2^2$$

Therefore, equation (4.13) with the Frobenius norm can be replaced by L_2 norm

$$L(\alpha,\beta) = \sum_{l=1}^{N}\left(y_l - \alpha R(X_l)\beta\right)^2 + \lambda_1 \parallel \alpha \parallel_2^2 \parallel \beta \parallel_2^2 + 2\lambda_2 \sum_{\substack{p\in N_1\\ q\in N_2}} \alpha(R(X_q) - R(X_p))\beta \qquad (4.14)$$

The problem has now become the following multi–objective optimization problem: Finding $\{T_k\}_1^K$, α and β to minimize objective function (4.10) under the constraints (4.7), (4.8) and minimize objective function (4.14). This multi–objective optimization problem leads to solve the following optimization problem

$$\begin{aligned} L(T_1,\ldots,T_K,\alpha,\beta) = &\sum_{l=1}^{N}\left(y_l - \alpha R(X_l)\beta\right)^2 + \lambda_1 \parallel \alpha \parallel_2^2 \parallel \beta \parallel_2^2 + \lambda_2 \sum_{\substack{p\in N_1\\ q\in N_2}} \alpha(R(X_q) - R(X_p))\beta \\ &+\lambda_3 \sum_{k=1}^{K}\left(\sum_{p,q\in N_1} d^2(x_p^{(k)}, x_q^{(k)}) + \sum_{p,q\in N_2} d^2(x_p^{(k)}, x_q^{(k)}) - \sum_{\substack{p\in N_1\\ q\in N_2}} d^2(x_p^{(k)}, x_q^{(k)})\right) \end{aligned}$$

$$\begin{gathered} \text{Subject to } \; T_k[i,j] \geq 0, \; g_m(T_k) \; < \; 0, \\ i = \; 1,...,4; \; j = \; 1,...,n; \; k \; = \; 1,..,K; \; m = \; 1,..,M_k. \end{gathered}$$

This optimization problem is solved by the following Lagrangian form

$$\begin{aligned} L = &\sum_{l=1}^{N}\left(y_l - \alpha R(X_l)\beta\right)^2 + \lambda_1 \parallel \alpha \parallel_2^2 \parallel \beta \parallel_2^2 + 2\lambda_2 \sum_{\substack{p\in N_1\\ q\in N_2}} \alpha(R(X_q) - R(X_p))\beta \\ &+\lambda_3 \sum_{k=1}^{K}\Big(\sum_{p,q\in N_1} d^2(x_p^{(k)}, x_q^{(k)}) + \sum_{p,q\in N_2} d^2(x_p^{(k)}, x_q^{(k)}) - \sum_{\substack{p\in N_1\\ q\in N_2}} d^2(x_p^{(k)}, x_q^{(k)})\Big) + \sum_{k=1}^{K}\sum_{m=1}^{M_k}\mu_m^{(k)} g_m(T_k) \end{aligned}$$

where $\mu_m^{(k)}, \; m = 1,...,M_k; \; k = 1,...,K$ and $\lambda_j, \; j = 1,\ldots,3$ are Lagrangian multipliers. To solve the problem, an iterative method is applied. For each column j, $T_k[.,j]$ is solved while keeping the other columns of T_k. α and β are also solved while keeping the others. The Karush-Kuhn-Tucker conditions are

- Stationarity: $\frac{\partial L}{\partial T_k[.,j]} = 0, \; \frac{\partial L}{\partial \alpha} = 0, \; \frac{\partial L}{\partial \beta} = 0,$
  $i = 1,...,4; \; k = 1,...,K;$ and $j = 1,\ldots,n.$

- Primal feasibility: $T_k[i,j] \geq 0,\ g_r(T_k) < 0,$
  $i = 1, ..., 4;\ j = 1, ..., n;\ r = 1, ..., R;\ k = 1, ..., K.$
- Dual feasibility: $\mu_m^{(k)} \geq 0, \lambda_j \geq 0,\ m = 1, \ldots, M_k;$
  $k = 1, ..., K;\ j = 1, ..., 3.$
- Complementary slackness: $\mu_m^{(k)} g_m(T_k) = 0,$
  $m = 1, \ldots, M_k;\ k = 1, ..., K.$

From the last three conditions, we have $\mu_m^{(k)} = 0,\ m = 1, \ldots, M_k;\ k = 1, ..., K.$ Therefore, the stationarity condition can be derived as follows

$$\frac{\partial L}{\partial T_k[.,j]} = \frac{\partial \sum_{l=1}^{N} (y_l - \alpha R(X_l)\beta)^2}{\partial T_k[.,j]} + 2\lambda_2 \frac{\partial \sum_{\substack{p\in N_1 \\ q\in N_2}} \alpha(R(X_q) - R(X_p))\beta}{\partial T_k[.,j]}$$
$$+\lambda_3 \left(\frac{\partial \sum_{k=1}^{K} (\sum_{p,q\in N_1} d^2(x_p^{(k)}, x_q^{(k)}) + \sum_{p,q\in N_2} d^2(x_p^{(k)}, x_q^{(k)})}{\partial T_k[.,j]} - \frac{\partial \sum_{\substack{p\in N_1 \\ q\in N_2}} d^2(x_p^{(k)}, x_q^{(k)}))}{\partial T_k[.,j]} \right)$$

$$= -2\alpha_k\beta_j \left(\sum_{l=1}^{N} (y_l - \alpha R(X_l)\beta)\, X_l^T[j,.] + \lambda_2 \sum_{\substack{p\in N_1 \\ q\in N_2}} (X_p[j,.] - X_q[j,.])^T \right)$$
$$+2\lambda_3 \sum_{p,q\in N_1} (\langle X_p[j,.], T_k[.,j]\rangle - \langle X_q[j,.], T_k[.,j]\rangle)(X_p[j,.] - X_q[j,.])^T$$
$$+2\lambda_3 \sum_{p,q\in N_2} (\langle X_p[j,.], T_k[.,j]\rangle - \langle X_q[j,.], T_k[.,j]\rangle)(X_p[j,.] - X_q[j,.])^T$$
$$-2\lambda_3 \sum_{\substack{p\in N_1 \\ q\in N_2}} (\langle X_p[j,.], T_k[.,j]\rangle - \langle X_q[j,.], T_k[.,j]\rangle)(X_p[j,.] - X_q[j,.])^T$$
$$= 0$$

Set $Z_{p,q} = (X_p - X_q)$ and set $\alpha(R(X_l))_{kj}\beta = \alpha R(X_l)\beta - \alpha_k\beta_j X_l[j,.]T_k[.,j]$. Therefore, the above formulation is derived as follows

$$\frac{\partial L}{\partial T_k[.,j]} = -2\alpha_k\beta_j \left(\sum_{l=1}^{N} (y_l - \alpha(R(X_l))_{kj}\beta)\, X_l^T[j,.] + \lambda_2 \sum_{\substack{p\in N_1 \\ q\in N_2}} Z_{p,q}[j,.]^T \right)$$
$$+2\Bigg(\lambda_3 \Big(\sum_{p,q\in N_1} Z_{p,q}^T[j,.] \otimes Z_{p,q}[j,.] + \sum_{p,q\in N_2} Z_{p,q}^T[j,.] \otimes Z_{p,q}[j,.]$$
$$- \sum_{\substack{p\in N_1 \\ q\in N_2}} Z_{p,q}^T[j,.] \otimes Z_{p,q}[j,.] \Big) + \alpha_k^2\beta_j^2 \sum_{l=1}^{N} X_l^T[j,.] \otimes X_l^T[j,.] \Bigg) T_k[.,j]$$
$$= 0$$

We define the following equations

$$S(k,j) = \lambda_3 \Bigg(\sum_{p,q \in N_1} Z_{p,q}^T[j,.] \otimes Z_{p,q}[j,.] + \sum_{p,q \in N_2} Z_{p,q}^T[j,.] \otimes Z_{p,q}[j,.] - \sum_{\substack{p \in N_1 \\ q \in N_2}} Z_{p,q}^T[j,.] \otimes Z_{p,q}[j,.] \Bigg) + \alpha_k^2 \beta_j^2 \sum_{l=1}^{N} X_l^T[j,.] \otimes X_l^T[j,.] \tag{4.15}$$

$$B(k,j) = \alpha_k \beta_j \Bigg(\sum_{l=1}^{N} \left(y_l - \alpha(R(X_l))_{kj} \beta\right) X_l^T[j,.] + \lambda_2 \sum_{\substack{p \in N_1 \\ q \in N_2}} Z_{p,q}[j,.]^T \Bigg) \tag{4.16}$$

