Splice Junction Identification using Long Short-Term Memory Neural Networks

Kevin Regan¹, Abolfazl Saghafi²*, and Zhijun Li¹

¹Department of Chemistry and Biochemistry, University of the Sciences, Philadelphia, PA, USA; ²Department of Mathematics, Physics and Statistics, University of the Sciences, Philadelphia, PA, USA

Abstract: Background: Splice junctions are the key to move from pre-messenger RNA to mature messenger RNA in many multi-exon genes due to alternative splicing. Since the percentage of multi-exon genes that undergo alternative splicing is very high, identifying splice junctions is an attractive research topic with important implications.

Objective: The aim of this paper is to develop a deep learning model capable of identifying splice junctions in RNA sequences using 13,666 unique sequences of primate RNA.

Methods: A Long Short-Term Memory (LSTM) Neural Network model is developed that classifies a given sequence as EI (Exon-Intron splice), IE (Intron-Exon splice), or N (No splice). The model is trained with groups of trinucleotides and its performance is tested using validation and test data to prevent bias.

Results: Model performance was measured using accuracy and f-score in test data. The finalized model achieved an average accuracy of 91.34% with an average f-score of 91.36% over 50 runs.

Conclusion: Comparisons show a highly competitive model to recent Convolutional Neural Network structures. The proposed LSTM model achieves the highest accuracy and f-score among published alternative LSTM structures.

Keywords: Splice junction, deep learning, neural networks, LSTM, RNA-seq, classification.

1. INTRODUCTION

Proteins make up many of the essential parts and processes of living organisms. The blueprint that determines the structure and function of proteins is coded in DNA, but not all parts of DNA are translated in protein synthesis. After transcription from DNA to pre-messenger RNA, RNA splicing occurs where sections of RNA, introns, are spliced out, and the remaining sections, exons, are joined together (Fig. 1). The boundaries separating the introns from the exons are called splice junctions. After the splicing process, a “mature messenger RNA” is formed and is ready to be translated by ribosomes into amino acids, the building blocks of proteins.

Identification of splice junctions is an important task in bioinformatics as they provide insight into the role of alternative splicing and increase the functional diversity of genes. Recent estimates suggest that more than 90% of multi-exon genes in the human body undergo alternative splicing [1], thus making the splice junction identification more crucial in predicting genes’ characteristics.

The importance of identifying splice junctions has attracted many researchers towards proposing new techniques. A panorama of the advantages and shortcomings of 15 different models and their scopes of application are provided by Ding et al. (2017), highlighting the importance of this problem and suggesting an ensemble method by joining multiple approaches to achieve higher accuracy of detection at the cost of increased computation time [2]. Moreover, the literature review suggests that not all techniques work equally well on detecting various types of splice junctions which in turn promotes the development of new techniques for the identification of splice sites.

A chronicle of recent advancements toward splice site identification is as follows. Mapleson et al. (2018) developed an RNA sequence mapping process called Portcullis reaching 98.17% f-score for detecting donor and acceptor sites using ~76 million simulated human training examples and some real data [3]. Zhang et al. (2018) developed DeepSplice, a customized Convolutional Neural Network to identify donor and acceptor splice sites from negative samples separately with high accuracy [4]. Zuullaert et al. (2018) proposed SpliceRover, a Convolutional Neural Network with interpretable visualization tools that could reach 96.12% (95.35%) accuracy in detecting acceptor (donor) sites [5].
Van Moerbeke et al. (2018) proposed a linear mixed model (REIDS) for the identification of alternative splicing events using Human Transcriptome Arrays (HTA). REIDS analytical framework detected between 65-77% of the validated exon probe sets [6]. Zhao et al. (2019) developed Assembling Splice Junctions Analysis (ASJA), a package that identifies splice junctions from high-throughput RNA sequencing data using a sophisticated three-step process [7]. Wang et al. (2019) developed the SpliceFinder package that utilizes a Convolutional Neural Network, achieving 90.25% accuracy in detecting splice sites [8].

Most recently, Lee et al. (2020) developed an LSTM model to predict the inclusion of a spliced exon based on adjacent epigenetic signals. Their method could achieve up to 86% f-core [9]. EDeepSSP [10], Splice2Deep [11], and InterSSPP [12] are other Convolutional Neural Network packages developed in 2020 that achieve a high degree of performance for double-binary classifications of the true acceptor (donor) sites from false acceptor (donor) sites trained on large datasets. It is important to note that increasing training examples increases the accuracy of well-defined deep learning models. Moreover, many of the abovementioned models are designed for double-binary classifications which have a bolster performance than multi-class problems. A summary of the cited methods with more details, including data size and overall results, is provided in Table 1.

