

String Kernels of Imperfect Matches for Off-target Detection in RNA Interference

Shibin Qiu and Terran Lane

Dept. of Computer Science, University of New Mexico,
Albuquerque, NM, 87131
{squi, terran}@cs.unm.edu

Abstract. RNA interference (RNAi) is a posttranscriptional gene silencing mechanism frequently used to study gene functions and knock down viral genes. RNAi has been regarded as a highly effective means of gene repression. However, an “off-target effect” deteriorates its specificity and applicability. The complete off-target effects can only be characterized by examining all factors through systematic investigation of each gene in a genome. However, this complete investigation is too expensive to conduct experimentally which motivates a computational study. The sequence matching between an siRNA and its target mRNA allows for mismatches, G-U wobbles, and the secondary structure bulges, in addition to exact matches. To simulate these matching features, we propose string kernels measuring the similarity between two oligonucleotides and develop novel efficient implementations for RNAi off-target detection. We apply the algorithms for off-target errors in *C. elegans* and human.

1 Introduction

In addition to its use as a powerful tool to study gene functions and knockdown viral genes, RNA interference (RNAi) [1] also has the potential for validating drug design and treating diseases. RNAi is initiated by short interfering RNA (siRNA) of about 21 nt in length, either generated from double stranded RNA (dsRNA) by the enzyme Dicer, or delivered experimentally. Target mRNA transcripts that are matched with the siRNA are destroyed by the silencing complex RISC. RNAi has been regarded as a highly effective means of gene repression [2]. However, the effectiveness of RNAi can be compromised by the off-target knockdown, also known as nonspecific, cross-reactive, and false positive knockdown, which is defined as gene silencing inadvertently induced when a true target gene is intended to be knocked down. Characterizing the specificity of RNAi is critical to fully applying this novel technology to therapeutics [3, 4].

Examinations of RNAi off-target effect have yielded mixed results. The predominant opinion is that RNAi is highly specific to its target [2, 5]. But significant non-specific gene knockdowns have also been reported [6, 7]. In addition, some RNAi applications did not evaluate off-target effects [8]. To thoroughly understand RNAi specificity, off-target effect of silencing each gene should be

evaluated in a genome by considering multiple factors. However, it is expensive to conduct this comprehensive study experimentally. Computational approach is inexpensive to implement and able to extend experimental parameters into wider ranges and simulate what is impossible under normal biological conditions.

If an RNAi experiment assesses its specificity, the recommended procedure is using BLAST [9] searches to find off-target genes [10]. The sequence binding between an siRNA and its target mRNA allows for mismatches, G-U wobbles, and the secondary structure bulges [7]. Though BLAST allows for deletion, insertion and mismatch based on alignment, it cannot control exact patterns of the imperfect matches, such as their positions and lengths. To simulate siRNA-target matching, we develop string kernels that accurately generate matching patterns by controlling the pattern length and position.

Related to RNAi, computational methods exist to predict microRNA genes and targets [11, 12], putative RNAi [13], and siRNA efficacy [14]. To our knowledge, related works have not studied RNAi specificity. Our string kernels are the first of their kind to simulate siRNA and target matching that allows for accurate pattern control. We measure the similarity between two oligonucleotides with inner products in the feature space, similar to [15]. Leslie et al. [15] constructed a mismatch tree for computing their mismatch string kernel for a support vector machine classifier to detect protein families. But this mismatch kernel did not control mismatch positions. Amir et al. [16] developed an algorithm for string searches with single letter mismatch, which is not enough for RNAi.

We first define off-target error rate using information retrieval theory. We then propose novel string kernels to simulate siRNA and its target binding and develop efficient implementations for genome-wide computation. We study the dependencies of non-specificity on factors derived from RNAi experiments including dsRNA length, siRNA length, lengths and positions of imperfect mismatches. Experiments are performed on *C. elegans* and human genomes.

2 String Kernels and Their Implementations

We first define the exact match string kernel and the off-target error rate. We then present the imperfect match kernels and their implementations.