Substitute equations (4.15) and (4.16) to $\frac{\partial L}{\partial T_k[.,j]}$, we have

$$T_k[.,j] = S(k,j)^{-1} B(k,j) \tag{4.17}$$

$$\begin{aligned} \frac{\partial L}{\partial \alpha} =& -2 \sum_{l=1}^{N} \left(y_l - \alpha R(X_l)\beta\right) \left(R(X_l)\beta\right)^T + 2\lambda_1 \parallel \beta \parallel_2^2 \alpha + 2\lambda_2 \Big(\sum_{\substack{p \in N_1 \\ q \in N_2}} \left(R(X_q) - R(X_p)\right)\beta \Big)^T \\ =& \sum_{l=1}^{N} \alpha \left(R(X_l)\beta\right) \left(R(X_l)\beta\right)^T - \sum_{l=1}^{N} y_l \left(R(X_l)\beta\right)^T + \lambda_1 \parallel \beta \parallel_2^2 \alpha - \lambda_2 \beta^T \Big(\sum_{\substack{p \in N_1 \\ q \in N_2}} \left(R(X_p) - R(X_q)\right) \Big)^T = 0 \end{aligned}$$

$$\alpha = \left(\sum_{l=1}^{N} y_l \left(R(X_l)\beta\right)^T + \lambda_2 \beta^T \Big(\sum_{\substack{p \in N_1 \\ q \in N_2}} \left(R(X_p) - R(X_q)\right) \Big)^T \right) \times \left(\sum_{l=1}^{N} \left(R(X_l)\beta\right) \left(R(X_l)\beta\right)^T + \lambda_1 \parallel \beta \parallel_2^2 I \right)^{-1} \tag{4.18}$$

$$\begin{aligned} \frac{\partial L}{\partial \beta} =& -2 \sum_{l=1}^{N} \left(y_l - \alpha R(X_l)\beta\right) \left(\alpha R(X_l)\right)^T + 2\lambda_1 \parallel \alpha \parallel_2^2 \beta + 2\lambda_2 \Big(\sum_{\substack{p \in N_1 \\ q \in N_2}} \alpha\left(R(X_q) - R(X_p)\right) \Big)^T \\ =& \sum_{l=1}^{N} \alpha R(X_l)\beta \left(\alpha R(X_l)\right)^T - \sum_{l=1}^{N} y_l \left(\alpha R(X_l)\right)^T + \lambda_1 \parallel \alpha \parallel_2^2 \beta - \lambda_2 \left(\alpha \sum_{\substack{p \in N_1 \\ q \in N_2}} \left(R(X_p) - R(X_q)\right) \right)^T \\ =& \sum_{l=1}^{N} \Big(\left(\alpha R(X_l)\right)^T \otimes \left(\alpha R(X_l)\right) \Big) \beta - \sum_{l=1}^{N} y_l \left(\alpha R(X_l)\right)^T + \lambda_1 \parallel \alpha \parallel_2^2 \beta - \lambda_2 \left(\alpha \sum_{\substack{p \in N_1 \\ q \in N_2}} \left(R(X_p) - R(X_q)\right) \right)^T \\ =& 0 \end{aligned}$$

$$\beta = \left(\sum_{l=1}^{N} \Big(\left(\alpha R(X_l)\right)^T \otimes \left(\alpha R(X_l)\right) \Big) + \lambda_1 \parallel \alpha \parallel_2^2 I \right)^{-1} \times \left(\sum_{l=1}^{N} y_l \left(\alpha R(X_l)\right)^T + \lambda_2 \left(\alpha \sum_{\substack{p \in N_1 \\ q \in N_2}} \left(R(X_p) - R(X_q)\right) \right)^T \right) \tag{4.19}$$

Algorithm 5 Tensor Regression Learning

Input: A data set $L = \{(s_l, y_l)\}_1^N$ where s_l are siRNA sequences and y_l are their labels, a set DR of K design rules, the length n of siRNA sequences. A data set $S = \{(s_i, y_i)\}_1^N$ where s_i are scored siRNA sequences and $y_i \in \mathbb{R}$

Output: K transformation matrices $T_1, T_2, \ldots, T_K$.

Encoding siRNA sequences in S and L.

for $rule_k$ in DR **do**

– Form the set of constraints C_k based on $rule_k$

– Initialize the transformation matrix T_k satisfying C_k.

end for

– Initialize α and β randomly.

$t = 0$ { Iterative step}

repeat

$t \leftarrow t + 1$

for $k = 1$ to K **do**

for $j = 1$ to n **do**

$v = S(k, j)^{-1} B(k, j)$ { Using equation (4.17)}

if (v satisfies the constraints at the position j in C_k) **then**

$T_k^{(t)}[., j] \leftarrow v$

end if

end for

end for

Compute $\alpha^{(t)}$ using equation (4.18)

Compute $\beta^{(t)}$ using equation (4.19)

until $((\frac{\|T_k^{(t)} - T_k^{(t-1)}\|_{Fro}}{\|T_k^{(t-1)}\|_{Fro}} \leq \epsilon)$ and $(\frac{\|\alpha^{(t)} - \alpha^{(t-1)}\|_2}{\|\alpha^{(t-1)}\|_2} \leq \epsilon_1)$ and $(\frac{\|\beta^{(t)} - \beta^{(t-1)}\|_2}{\|\beta^{(t-1)}\|_2} \leq \epsilon_2))$ or $(t > t_{Max})$

The learning phase of the proposed bilinear tensor regression model is summarized in Algorithm 5. In this algorithm, siRNA sequences are first represented as encoding matrices, and the transformation matrices, vectors α and β are initialized. K design rules are used to learn K transformation matrices. For each siRNA design rule, the algorithm will update elements in each column of the transformation matrix according to equation (4.17). In each iterative step, the transformation matrix without trick inequality constraints is updated to reach the global optimal solution. If updated elements in a column satisfy the trick inequality constraints characterizing the condition at the corresponding position of the rule, that row will be updated to the solution containing these constraints. The transformation matrices, vectors α and β are updated until meeting the convergence criteria, where t_{Max} denotes the maximum iterative step to update α and β, and ϵ, ϵ_1 and ϵ_2 are thresholds for the transformation matrices, vectors α and β, respectively.

4.3 Experimental evaluation

This section presents experimental evaluation of three models that were built based on two sets of design rules. The first set of siRNA design rules was discovered by empirical processes and the other was the set of predictive rules detected by the LUPC method. The first model (called BiLTR model) was learned by incorporating siRNA design rules to learn transformation matrices in the bilinear tensor regression method. The second model (called $SSTR_1$ model) and the last model (called $SSTR_2$ model) were learned by integrating siRNA design rules (the first set of design rule) and predictive rules (the second set of design rules) in the semi–supervised tensor regression method, respectively.

4.3.1 Experiment setting

Data sets and siRNA design rules

The proposed methods are experimentally evaluated for the prediction problem of highly effective siRNAs. The methods are compared to most state-of-the-art methods for siRNA knockdown efficacy prediction recently reported in the literature. As experiments in those methods cannot be repeated directly, we employed the results reported in the literature and carried out experiments on our developed methods in the same conditions of the other works. The comparison is carried out using four datasets

- The Huesken dataset of 2431 siRNA sequences targeting 34 human and rodent mRNAs, commonly divided into the training set HU_train of 2182 siRNAs and the testing set HU_test of 249 siRNAs [Huesken *et al.*, 2005].
- The Reynolds dataset of 240 siRNAs [Reynolds *et al.*, 2004].
- The Vicker dataset of 76 siRNA sequences targeting two genes [Vickers *et al.*, 2003].
- The Harborth dataset of 44 siRNA sequences targeting one gene [Takasaki *et al.*, 2010].

In the training phase, the labeled dataset used for the proposed methods was collected from siRecord database [Ren *et al.*, 2006]. This dataset consists of 2470 siRNA sequences in 'very high' class and 2514 siRNA sequences in 'low' and 'medium' classes. To improve the balance between classes while keeping the separation between them, 'medium' and 'low' siRNAs were merged into one class. Each siRNA sequence has 19 nucleotides. The scored siRNA dataset was used to learn models is the HU_train dataset consisting of 2181 siRNAs.

The convergence criteria in Algorithm 3 are set up as following: threshold ϵ for transformation matrices is $2.5E^{-8}$ and the maximum iterative step is 5000.

We alternatively employed two groups of prediction rules to learn transformation matrices and two models :

- Prediction rules found by empirical processes: Seven siRNA design rules used to learn matrices are the Reynolds rule, the Uitei Rule, the Amarzguioui rule, the Jalag rule, the Takasaki rule, the Hsieh rule, and the Huesken rule [Reynolds *et al.*, 2004, Uitei *et al.*, 2004, Amarzguioui *et al.*, 2004, Takasaki *et al.*, 2010, Jagla *et al.*, 2005, Hsieh *et al.*, 2004, Huesken *et al.*, 2005]. These design rules were used to learn transformation matrices in method 1, and employed to learn transformation matrices and parameters of model in method 2. As above–mentioned, these siRNA design rules were used to learn STR and SSTR_1 models.