Sequence matching between an siRNA from a given gene and another endogenous gene signifies an occurrence of off-target knockdown, which is a typical setting for gene function analysis [8]. For viral gene silencing, we conceptually combine the viral gene into the host genome and use the same setting.

2.1 Matching siRNA and Target with the Exact Match String Kernel

We describe each gene by its possible contiguous subsequences of length n (17-28 nt), called n -mers, or n -grams, representing siRNAs. Gene g_x in the input space \mathcal{X} , consisting of sequences drawn from the alphabet $\mathcal{A} = \{a, c, g, t\}$, is mapped into an n -gram feature space \mathbb{R}^{4^n} by the feature map of exact match,

$$\Phi^{ex}(g_x) = (\phi_a(g_x))_{a \in \mathcal{A}^n}, \quad (1)$$

where $\phi_a(g_x)$ is the number of times n-gram a occurs in g_x . Therefore, the image of g_x is the coordinates in the feature space indexed by the number of occurrences of its constituent n-mers. A gene g_y is said to match g_x if the following is true,

$$K(g_x, g_y) = \langle \Phi^{ex}(g_x), \Phi^{ex}(g_y) \rangle \geq T, \tag{2}$$

for a threshold T . The similarity measure $K(g_x, g_y) = \langle \Phi^{ex}(g_x), \Phi^{ex}(g_y) \rangle$ in (2) defines a kernel as used for a support vector machine classifier [17]. We use the kernel to match an siRNA and its target, instead of classifying. Since any match between an siRNA and an mRNA will silence the gene, we choose $T = 1$.

Implementation of the Exact Match Kernel. Computing the similarity of (2) directly in a vector space requires $O(DF4^n)$ time, where F is the number of n-grams in the genome (40×10^6 for *C. elegans* and 60×10^6 for human) and D (close to F) is the amount of n-mers to be compared in the coding sequences. For genome-wide scan, this computing time is prohibitive and can be improved since the the feature vectors are sparse. We use an *inverted file* where the n-mers serve as identifiers and their gene names and positions within the genes serve as attributes. We use the positions for imperfect matches later. If we ignore n-mers having zero occurrence and allow for duplicate n-mers, a gene g_x can be represented in the feature space compactly,

$$\Phi^{ex}(g_x) = \{(a_1, p_1), (a_2, p_2), \dots, (a_{k_x}, p_{k_x})\}, \tag{3}$$

where $a_j, 1 \leq j \leq k_x$, is the j th n-gram of g_x , p_j is its position on g_x , and k_x is the number of n-mers in g_x . In the inverted file, the records for g_x contains the triples $\langle a_1, g_x, p_1 \rangle, \langle a_2, g_x, p_2 \rangle, \langle a_3, g_x, p_3 \rangle, \dots, \langle a_{k_x}, g_x, p_{k_x} \rangle$. The inverted file for a genome is the collection of the triples of its genes. To speed up computation, we sort the inverted file on the n-mer field using a binary search tree (BST).

$K(g_x, g_y)$ in (2) is computed by searching each n-mer of g_x for g_y in the inverted file. $K(g_x, g_y)$ is the number of occurrences of g_y among the matched genes. Each search in the BST takes $O(\log F)$ time, resulting in a time of $O(k_x \log F)$ for computing $K(g_x, g_y)$.

2.2 Definition of Off-target Error Rate

We define the off-target error using the exact match kernel, but it is the same for other kernels. To simulate Dicer’s cleavage of dsRNA (100-400bp) into siRNAs, we take an oligonucleotide o_x , as dsRNA, from gene g_x and map it into the feature space using the exact match kernel. Expressed compactly as in (3), $\Phi^{ex}(o_x) = \{(s_1, p_1), (s_2, p_2) \dots (s_{l_x}, p_{l_x})\}$, where s_j is the j th n-mer in o_x . To obtain the matched genes based on (2), we compute the kernel $K(o_x, g_y)$ for each gene g_y , for $1 \leq y \leq G$, where G is the total number of genes in a genome. These calculations use BST searches, as described previously.

Let $C_x = \{g_{x_1}, g_{x_2}, \dots\}$ be the set of genes whose kernel values with o_x satisfies (2), excluding g_x itself. The *precision* of a search is the proportion of correct documents to the total number of documents matched. Here, only g_x is correct and the number of genes returned is $1 + |C_x|$. So the precision of g_x is $P_x =$

$1/(1+|C_x|)$. We define the *off-target error* of g_x as $E_x = 1 - P_x = |C_x|/(1+|C_x|)$. We define the average error rate as,

$$E(\Theta) = \sum_{i=1}^G E_i/G, \quad (4)$$

where E_i is the error for silencing gene g_i . And $\Theta = \langle l, n, m, p, b, q, w, r \rangle$ is the set of parameters, where l is the dsRNA length; n , siRNA length; m , mismatch length; p , position of the mismatch; b and w are lengths of the bulge and wobble; and q and r are positions of the bulge and the wobble, respectively. Figure 1 shows some of the parameters. The average number of incorrect genes targeted by silencing each gene in the genome is, $Z(\Theta) = E(\Theta)/(1 - E(\Theta))$.



Fig. 1. Exact match (left), mismatch (and wobble, middle), and bulge (right)

2.3 The Feature Maps of Mismatch, Bulge, and Wobble

For an n -mer a from an alphabet \mathcal{A} , define its mismatch neighborhood $N_{m,p}^{mis}(a)$ as all n -mer γ from \mathcal{A} that differ from a by at most m contiguous mismatches starting at position p in a , and $\Phi_{m,p}^{mis}(a) = (\phi_\gamma(a))_{\gamma \in \mathcal{A}^n}$, where $\phi_\gamma(a) = 1$ if $\gamma \in N_{m,p}^{mis}(a)$, and $\phi_\gamma(a) = 0$, otherwise. The feature map of a gene g_x is defined as the sum of the feature maps of its n -mers,

$$\Phi_{m,p}^{mis}(g_x) = \sum_{a \in g_x} \Phi_{m,p}^{mis}(a). \quad (5)$$

The bulge neighborhood $N_{b,q}^{bulge}(a)$ for n -mer a is defined as all $(n+b)$ -mers γ from the target that match a exactly everywhere except by a bulge of b nt long starting at position q on γ , and $\Phi_{b,q}^{bulge}(a) = (\phi_\gamma(a))_{\gamma \in \mathcal{A}^{n+b}}$, where $\phi_\gamma(a) = 1$ if $\gamma \in N_{b,q}^{bulge}(a)$, and $\phi_\gamma(a) = 0$, otherwise. The feature map of g_x is defined as the sum of the feature maps of its n -mers,

$$\Phi_{b,q}^{bulge}(g_x) = \sum_{a \in g_x} \Phi_{b,q}^{bulge}(a). \quad (6)$$

The wobble feature map $\Phi_{w,r}^{wobble}(\cdot)$ is defined similarly to $\Phi_{m,p}^{mis}(\cdot)$, except only G-U wobble is allowed in its neighborhood.

Note the mismatch, bulge and wobble neighborhoods are supersets of the exact match region. By defining the similarity neighborhood for the combination of mismatches, bulges, and wobbles as the union of the neighborhoods defined above, we can define the feature map of simultaneous mismatch, bulge and wobble $\Phi_{m,p,b,q,w,r}^{mbw}(\cdot)$ accordingly. The string kernels are derived using inner products based on the above feature maps, as in (2).

2.4 Implementations of Mismatch, Bulge, and Wobble Kernels

Let $S = \{s_1, s_2 \dots s_N\}$ be a set of strings of length k from an alphabet \mathcal{A} . Suppose string $s_i = a_1 a_2 \dots a_k$, where $a_j \in \mathcal{A}$, $1 \leq j \leq k$, has reverse string $\overline{s_i} = a_k a_{k-1} \dots a_1$.

Definition A *mirrored tree* of a binary search tree (BST) populated with strings from S is the BST populated with reverse strings $\overline{s_1}, \overline{s_2}, \dots, \overline{s_N}$. A *u leading range* of a string s from S searched in a BST is the set of nodes returned by a search that only matches the beginning u letters of s .