- Prediction rules learned by the LUPC method from labeled dataset in siRecord database: The cover for the 'very high' class is set $\frac{2314}{2655}$ with the accuracy of $\frac{2615}{2956}$, the LUPC method detected 114 predictive rules when tested on labeled dataset. These predictive rules were employed to learn transformation matrices and parameters of model in method 2. These descriptive rules were employed to learn SSTR_2 model.

Parameter setting

In the BiLTR model, there are two steps to learn the final model. The first step is to learn transformation matrices by using Algorithm 3. As above mentioned, the convergence criteria to implement this algorithm were set up as following: threshold ϵ for transformation matrices is $2.5E^{-8}$ and the maximum iterative step is 5000. The second step is learn parameters of tensor regression model using Algorithm 4. The thresholds for the weight vector α and the coefficient vector β were set up as 0.001 and the maximum iterative step is 1000. The turning parameter λ was chosen by minimizing the risk function when tested on validation dataset. Particularly, we did 10–fold cross validation on the training set for each λ belonging to $[0, \log 50]$ and computed the risk function

$$R(\lambda) = \frac{1}{F}\sum_{i=1}^{F}\left(\frac{1}{\parallel fold_i \parallel}\sum_{x_j \in fold_i}(y_j - f(x_j))^2\right)$$

where $fold_i$ is the validation set, $f(x)$ is a tensor predictor learned by employing the training set except the validation set $fold_i$. F is the number of folds to do cross validation on training set. In our work, we do F-fold cross validation thus F equals to 10.

In the SSTR_1 and SSTR_2 models, thresholds of transformation matrices, vectors α and β to run Algorithm 5 were set by small numbers, actually 0.001. The maximum iterative step of the algorithm is 2000 steps. To compute transformation matrices, weight vector α and coefficient vector β, turning parameters λ_1, λ_2 and λ_3 have to be chosen. By doing cross validation, these parameters were chosen by minimizing the risk function when testing on validation set. Particularly, we do 10–fold cross validation on the Huesken

training set for each turning parameter belonging to the interval [0, log(10)]. The model was trained for each triple of (λ_1, λ_2, λ_3). After that, the risk function was computed

$$R(\lambda_1, \lambda_2, \lambda_3) = \frac{1}{F} \sum_{i=1}^{F} \frac{1}{\parallel fold_i \parallel} L(T_1, \ldots, T_K, \alpha, \beta)$$

where $fold_i$ is the validation set. F is the number of folds to do cross validation on training set. In the experiments we also chosen $F = 10$.

4.3.2 Comparative evaluation

The comparative evaluation is carried out as follows:

1. Comparison of the proposed models with multiple kernel support vector machine proposed by Qui *et al.* [Qiu *et al.*, 2009] whose reported Pearson correlation coefficient (R) of 0.62 obtained by 10-fold cross validation on the whole Huesken dataset. The Pearson correlation coefficient (R) is carefully evaluated by $SSTR_2$ by 10 times of 10-fold cross validation with the average value of 0.64 for all of three models (Table 4.2).
2. Comparison of the three models with four methods Thermocomposition21 [Shabalina *et al.*, 2006], SVM [Sciabola *et al.*, 2013], and DSIR [Vert *et al.*, 2006], BIOPREDsi [Huesken *et al.*, 2005]by HU_train and HU_test. The Pearson correlation coefficients of the four models BIOPREDsi, Thermocomposition21, DSIR and SVM are 0.66, 0.66, 0.67 and 0.80, respectively. The performances of $SSTR_1$, STR and $SSTR_2$ estimated on HU_test are 0.67, 0.68, 0.67, respectively. The performances of our models are equivalent to the performance of the DSIR model, slightly higher than that of the first two models but lower than that of the last model (Table 4.2).
3. Comparison of proposed models with 18 methods including BIOPREDsi, DSIR, Thermocomposition21, and SVM when trained on the HU_train dataset and tested on three independent datasets of Reynolds, Vicker and Harborth as reported in the recent article [Sciabola *et al.*, 2013]. The three models considerably achieved better results than all of 18 methods on the all three independent datasets as shown in Table 4.3 (taken from [Sciabola *et al.*, 2013] with the added the results of three models to the last three rows).

Figure 4.2 shows the relationship between predicted and original inhibition of siRNA sequences of four models $SSTR_1$, i–score, DISR, and BIOPREDsi when tested on the Reynolds dataset. The Pearson correlation coefficient of $SSTR_1$ is greater than those of the others. This figure also shows that most of siRNAs in the Reynolds dataset have

Table 4.2: The R values of models on the the whole Huesken dataset and HU_test dataset

Algorithm	Huesken dataset (2431 siRNAs)	HU_test (249 siRNAs)
Qui's method	0.62	–
BIOPREDsi	–	0.66
Thermocomposition21	–	0.66
DSIR	–	0.67
SVM	–	0.80
BiLTR	**0.64**	**0.68**
SSTR$_1$	**0.64**	**0.67**
SSTR$_2$	**0.64**	**0.67**

Table 4.3: The R values of 18 models and SSTR_1 on three independent data sets

Algorithm	Year	$R^{Reynolds}$ (244si/7g)	R^{Vicker} (76si/2g)	$R^{Harborth}$ (44si/1g)
GPboot	2004	0.55	0.35	0.43
Uitei	2004	0.47	0.58	0.31
Amarzguioui	2004	0.45	0.47	0.34
Hsieh	2004	0.03	0.15	0.17
Takasaki	2010	0.03	0.25	0.01
Reynolds 1	2004	0.35	0.47	0.23
Reynolds 2	2004	0.37	0.44	0.23
Schawarz	2003	0.29	0.35	0.01
Khvorova	2003	0.15	0.19	0.11
Stockholm 1	2004	0.05	0.18	0.28
Stockholm 2	2004	0.00	0.15	0.41
Tree	2004	0.11	0.43	0.06
Luo	2004	0.33	0.27	0.40
i-score	2007	0.54	0.58	0.43
BIOPREDsi	2006	0.53	0.57	0.51
DSIR	2006	0.54	0.49	0.51
Katoh	2007	0.40	0.43	0.44
SVM	2013	0.54	0.52	0.54
BiLTR	**0.60**	**0.58**	**0.55**	
SSTR$_1$	**0.57**	**0.58**	**0.57**	
SSTR$_2$	**0.57**	**0.57**	**0.57**	

high inhibition ability. In the SSTR$_1$ model as well as the BiLTR and SSTR$_2$ models, the transformation matrices are learned by incorporating effective siRNA design rules. Therefore it leads to better performance.

We found that the performance of SSTR$_1$ is more stable and higher comparing with other models. The first reason is that siRNAs are designed by the same siRNA design rules in each dataset and by different siRNA design rules in different datasets. Therefore, the combination of siRNAs design rules is necessary to represent siRNAs. The second

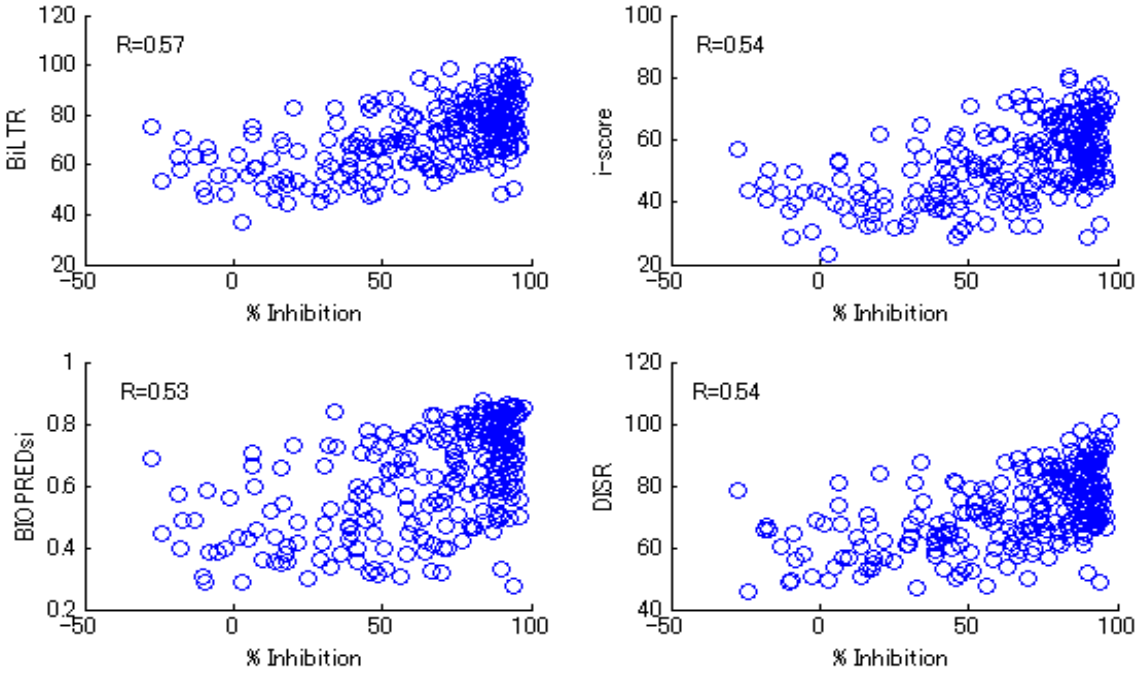


Figure 4.2: Observed siRNA inhibition in the Huesken dataset are plotted against predicted siRNA inhibition by $SSTR_1$, i-score, BIOPREDsi, and DSIR. 'R' values represent the Pearson correlation coefficients, which are also indicated in Table 4.3.