Algorithm 1. Mirrored Tree Search, MTS(n, m, p)

- 1: Build BST T_1 and mirrored BST T_2 for the inverted file using n -mer
 - 2: **for each** gene g_i in the genome **do**
 - 3: Take a subsequence d_i in g_i
 - 4: **for each** n -mer s_j^i in d_i **do**
 - 5: Get R_1 , the $p - 1$ leading range of s_j^i from T_1
 - 6: Get R_2 , the $n - m - p + 1$ leading range of $\overline{s_j^i}$ from T_2
 - 7: Find $C_j^i = R_1 \cap R_2$
 - 8: **end for**
 - 9: Calculate off-target error E_i for g_i using $C_i = \bigcup_j C_j^i$
 - 10: **end for**
 - 11: Calculate average off-target error for the genome
-

The mismatch kernel in (5) can be computed by the mirrored tree search (MTS) in Algorithm 1, whose correctness proof we omit due to space limitation. At step 5 and 6, the substring before the mismatch is exact-matched in T_1 and the leading range is stored in R_1 , the substring after the mismatch is exact-matched in T_2 and the leading range is stored in R_2 . The genes corresponding to the mismatch letters are sandwiched in C at step 7 by the intersection based on gene names and positions of the n -mers. At step 9, E_i is computed using C_i .

Let the size of the inverted file be F and the total number of n -grams from all dsRNAs be D . MTS has a cost of $T_{mt} = O(D(2 \log F + C))$, where $O(2 \log F)$ is the search time in the BSTs, and C is the cost of obtaining the leading ranges and the intersection. Using proper join algorithms, C can be bounded by $O(|R_1| + |R_2|)$ [18]. Empirically, C is small and treated as a constant. The mismatch tree algorithm [15] has a complexity of $O(DLn^m 4^m)$, where L is the average length of genes. Since usually $F < 10^8$ and $\log F < 30$, MTS is much faster. A straightforward way of mismatch search—done by searching each variant of the n -mer once—needs 4^m exact-match searches in the BST and consumes a time of $T_{sf} = O(4^m D \log F)$. MTS’s speedup over the straightforward method is roughly, $Sp \approx O(4^m)$. Empirical results demonstrated that MTS achieved speedups of 2 orders of magnitude on average for the 2 organisms. However, it uses more space because of the mirrored tree.

The wobble kernel can be implemented by modifying Algorithm 1 to allow only G-U wobbles in step 7. The bulge kernel can also be computed by modifying

MTS. At step 1, use $(n + b)$ -mer on the target; at step 4, elongate s_i and \bar{s}_i to length $n + b$ by appending b arbitrary letters; get $q - 1$ leading range R_1 from T_1 at step 5; get $n + b - q + 1$ leading range R_2 from T_2 at step 6. Thus R_1 contains the genes corresponding to the exact matches on the first $q - 1$ letters, leaving $n + b - (q - 1)$ letters unconstrained in T_1 . And R_2 contains the genes corresponding to exact matches on the last $n - q + 1$ letters, leaving $b + q - 1$ letters unconstrained in T_2 . The intersection yields bulge matches of the $(n + b)$ -mer on the target and n -mer of the siRNA. The combination of imperfect matches can also be computed using MTS. These implementations are efficient and flexible, allowing the imperfect match patterns to occur anywhere in the n -grams.

3 Experiments

The cDNA sequences of 22,168 genes of *C. elegans* (release WS110) were obtained from Sanger Institute. The human genes representing 27,852 mRNAs were taken from the RefSeq database at NCBI.