reason is that most current models are trained on the Huesken dataset. The distribution of this dataset is Gaussian and thus those models cannot predict well the knockdown efficacy of siRNAs belonging to other differently distributed datasets. By using labeled siRNA sequences in different distributions to learn our model, $SSTR_1$ model can predict more accurate knockdown efficacy of siRNAs.

4.4 Discussion

4.4.1 Discussion on the BiLTR model

As reported in the experimental comparative evaluation, the proposed BiLTR achieved higher results than most other methods for prediction of siRNA knockdown efficacy. There are some reasons of that. First, it is expensive and hard to analyze the knockdown efficacy of siRNAs, and thus most available datasets are of relatively small size leading to limited results. Second, BiLTR has its advantages by incorporating domain knowledge (siRNA design rules) found from different datasets in experiments. Third, BiLTR is generic and can be easily exploited when new design rules are discovered or more analyzed siRNAs be obtained. Four, one drawback of BiLTR is its transformation matrices are learned

	1	2	3	4		10	11	12		19
A	0.297	0.217	0.423	0.266	...	0.363	0.246	0.224	...	0.393
C	0.231	0.235	0.255	0.226	...	0	0.252	0.267	...	0.0757
G	0.155	0.211	0.0459	0.237	...	0.229	0.221	0.260	...	0.161
U	0.316	0.334	0.275	0.268	...	0.406	0.28	0.246	...	0.368

Table 4.4: The learned transformation matrix containing characteristics of the Reynolds rule.

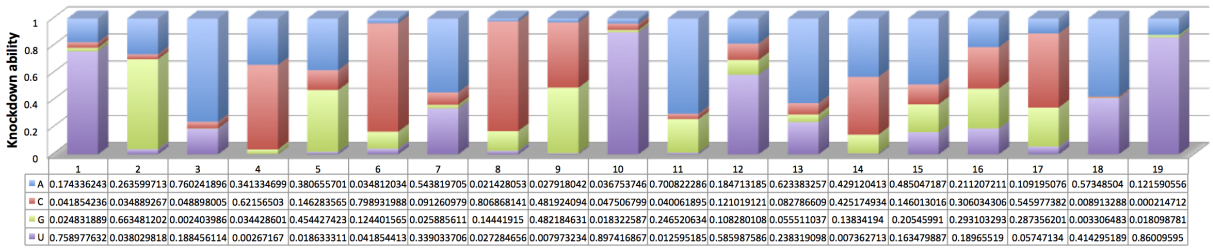
using positional features of available design rules, and thus they lack some characteristics effecting to knockdown efficacy of siRNA sequences such as GC content, thermodynamic properties, GC stretch, etc. It may be one of reasons that at this moment BiLTR cannot get higher performance when testing on HU_test set than the best current model SVM [Sciabola *et al.*, 2013].

Table 4.4 shows the learned transformation matrix capturing positional characteristics of Reynolds rule. One of characteristics is described as "An nucleotide 'A' at position 19". That characteristic means that at column 19, the cell (1,19) has to be the maximum value. In the matrix, the value at this cell is 0.393939 and is the highest value of this column. In this column, we also know knockdown efficacy of each nucleotide at position 19. Therefore, nucleotides can be arranged by the decreasing order of their efficacy: A,U, G, and C. In the order, nucleotide U has efficacy of 0.368687 that also can be used to design effective siRNAs. In addition, if a position on siRNAs is not described in characteristics of the design rules, values at the column corresponding to this position is learned to satisfy classification assumption and property to get knockdown efficacy of each nucleotide such as values at columns 1, 2, 4 and so on.

4.4.2 Discussion on the $SSTR_1$ model

The $SSTR_1$ and $SSTR_2$ models were learned from the same method. Therefore in this section, we only discuss the $SSTR_1$ model that is used to be representative our proposed method. There are three main issues are discussed in more detail: the performance of $SSTR_1$ model, the importance of learned transformation matrices and the effect of nucleotide design at particular positions on siRNAs.

Concerning the first issue, as presented in the experimental comparative evaluation, $SSTR_1$ achieved better results than most of other methods in predicting siRNA knockdown efficacy. There are some reasons for that. First, it is expensive to experimentally analyze the knockdown efficacy of siRNAs, and thus most of available datasets have relatively small sizes leading to limited results. Second, $SSTR_1$ has its advantages by incorporating domain knowledge (siRNA design rules) experimentally found from different datasets. Third, $SSTR_1$ is generic and can be easily exploited when new design rules are discovered,



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
A	0.174336243	0.263599713	0.760241896	0.341334699	0.380655701	0.034812034	0.543819705	0.021428053	0.027918042	0.036753746	0.700822286	0.184713185	0.623383257	0.429120413	0.485047187	0.211207211	0.109195076	0.57348504	0.121590556
C	0.041854236	0.034889267	0.048898005	0.62156503	0.146283565	0.798931988	0.091260979	0.806868141	0.481924094	0.047506799	0.040061895	0.121019121	0.082786609	0.425174934	0.146013016	0.306034306	0.545977382	0.008913288	0.000214712
G	0.024831889	0.663481202	0.002403986	0.034428601	0.454427423	0.124401565	0.025885611	0.14441915	0.482184631	0.018322587	0.246520634	0.108280108	0.055511037	0.13834194	0.20545991	0.293103293	0.287356201	0.003306483	0.018098781
U	0.758977632	0.038029818	0.188456114	0.00267167	0.018633311	0.041854413	0.339033706	0.027284656	0.007973234	0.897416867	0.012595185	0.585987586	0.238319098	0.007362713	0.163479887	0.18965519	0.05747134	0.414295189	0.86009595

Figure 4.3: The learned transformation matrix incorporating positional features of the Reynolds rule. Histogram shows knockdown efficacy strength of each nucleotide at positions on sense siRNA strand.

or more scored or labeled siRNAs are obtained. Fourth, one drawback of $SSTR_1$ is its transformation matrices are learned using positional features of available design rules, and thus they lack some characteristics effecting to the knockdown efficacy of siRNAs such as GC content, thermodynamic properties, GC stretch, etc. It may be one of reasons that at this moment the performance of the $SSTR_1$ model is similar to that of BIOPREDsi, Thermocomposition21, DISR models but cannot achieve higher performance than the best current model SVM [Sciabola *et al.*, 2013] when tested on the HU_test set (Table 4.2). However, when tested on the three independent datasets generated by different empirical experiments, the performance of $SSTR_1$ is better than that of the four above models. Additionally, some models achieve the best results as the $SSTR_1$ model when tested on the Vicker dataset (e.g., i-score, Uitei models) but none of them simultaneously reaches the highest result as $SSTR_1$ when tested on the three independent datasets (Table 4.3).

On the other hand, it is easy to see that the weights α_i, $i = 1, \ldots, K$ show the importance of the siRNA design rules that affect the knockdown efficacy of siRNAs. Figure 4.4 shows the weights of the seven siRNA design rules. The second and the fourth siRNA ones corresponding to the Uitei and Jalag rules have the smallest and highest weights, respectively. The Uitei rule shows that nucleotides 'G/C' at position 1 and 'A/U' at position 19 correlate to effective siRNAs and nucleotides 'A/U' at position 1 and 'G/C' at position 19 correlate to ineffective siRNAs. These characteristics are consistent with most of the other siRNA design rules. However, these characteristics based on positions 1 and 19 are insufficient to generate effective siRNAs. In the fourth rule, except characteristics of the Uitei rule, Jagla and colleagues discovered that effective siRNA have an 'A/U' nucleotide at position 10. It also shows the importance of these nucleotides at position 10 when designing effective siRNAs.