We show off-target error rates using an exact match in Fig. 2(a) (*H. sapiens*) and (b) (*C. elegans*). When siRNA length increases, the off-target error decreases. When an siRNA is short, a small length increase improves the error dramatically and the error rate curve is steep. But when the siRNA is relatively long, further length increase only improves the error slightly. The observed siRNA length in biological experiments is between 19 and 25 nt, which is consistent with our finding that increasing siRNA length beyond 25 does not improve specificity much. When siRNAs are short, longer dsRNAs generate larger errors. When they are relatively long, longer dsRNAs cause slight increases in the error. The error rate in human is higher than in *C. elegans*, but they present similar patterns. Therefore we focus on the *C. elegans* genome in the following. Fig. 2(c) plots the error rates when bulge, mismatch and wobble exist separately. Bulge barely increases off-target error. This is because the larger neighborhood caused by a bulge is cancelled by the reduced chances of target binding made by an increased length of the subsequence on the target to form the bulge. Wobble increases the error slightly, but its increase becomes insignificant for siRNAs longer than 21

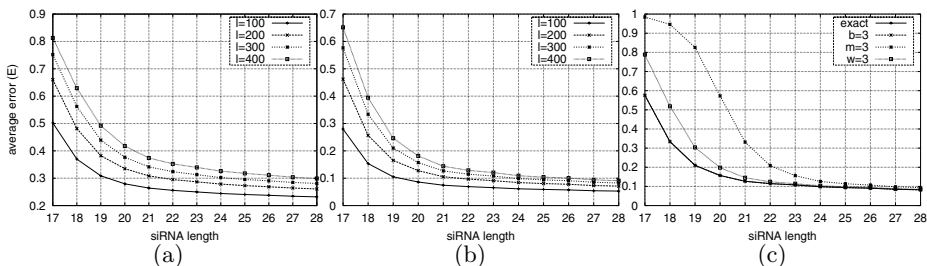


Fig. 2. Off-target error rates. (a) *H. sapiens*, exact match; (b) *C. elegans*, exact match; ($l = 100 - 400$ and $n = 17 - 28$). (c) *C. elegans* ($l=300$, $b = m = w = 3$)

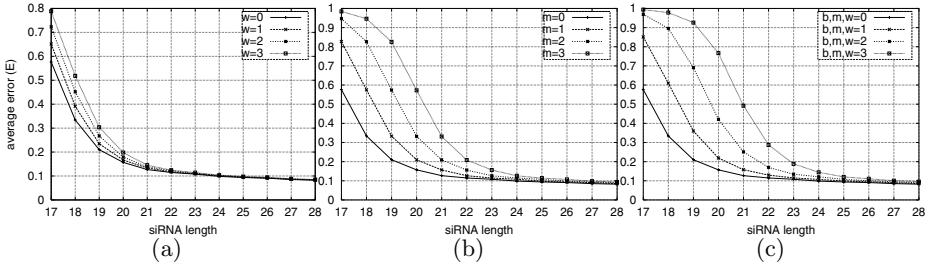


Fig. 3. Effects of imperfect matches in *C. elegans* ($l=300$). (a)wobble; (b)mismatch; (c) simultaneous mismatch, bulge, and wobble

nt. Mismatch caused a dramatic increase in error rate, but its effect diminished when $n \geq 25$. Results also show that positions of these imperfect matches did not change error rates. Therefore, position effects observed in biological experiments are due to thermodynamic properties of the sequences and RISC [7]. Fig. 3(a) and (b) show the effects of a range of m and w , suggesting, again, that mismatch is more critical than G-U wobble. Fig. 3(c) displays simultaneous mismatch, bulge, and wobble using a combined kernel, indicating that the increase in off-target errors did not diminish until $n \geq 25$.

$$\hat{E} = 18.87 - 0.58n + 0.22l + 51m + 10w - 0.008nl - 2nm - 0.4nw. \quad (7)$$

To predict off-target errors in *C. elegans*, we fitted a nonlinear regression function (7) using SAS software package, where $\hat{E} = \hat{E}(n, l, m, w, b)$ is in percents and length parameters are in nucleotides. As shown, n has a negative coefficient, consistent with off-target errors decreasing with n . The large coefficient with the m term shows that mismatch raises the error remarkably. The nonlinear terms are interactions of n with imperfect match parameters and also have negative coefficients, demonstrating the dominance of n over other parameters. Given a range of parameters, predictions using this function provide estimates of the off-target error rates and save computational effort.