Concerning the second issue, the learned transformation matrices not only capture the characteristics of the siRNA design rules but also guide to create new design rules for generating effective siRNA candidates. Table 4.5 shows the positional features of the Reynolds rule. In this siRNA design rule, effective siRNAs satisfy the following criteria

Figure 4.4: Contributions of seven siRNA design rule to knockdown ability of siRNAs

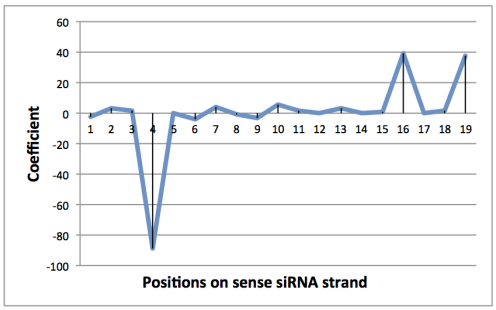


Figure 4.5: Coefficients of 19 dimensions corresponding to 19 position on siRNAs

.

Table 4.5: Characteristics of Huesken rule

Position	3	10	13	19
Effective	A	U	A/C/G	A/U

on sense siRNA strands: (i) nucleotide 'A' at position 3; (ii) nucleotide 'U' at position 10; (iii) nucleotides 'A/C/U' at position 13 and (iv) nucleotides 'A/U' at position 19. After learning $SSTR_1$, the transformation matrix capturing positional features of the Reynolds rule is determined. Figure 4.3 shows the learned transformation matrix incorporated with the Reynolds rule. In this figure, each column of the matrix is normalized to easily observe. One of the characteristics is described as "an nucleotide 'A/U' at position 19". This characteristic means that at column 19, the cell (4,19) should contain the maximum value. In the matrix, the value at this cell is 0.86009595 and is the greatest value in this column. We now consider other characteristics of the Reynolds rule. Another characteristic of this rule is that effective siRNAs have at least three nucleotides 'A/U' at positions from 15 to 19. In learned transformation matrix, corresponding values of nucleotides 'A/U' at positions 15, 18 and 19 are the greatest ones (see Figure 4.3). Therefore, the transformation matrix can preserve this characteristic of the Reynolds rule. One characteristic of siRNAs such as 'G/C' content ranging from 30% to 52% is also preserved in the learned transformation matrix. In addition, positions on siRNAs are not described in characteristics of the design rules, the knockdown efficacy of nucleotides at columns corresponding to these positions are also learned to satisfy the classification assumption and constraints of $SSTR_1$ as values at columns 1, 2, 4 and so on. Therefore, after learning the transformation matrices based on the siRNA design rules, these transformation matrices can guide to generate effective siRNAs. For example, Figure 4.3 shows the Reynolds rule based transformation matrix and its histogram of nucleotides at positions on sense siRNA strand. We can see that effective siRNAs can be designed by using the Reynolds rule and other characteristics such as: 'U' at position 12, 'A' at position 13, and so on.

Concerning the last issue, we consider the effect of nucleotides at particular positions

on siRNAs. In SSTR_1 model, coefficients $\beta_j,\ j = 1, \ldots, 19$, show the strength of the relationship between each variable corresponding to each column of tensors representing siRNAs and the inhibition ability of siRNAs. We know that values of each column show the knockdown efficacy of each nucleotide in a siRNA sequence by incorporating the seven siRNA design rules. Therefore, the coefficients show the influence of nucleotide design at positions on siRNAs to the inhibition ability. In Figure 4.5, the coefficients at positions 4, 16 and 19 show that the siRNA design at these positions will strongly influence the knockdown efficacy or inhibition of siRNAs. Most of the siRNA design rules also capture the importance of designing nucleotides at positions 16 and 19 but they do not mention the designing of nucleotides at position 4. Therefore, the influence of nucleotides at this position can be considered to design effective siRNAs.

4.5 Conclusion

In this paper, we have proposed a novel method to predict the knockdown efficacy of siRNA sequences by using both labeled and scored datasets as well as available design rules to transform the siRNAs into enriched matrices, then learn a bilinear tensor regression model for the prediction purpose.

The experimental comparative evaluation on commonly used datasets with standard evaluation procedure in different contexts shows that the proposed method achieved better results than most existing methods in doing the same task. One significant feature of the proposed method is it can easily be extended when new design rules are discovered as well as more siRNAs are analyzed by empirical processes. By analyzing SSTR_1 model, we provide guidelines to generate effective siRNAs, and detect positions on siRNAs where nucleotides can strongly effect the inhibition ability.

Chapter 5

Conclusion

5.1 Dissertation summary

We have presented our research focusing on an interesting biology problem that is how to synthesize effective siRNAs in order to design novel drugs for treating many kinds of disease such as HIV, cancer, influenza A virus, hepatitis B virus and so on. To tackle this problems, biologists have implemented and analyzed empirical processes and they discovered important characteristics effecting knockdown efficacy of siRNAs. As as result, they reported design rules for effective siRNAs. In computational biology research, research groups have been applied alternative machine learning techniques to detect siRNA design rules and predict knockdown efficacy of siRNAs. We considered both biological and computational aspects of this problem.

Concerning on the detection of design rules for effective siRNAs problem, we first applied the LUPC algorithm proposed by Ho *et al.* to discover predictive rules, and then proposed an descriptive approach to design rational design rules. In the proposed method, some new characteristics of siRNAs that influence knockdown efficacy of siRNAs, were detected. These methods were presented in Chapter 2.

Concerning on the prediction of knockdown efficacy of siRNAs, we first proposed computational methods to enrich siRNA representation. The first representation learning method focuses on the learning of transformation matrices that were incorporated background knowledge of existing design rules in empirical processes as well as predictive rules detected by the LUPC method. Learned transformation matrices are then used to transform binary matrices encoded siRNAs to enriched matrices (the second order tensors). This developed method was reported in Chapter 3 and siRNA representation learned by this method is used for the siRNA knockdown efficacy prediction problem mentioned in Chapter 4. In the second proposed method, transformation matrices were learned by integrating into the learning phase of tensor regression model which also mentioned in Chapter 4. They were learned by using not only siRNA design rules and labeled datasets but also

scored datasets. To build better predictive models, we developed the two methods. The first method used transformation matrices learned in the first data representation learning method to enrich siRNAs and the bilinear tensor regression model was proposed to predict siRNA knockdown efficacy. The second one combined the transformation matrices and parameters of tensor regression model learning phases together to make more higher performance of the model and more precise data representation. Labeled dataset was not only used to transformation matrices but also employed to supervise the learning phase of parameters of model. These two proposed predictive methods was presented in Chapter 4.

The contributions of our research are summarized as follows

- siRNA design rule detection: we developed an adaptive Apriori algorithm to detect effective siRNAs design rules. Detected descriptive rules are filtered and analyzed to generate two rational design rules for effective siRNAs having 19 and 21 nucleotides in length, respectively. New characteristics were also discovered such as GC content ranges from 36% to 52%); the seed region ranging from position 2 to position 7 may play an important role to avoid off-target effects of siRNA that is also one of challenging problems in RNAi [Pei *et al.*, 2006]. Moreover, characteristics of our rational design rule in the region (9-11) can make siRNAs recognize and cleave target mRNAs [Reynolds *et al.*, 2004].

- siRNA knockdown efficacy prediction: we proposed siRNA representation methods that is incorporated important characteristics discovered by empirical works as well as predictive rules found by Ho *et al.*. Tensor regression methods were developed to predict siRNA efficacy by enriching siRNA sequences with domain knowledge and appropriately using bilinear tensor regression. In the objective functions, L2 norm was used instead of Frobenius norm in bilinear tensor regression that allows effectively learning the set of model parameters. By analyzing the models, we quantitatively determined positions on siRNAs where nucleotides can strongly influence inhibition ability of siRNAs. We also provided guidelines based on positional features for generating highly effective siRNAs. A predictor called BiLTR using C plus plus programming language was developed.

5.2 Future work

As above mentioned, our research focused on an interesting and challenging problem of biology. We achieved interesting and promising results, however, our research as well as previous research are still some limitations. In the siRNA design rule detection problem, rational siRNA design rules and new characteristics was found by applying an descriptive

method, however, these rational design rules and new characteristics need to be evaluated by empirical process as well as experts in biology research. Therefore, joint research between biologists and bioinformaticians will be a strong cooperation to solve the biology problem and bring results of research to real applications. In the prediction of siRNA knockdown efficacy, we proposed siRNA representation learning and prediction methods by incorporating background knowledge of siRNA design rule, labeled datasets, and scored datasets, but they lacked of some characteristics effecting to knockdown efficacy of siRNA sequences such as GC content, thermodynamic properties, GC stretch, etc. It can caused that at this moment predictive models do not achieve higher performance. Based on these limitations and current research in both biology and computational biology approaches, our purposes are to study the following problems in our future research

- Finding highly effective siRNAs based on siRNA design rules and predictive models: In our previous works, regression models can predict knockdown efficacy of siRNAs and detected design rules can generate effective siRNAs, but siRNA design rules can not generate all effective ones in the population of 4^{19} siRNAs. Therefore, we should have a strategy to find highly effective siRNAs that can be synthesized to make drugs. In this work, all of important characteristics discovered by previous research should be considered to make siRNA design rules and predictive models more accurate and high performance. To archive good results, the cooperation between our team and biologists is being considered and the results of the research work should be evaluated by empirical processes.

- Designing effective siRNAs targeting specific disease genes. There are important characteristics describing specific diseases that relate to infection, genetic variation, protein structures and so on. Therefore, siRNA based drugs for treating and preventing each disease are the very crucial problem.

- Building a predictive model for minimizing off–target effects problem. Off–target effects of siRNAs is defined as phenomena that siRNAs target to unintended mRNAs and they silence these mRNAs. It leads to the side effects of siRNA based drugs. This problem are now considering one of challenge problems in the designing of effective siRNAs. Therefore, we intend to build models that can predict off-target ability of siRNAs. Models help to find out siRNAs that not only have high knockdown efficacy but also have minimum off-target ability.

Bibliography

[Alistair *et al.*, 2008] Alistair M. C., Erik L. L. Sonnhammer: siRNA specificity searching incorporating mismatch tolerance data. *Bioinformatics*, 24(10), 1316–1317 (2008)

[Amarzguioui *et al.*, 2004] Amarzguioui, M., Prydz, H.: An algorithm for selection of functional siRNA sequences. *Biochem Biophys Res Commun.*, 316(4), 1050–8 (2004).

[Bertrand *et al.*, 2002] Bertrand, J. R., Pottier, M., Vekris, A., Opolon, P., Maksi–menko, A., Malvy, C.: Comparison of antisense oligonucleotides and siRNAs in cell culture and in vivo. *Biochem. Biophys. Res. Commun.*, 296, 1000–1004 (2002).

[Birmingham *et al.*, 2006] Birmingham A., Anderson E.M., Reynolds A. et al.: 3' UTR seed matches, but not overall identity, are associated with RNAi off–targets. *Nat. Methods*, 3, 199–204 (2006).

[Bitko *et al.*, 2001] Bitko, V., Barik, S.: Phenotypic silencing of cytoplasmic genes using sequence-specific double-stranded short interfering RNA and its application in the reverse genetics of wild type negative-strand RNA viruses. *BMC Microbiol.*, 1, 34 (2001).

[Boden *et al.*, 2003] Boden, D., Pusch, O., Lee, F., Tucker, L., Ramratnam, B.: Human Immunodeficiency Virus Type 1 Escape from RNA Interference. *J. Virol.*, 77, 11531–11535 (2003).

[Braasch *et al.*, 2003] Braasch, D.A., Jensen, S., Liu, Y., Kaur, K., Arar,K., White, M.A., Corey, D. R.: RNA Interference in Mammalian Cells by Chemically-Modified RNA. *Biochemistry*, 42, 7967–7975 (2003)

[Brown *et al.*, 2005] Brown, K. M., Chu, C. Y., Rana, T. M.: Target accessibility dictates the potency of human RISC. *Nat. Struct. Mol. Biol.*, 12, 469–470 (2005).

[Buhler *et al.*, 2007] Buhler, M., Moazed, D.: Transcription and RNAi in heterochromatic gene silencing. *Nat. Struct. Mol. Biol.*, 14, 1041–1048 (2007).

[Chalk *et al.*, 2004] Chalk, A.M., Wahlestedt, C., Sonnhammer, E.L.L.: Improved and automated prediction of effective siRNA. *Biochem Biophys Res Commun.*, 319, 264–274 (2004).

[Chang *et al.*, 2012] Chang, P.C., Pan, W.J., Chen, C.W., Chen, Y.T., Chu DEsi, Y.W.: A design engine of siRNA that integrates SVMs prediction and feature filters. *Biocatalysis and Agricultural Biotechnology*, 1, 129–134 (2012).

[Chatterjiee–Kishore *et al.*, 2005] Chatterjee–Kishore M., Miller C. P.: Exploring the sounds of silence: RNAi-mediated gene silencing for target identification and validation. *Drug Discov.*, 1559–1565 (2005).

[Chiu *et al.*, 2003] Chiu, Y. L., Rana,T. M.: siRNA function in RNAi: A chemical modification analysis. *RNA*, 9, 1034–1048 (2003).

[Christoph *et al.*, 2006] Christoph, T., Grunweller, A., Mika, J., Schafer, M. K., Wade, E. J., Weihe, E., Erdmann, V. A., Frank, R., Gillen, C., Kurreck, J.: Silencing of vanilloid receptor TRPV1 by RNAi reduces neuropathic and visceral pain in vivo. *Biochem. Biophys. Res. Commun.*, 350, 238–243 (2006).

[Chuang *et al.*, 2000] Chuang, C. F., Meyerowitz, E. M.: Specific and heritable genetic interference by double-stranded RNA in Arabidopsis thaliana. *Proc. Natl. Acad. Sci.*, 97, 4985–4990 (2000).

[Clemens *et al.*, 1997] Clemens, M.J.,Elia, A.: The mRNA of the translationally controlled tumor protein P23/TCTP is a highly structured RNA, which activates the dsRNA-dependent protein kinase PKR. *J. Interferon Cytokine Res.*, 17, 503–524 (1997).

[Corey *et al.*, 2007] Corey, D. R.: RNAi learns from antisense. *Nat. Chem. Biol.*, 3, 8–11 (2007).

[Crooke *et al.*, 2004] Crooke, S. T.: Progress in Antisense Technology. *Annu. Rev. Med.*, 55, 61–95 (2004).

[Du *et al.*, 2005] Du Q, Thonberg H, Wang J, Wahlestedt C, Liang Z.: A systematic analysis of the silencing effects of an active siRNA at all single–nucleotide mismatched target sites. *Nucleic Acids Res.*, 33(5):1671-7 (2005).

[Elbashir *et al.*, 2001] Elbashir, S.M., Lendeckel, W., Tuschl, T.: RNA interference is mediated by 21– and 22–nucleotide RNAs. *Genes Dev.*, 15, 188–200 (2001).

[Elbashir *et al.*, 2001] Elbashir, S. M., Harborth, J., Lendeckel, W., Yalcin, A., Weber, K.,Tuschl, T.: Duplexes of 21-nucleotide RNAs mediate RNA interference in cultured mammalian cells. *Nature*, 411, 494–498 (2001).

[Elbashir *et al.*, 2001] Elbashir, S.M., Martinez, J., Patkaniowska, A., Lendeckel, W., Tuschl, T.: Functional anatomy of siRNAs for mediating efficient RNAi in Drosophila melanogaster embryo lysate. *EMBO J.*, 20, 6877–6888 (2001).

[Elbashir *et al.*, 2002] Elbashir, S.M., Harborth, J., Weber, K., Tuschl, T.: Analysis of gene function in somatic mammalian cells using small interfering RNAs. *Methods*, 26, 199–213 (2002).

[Elmen *et al.*, 2008] Elmen, J., Lindow, M., Schutz, S., Lawrence, M., Petri, A., Obad, S., Lindholm, M., Hedtjarn, M., Hansen, H. F., Berger, U., Gullans, S., Kearney, P., Sarnow, P., Straarup, E. M., Kauppinen, S.: LNA-mediated microRNA silencing in non-human primates. *Nature*, 452, 896–899 (2008).

[Escobar *et al.*, 2001] Escobar, M. A., Civerolo, E. L., Summerfelt, K. R., Dandekar, A. M.: RNAi-mediated oncogene silencing confers resistance to crown gall tumorigenesis. *Proc. Natl. Acad. Sci.*, 98, 13437–13442 (2001).

[Francesco *et al.*, 2001] Francesco, D. S., Hanspeter, S. , Alejandro, L., Cornia, T., Estelle, B., Frederick, M.: Sense and anti sense mediated gene silencing in tobacco is inhibited by the same viral suppressors and is associated with accumulation of small RNAs. *Proc. Natl. Acad. Sci.*, 96, 6506–6510 (2001).