4 Conclusions

Because the specificity of RNA interference is of fundamental importance and mixed results have been reported, we conducted a computational study of the off-target effects. To simulate the matching between an siRNA and its target mRNA, we introduced string kernels of imperfect matches including mismatches, bulges, and wobbles that are able to control the positions and lengths of the patterns. To improve computational performance for genome-wide scans, we proposed efficient implementations for the string kernels, which sped up computation

substantially. Results in *C. elegans* and human genomes indicated that off-target errors were substantial (5% to 80%). We found that bulge did not significantly increase off-target error, wobble raised the error slightly, and mismatch elevated the error dramatically. But the increase in error rates all diminished when siRNAs are long enough. Computationally, positions of the imperfect matches did not change off-target errors. Therefore, position-effects are due to thermodynamic properties of the components. A nonlinear regression function was fitted to predict off-target error rates. In summary, the off-target effect presents a real but not prohibitive issue and should be controlled in RNAi experiments.

Acknowledgement

This work is supported by NIH under grant number P20RR18754 from the Institutional Development Award Program of the National Center for Research Resource. We thank Coenraad M. Adema for his constructive suggestions.

References

1. Fire, A., Xu, S.Q., et al.: Potent and specific genetic interference by double-stranded RNA in *Caenorhabditis elegans*. *Nature* **391** (1998) 806–811
2. Elbashir, S.M., Martinez, J., et al.: Functional anatomy of siRNA for mediating efficient RNAi in *Drosophila melanogaster* embryo lysate. *The EMBO Journal* **20** (2001) 6877–6888
3. Dillin, A.: The specifics of small interfering RNA specificity. *Proc. Nat. Acad. Sci. USA* **100** (2003) 6289–6291
4. Check, E.: Hopes rise for RNA therapy as mouse study hits target. *Nature* **432** (2004) 136
5. Tuschl, T., Zamore, P.D., et al.: Targeted mRNA degradation by double-stranded RNA *in vitro*. *Genes Dev.* **13** (1999) 3191–3197
6. Jackson, A.L., Bartz, S.R., et al.: Expression profiling reveals off-target gene regulation by RNAi. *Nature Biotechnology* **21** (2003) 635–637
7. Saxena, S., Jonsson, Z.O., Dutta, A.: Small RNAs with imperfect match to endogenous mRNA repress translation. *Journal of Biological Chemistry* **278** (2003) 44312–44319
8. Kamath, R.S., Fraser, A.G., et al.: Systematic function analysis of the *C. elegans* genome using RNAi. *Nature* **421** (2003) 231–237
9. Altschul, S.F., et al.: Basic local alignment search tool. *J. Mol. Biol.* **215** (1990) 403–410
10. Elbashir, S.M., Harborth, J., et al.: Analysis of gene function in somatic mammalian cells using small interfering RNAs. *Methods* **26** (2002) 199–213
11. Lim, L.P., Glasner, M.E., et al.: Vertebrate microRNA genes. *Science* **299** (2003) 1540
12. Lewis, B.P., et al.: Prediction of mammalian microRNA targets. *Cell* **115** (2003) 787–798
13. Horesh, Y., Amir, A., et al.: A rapid method for detection of putative RNAi target genes in genomic data. *Bioinformatics* **19** (2003) ii73–ii80 Suppl. 2.

14. Chalk, A.M., Wahlestedt, C., Sonnhammer, E.L.: Improved and automated prediction of effective siRNA. *Biochemical and Biophysical Research Comm.* **319** (2004) 264–274
15. Leslie, C., Eskin, E., et al.: Mismatch string kernels for discriminative protein classification. *Bioinformatics* **1** (2003) 1–10
16. Amir, A., Landau, G., et al.: Text indexing and dictionary matching with one error. *J. Algorithms* **37** (2000) 309–325
17. Vapnik, V.N.: *Statistical Learning Theory*. Wiley, New York (1998)
18. Garcia-Molina, H., Ullman, J.D., Widom, J.D.: *Database Systems: The Complete Book*. Prentice Hall Inc., New Jersey (2002)