[Gitlin *et al.*, 2005] Gitlin, L., Stone, J. K., Andino, R.: Poliovirus Escape from RNA Interference: Short Interfering RNA-Target Recognition and Implications for Therapeutic Approaches. *J. Virol.*, 79, 1027–1035 (2005).

[Gong *et al.*, 2006] Gong, W., Ren, Y., Xu, Q., Wang, Y., Lin, D., Zhou H. and Li, T. Integrated siRNA design based on surveying of features associated with high RNAi effectiveness. *BMC Bioinformatics*, 7, 516 (2006).

[Grunweller *et al.*, 2003] Grunweller, A.,Wyszko, E., Bieber, B., Jahnel, R., Erdmann, V. A. , Kurreck, J.: Comparison of different antisense strategies in mammalian cells using locked nucleic acids, 2'–O–methyl RNA, phosphorothioates and small interfering RNA. *Nucleic Acids Res.*, 31, 3185–3193 (2003).

[Grunweller *et al.*, 2003] Grunweller, A., Gillen, C., Erdmann, V.A., Kurreck, J.: Cellular Uptake and Localization of a Cy3-Labeled siRNA Specific for the Serine/Threonine Kinase Pim-1. *Oligonucleotides*, 13, 345–352 (2003).

[Haasnoot *et al.*, 2007] Haasnoot, J. Westerhout, E. M., Berkhout, B.: Review: RNA interference against viruses: strike and counterstrike. *Nat. Biotechnol.*, 25, 1435–1443 (2007).

[Harborth *et al.*, 2003] Harborth, J., Elbashir, S. M., Vandenburgh, K., Manninga, H., Scaringe, S. A., Weber, K., Tuschl, T.: Sequence, Chemical, and Structural Variation of Small Interfering RNAs and Short Hairpin RNAs and the Effect on Mammalian Gene Silencing. *Antisense Nucleic Acid Drug Dev.*, 13, 83–105 (2003).

[Ho *et al.*, 2003] Ho, T.B., Nguyen, D.D.: Chance Discovery and Learning Minority Classes. *Journal of New Generation Computing*, 21(2), 147–160 (2003).

[Holen *et al.*, 2006] Holen, T., Amarzguioui, M., Wiiger, M.T., Babaie, E., Prydz, H.: Positional effects of short interfering RNAs targeting the human coagulation trigger Tissue Factor. *Nucleic Acids Res.*, 30, 1757–1766 (2002).

[Hsieh *et al.*, 2004] Hsieh, A.C., Bo, R., Manola, J., Vazquez, F., Bare, O., Khvorova, A., Scaringe, S., Sellers, W.R.: A library of siRNA duplexes targeting the phosphoinositide 3-kinase pathway: determinants of gene silencing for use in cell-based screens. *Nucleic Acids Res.*, 32(3), 893–901 (2004).

[Huesken *et al.*, 2005] Huesken, D., Lange, J., Mickanin, C., Weiler, J., Asselbergs, F., Warner, J., Mellon, B., Engel, S., Rosenberg, A., Cohen, D., Labow, M., Reinhardt, M., Natt, F., Hall, J.: Design of a Genome–Wide siRNA Library Using an Artificial Neural Network. *Nature Biotechnology*, 23(8), 955–1001 (2005).

[Ichihara *et al.*, 2007] Ichihara, M., Murakumo, Y., Masuda, A., Matsuura, T., Asai, N., Jijiwa, M., Ishida, M., Shinmi, J., Yatsuya, H., Qiao, S. et al. Thermodynamic instability of siRNA duplex is a prerequisite for dependable prediction of siRNA activities. *Nucleic Acids Res.*, 35, e123 (2007).

[Ingelbrecht *et al.*, 1994] Ingelbrecht, I., Van Houdt, H., Van Montagu, M., Depicker, A.: Post-transcriptional silencing of reporter transgenes in tobacco correlates with DNA methylation. *Proc. Natl. Acad. Sci.* , 91,10502–10506 (1994).

[Jackson *et al.*, 2003] Jackson A.L., Bartz S.R., Schelter J., et al.: Expression profiling reveals off-target gene regulation by RNAi. *Nature Biotechnol.*, 21, 635–637(2003).

[Jackson *et al.*, 2006] Jackson A.L., Burchard J., Leake D., et al.: Position–specific chemical modification of siRNAs reduces "off–target" transcript silencing. *RNA*, 12, 1197-1205 (2006).

[Jagla *et al.*, 2005] Jagla, B., Aulner, N., Kelly, P.D., Song, D., Volchuk, A., Zatorski, A., Shum, D., Mayer, T., De Angelis, D.A., Ouerfelli, O., Rutishauser, U., Rothman, J.E.: Sequence characteristics of functional siRNAs. *Rna 2005*, 11(6), 864–872 (2005).

[Karol K. *et al.*, 2010] Karol K., Gabor C.: Kernel Based Off–Target Analysis of Rnai Experiments Global. *Journal of Medical Research*, Vol. 1, Issue 1, Ver 1.0, (2010).

[Klingelhoefer *et al.*, 2009] Klingelhoefer, J.W., Moutsianas, L., and Holmes, C.C. Approximate Bayesian feature selection on a large meta-dataset offers novel insights on factors that effect siRNA potency. *Bioinformatics*, 25, 1594–1601(2009).

[Komarov *et al.*, 1999] Komarov, P. G., Komarova, E. A., Kondratov, R. V., Christov–Tselkov, K., Coon, J. S., Chernov, M. V., Gudkov, A. V.: A Chemical Inhibitor of p53 That Protects Mice from the Side Effects of Cancer Therapy. *Science*, 285, 1733–1737 (1999).

[Kooter *et al.*, 1999] Kooter, J. M., Matzke, M. A., Meyer, P.: Listening to silent gene: transgene silencing, gene regulation and pathogen control. *Trends Plant Sci.*, 4, 340–347 (1999).

[Kretschmer *et al.*, 2003] Kretschmer–Kazemi Far, R., Sczakiel, G.: The activity of siRNA in mammalian cells is related to structural target accessibility: a comparison with antisense oligonucleotides. *Nucleic Acids Res.*, 31, 4417–4424 (2003).

[Kurreck 2006] Kurreck, J.: Review. *J. Biomed. Biotechnol.*, 83, 757 (2006).

[Kurreck , 2009] Kurreck, J., RNA interference: from basic research to therapeutic applications. *Angew. Chem.*, 121, 1404–1426 (2009).

[Krueger *et al.*, 2007] Krueger, U., Bergauer, T.,Kaufmann, B., Wolter, I., Pilk, S., Heider-Fabian, M., Kirch, S.,Artz–Oppitz, C., Isselhorst, M., Konrad, J.: Insights into Effective RNAi Gained from Large-Scale siRNA Validation Screening. *Oligonucleotides*, 17, 237–250 (2007).

[Ladunga *et al.*, 2007] Ladunga, I.: More complete gene silencing by fewer siRNAs: Transparent optimized design and biophysical signature. *Nucleic Acids Res.*, 35, 433 – 440 (2007).

[Lim *et al.*, 2005] Lim L., Lau N., Garrett–Engele P. et al.: Microarray analysis shows that some microRNAs downregulate large numbers of target mRNAs. *Nature*, 433,769–773 (2005).

[Liu *et al.*, 2004] Liu J., Carmell, M.A.,Rivas F.V., Marsden, C.G.,Thomson, J.Ms., Song, J.J., Hammond, S.M., Joshua–Tor, L., Hannon, G.J.: Argonaute2 is the catalytic engine of mammalian RNAi. *Science*, 305, 1437–1441 (2004).

[Mysara *et al.*, 2012] Mysara, M., Elhefnawi, M. and Garibaldi, JM. MysiRNA: improving siRNA efficacy prediction using a machine-learning model combining multi-tools and whole stacking energy. *J Biomed Inform.*, 45, 528-34 (2012).

[1] Nat. Cell Biol., 5, 489–490 (2003).

[Napoli *et al.*, 1990] Napoli, C., Lemieux, C., Jorgensen, R.: Introduction of chimeric chalcone synthase gene into Petunia results in reversible cosuppression of homologous genes in trans. *Plant Cell*, 2, 279–289 (1990).

[Overhoff *et al.*, 2005] Overhoff, M., Alken, M., Far, R. K., Lemaitre, M., Lebleu, B., Sczakiel, G., Robbins, I.: Local RNA Target Structure Influences siRNA Efficacy: A Systematic Global Analysis. *J. Mol. Biol.*, 348, 871– 881(2005).

[Pai *et al.*, 2006] Pai, S. I., Lin, Y. Y., Macaes, B., Meneshian, A., Hung, C. F., Wu, T. C.: Prospects of RNA interference therapy for cancer. *Gene Ther.*, 13, 464 – 477 (2006).

[Pei *et al.*, 2006] Pei, Y., Tuschl, T.: On the art of identifying effective and specific siRNAs. *Nat Methods* 3, 670–676 (2006).

[Qiu *et al.*, 2009] Qiu, S. and Lane, T. A Framework for Multiple Kernel Support Vector Regression and Its Applications to siRNA Efficacy Prediction. *IEEE/ACM Trans. Comput. Biology Bioinform.*, 6, 190–199 (2009).

[Ren *et al.*, 2006] Ren, Y., Gong, W., Xu, Q., Zheng, X., Lin, D. and et al. siRecords: an extensive database of mammalian siRNAs with efficacy ratings. *Bioinformatics*, 22, 1027–1028 (2006).

[Reynolds *et al.*, 2004] Reynolds, A., Leake, D., Boese, Q., Scaringe, S., Marshall, W.S., Khvorova, A.: Rational siRNA design for RNA interference. *Nat Biotechnol.*, 22(3), 326–330 (2004).

[Santel *et al.*, 2006] Santel, A., Aleku, M., Keil, O., Endruschat, J., Esche, V., Durieux, B., Loffler, K., Fechtner, M., Rohl, T., Fisch, G., Dames, S., Arnold, W., Giese, K., Klippel, A., Kaufmann, J.: RNA interference in the mouse vascular endothelium by systemic administration of siRNA-lipoplexes for cancer therapy. *Gene Ther.*, 13, 1360–1370 (2006).

[Sciabola *et al.*, 2013] Sciabola, S., Cao, Q., Orozco, M., Faustino, I. and Stanton, R.V. Improved nucleic acid descriptors for siRNA efficacy prediction. *Nucl. Acids Res.*, 41, 1383–1394 (2013).

[Scherer *et al.*, 2003] Scherer, L.J., Rossi, J.J.: Approaches for the sequence-specific knockdown of mRNA. *Nat Biotechnol.*, 21, 1457–465 (2003).

[Schubert *et al.*, 2004] Schubert, S., Kurreck, J.: Human Gene Therapy. *Curr. Drug Targets*, 5, 667–681 (2004).

[Sen *et al.*, 2005] Sen, G. L., Blau, H. M.: Argonaute 2/RISC resides in sites of mammalian mRNA decay known as cytoplasmic bodies. *Nat. Cell Biol.*, 7, 633–636 (2005).

[Shen *et al.*, 2006] Shen, J., Samul, R., Silva, R. L., Akiyama, H., Liu, H., Saishin, Y., Hackett, S.F., Zinnen, S., Kossen, K., Fosnaugh, K., Vargeese, C., Gomez, A., Bouhana, K., Aitchison, R., Pavco, P., Campochiaro, P. A.: Suppression of ocular neovascularization with siRNA targeting VEGF receptor 1. *Gene Ther.*, 13, 225–234 (2006).

[Shabalina *et al.*, 2006] Shabalina, S.A., Spiridonov, A.N., Ogurtsov, A.Y. Computational models with thermodynamic and composition features improve siRNA design. *BMC Bioinformatics*, 7, 65 (2006).

[Smith *et al.*, 2008] Smith, F. J., Hickerson, R. P., Sayers, J. M., Reeves, R. E., Contag, C. H., Leake, D., Kaspar, R. L., McLean, W. H.: Development of Therapeutic siRNAs for Pachyonychia Congenita. *J. Invest. Dermatol.*, 128, 50–58 (2008).

[Smith *et al.*, 2000] Smith, N. A., Singh, S. P., Wang, M. B., Stoutjesdijk, P. A., Green, A. G., Waterhouse, P. M.: Total silencing by intron spliced hairpin RNA. *Nature*, 407, 319–320 (2000).

[Uitei *et al.*, 2004] Ui-Tei, K., Naito, Y., Takahashi, F., Haraguchi, T., Ohki-Hamazaki, H., Juni, A., Ueda, R., Saigo, K.: Guidelines for the selection of highly effective siRNA sequences for mammalian and chick RNA interference. *Nucleic Acids Res.*, 32, 936–948 (2004).

[Van Blokland *et al.*, 1994] Van Blokland, R., Vander Geest, N., Mol, J. N. M., Kooter, J. M.: Transgene-mediated suppression of chalcone synthase expression in Petu- nia hybrida results from an increase in RNA turnover. *Plant J.*, 6, 861–877 (1994).

[Vander *et al.*, 1990] Vander Krol, A. R., Mur, L. A., Beld, M., Mol, J. N. M.,Stuitje, A. R.: Flavanoid genes in petunia: addition of limited number of gene copies may lead to a suppression of gene expression. *Plant Cell*, 2, 291–299 (1990).

[Vert *et al.*, 2006] Vert, J.P., Foveau, N., Lajaunie, C., Vandenbrouck, Y. An accurate and interpretable model for siRNA efficacy prediction. *BMC Bioinformatics*, 7, 520 (2006).

[Vickers *et al.*, 2003] Vickers, T.A., Koo, S., Bennett, C.F., Crooke, S.T., Dean, N.M. and Baker, B.F. Efficient reduction of target RNAs by small interfering RNA and RNase H-dependent antisense agents. A comparative analysis. *J. Biol. Chem.*, 278, 7108–7118 (2003).

[Takasaki *et al.*, 2010] Takasaki, S.: Efficient prediction methods for selecting effective siRNA sequences. *Comput Biol Med.*, 40, 149–158 (2010).

[Takasaki *et al.*, 2013] Takasaki, S.: Methods for Selecting Effective siRNA Target Sequences Using a Variety of Statistical and Analytical Techniques. *Methods Mol Biol.*, 942, 17–55 (2013).

[Teramoto *et al.*, 2005] Teramoto, R., Aoki, M., Kimura, T., Kanaoka, M.: Prediction of siRNA functionality using generalized string kernel and support vector machine. *FEBS Lett.*, 579, 2878–2882 (2005).

[Warnecke *et al.*, 2004] Warnecke, C., Zaborowska, Z., Kurreck, J., Erdmann, V. A., Frei, U., Wiesener, M., Eckardt, K. U.: Differentiating the functional role of hypoxia-inducible factor (HIF)-1alpha and HIF-2alpha (EPAS-1) by the use of RNA interference: erythropoietin is a HIF-2alpha target gene in Hep3B and Kelly cells. *FASEB J.*, 18, 1462–1464 (2004).

[Watanabe *et al.*, 2004] Watanabe, A., Arai, M., Yamazaki, M., Koitabashi, N., Wuytack, F., Kurabayashi, M.: Phospholamban ablation by RNA interference increases Ca2+uptake into rat cardiac myocyte sarcoplasmic reticulum. *J. Mol. Cell. Cardiol.*, 37, 691–698 (2004).

[Weitzer *et al.*, 2007] Weitzer S1, Martinez J.: The human RNA kinase hClp1 is active on 3' transfer RNA exons and short interfering RNAs. *Nature*, 447, 222–226 (2007).

[Wu *et al.*, 2003] Wu, H., Hait, W. N., Yang, J. M.: Small interfering RNA-induced suppression of MDR1 (P-glycoprotein) restores sensitivity to multidrug–resistant cancer cells. *Cancer Res.*, 63, 1515–1519 (2003).

[Zimmermann *et al.*, 2006] Zimmermann, T. S., Lee, A. C., et al.: RNAi-mediated gene silencing in non-human primates. *Nature*, 441, 111–114 (2006).

Publications

[1] Bui, N.T., Ho, T.B., Kawasaki, S., A Sequential Apriori Algorithm for Discriminative Design Rules of Effective siRNA Sequences. *13th International Symposium on Knowledge and Systems Science*, 187-194 (2012).

[2] Ho, T.B., Takabayashi, T., Kanda, T., Kawasaki, S., Le, T.N., Bui, N.T., Than, Q.K, From Clinical to Genomics Data in Hepatitis Study. *The First Asian Conference on Information Systems*, (2012).

[3] Bui, N.T., Ho, T.B., Kawasaki, S., A Descriptive Method for Generating siRNA Design Rules. *The 5th Asian Conference On Intelligent Information and Database Systems*, 7803, 196-205 (2013).

[4] Bui Thang Ngoc, Ho Tu Bao, Nguyen Linh Vu, The prediction of siRNA knockdown efficacy. *RIVF workshop on computational biomedicine*, 23-32 (2013).

[5] Bui Thang, A Novel Framework to Improve siRNA Efficacy Prediction. *PAKDD*, 8444, 400-412 (2014).

[6] Bui Thang Ngoc, Tu Bao Ho, Tatsuo Kanda, A semi-supervised tensor regression model for siRNA efficacy prediction. *BMC Bioinformatics* (2015). (Under revision)