In this article, an original method is developed to predict whether a given sequence of primate DNA contains a splice junction or not (N), and furthermore, to determine whether the splice junction is exon-intron (EI) or intron-exon (IE). This will be accomplished using a sophisticated classification technique utilizing Long Short-Term Memory Networks (LSTM). The outcome of this research is the development of an original classification technique for splice junction identification with high accuracy, fast implementation, easy interpretability, and less complexity compared to available alternatives.

Fig. (1). Pre-mRNA spliced into mature mRNA (figure obtained from https://www.khanacademy.org). (A higher resolution / colour version of this figure is available in the electronic copy of the article).

Fig. (2). Outline of the model development process. (A higher resolution / colour version of this figure is available in the electronic copy of the article).
2. MATERIALS AND METHODS

An open dataset of primate splice-junction gene sequences is utilized that contains 3,190 separate instances of primate data, each instance having 62 attributes [22]. The first attribute is the type of splice junction contained within the sequence with one of three possible values: EI (Exon-Intron), IE (Intron-Exon), and N (Neither). The second attribute contains the name of the specific instance. The last 60 attributes contain the full sequence of each instance, starting at position -30 and ending at position +30 relative to a splice junction if it exists.

Moreover, a second open dataset of Homo sapiens splice junction sequence is used that contains 10,676 unique sequences with 142 attributes [15]. Similar to the first dataset, the first attribute represents the type of splice junction (EI/IE/N), the second attribute is a unique name for the sequence, and the last 140 attributes contain the full sequence, starting at position -70 and ending at +70 relative to a splice junction if it exists. Values of each sequence contain either A, C, G, or T, representing the four possible nucleotides. In total, there are 3470 instances (25.39%) of class EI, 3,550 instances (25.98%) of class IE, and 6,646 instances (48.63%) of class N.

Fig. (2) illustrates the outline of the developed model that includes several steps. These steps are explained in detail below.

1. Data Cleaning. Originally, the first dataset included 3,190 cases [22]. After removing 15 cases with missing values and 186 duplicates, 2990 cases were used for processing. An additional 10,676 unique cases were added from the second Homo sapiens Splice Site Dataset [15]. In total, 13,666 cases with 3,470 instances (25.39%) of class EI, 3,550 instances (25.98%) of class IE, and 6,646 instances (48.63%) of class N were utilized for processing.

The length of sequences in the first and second dataset is, respectively, 60 and 140 letters, where each letter is either A, C, G, or T. The longer 140 letters sequences were cut to 60 letters with 30 nucleotides coming before the splice-junction, if present, and 30 coming after. All 60-letter sequences were then dispersed into 20 3-letter sequences. It is known that cells decode mRNAs by reading their nucleotides in groups of three [23]; thus, this arrangement is rational.

2. Train/Validation/Test Split. A 70% train, 20% validation, and 10% test split is used where cases are randomly assigned to these sets. The random split was proportional to the available cases in each class to ensure an unbiased and representative split. The validation set is used for tuning the network structure trained on the train set. After finalizing a network structure, the train and the validation sets were combined to retrain the finalized model. The test set is then used to assess the performance of this finalized model.

3. Model Building. A customized Long Short-Term Memory Recurrent Neural Network (LSTM) is developed to perform the classification task. LSTM networks introduced in 1997 are capable of learning long-term dependencies within sequences, storing information about past inputs for an amount of time that is not fixed a priori but rather depends on the input data [24]. They store information long-term using a system of cell states and gates. An illustrative explanation of the process involved with an LSTM layer is provided by Amidi & Amidi [25]. Different components of the developed model are described below.

- Embedding: In this embedding layer, each three-letter word from the input is mapped into a real-valued vector of dimension 18. This mapping that is learned through training provides a dense and efficient representation in which similar three-letter words have a similar encoding. The vectors are then sequentially fed into the next layer.

- LSTM: Three LSTM layers with each having 26 units separated by dropout layers are utilized in the developed model. These LSTM layers process information forward and keep long-term memory of sequences.

- Dropout: Deep learning neural networks are likely to quickly overfit a training dataset. Dropout layer is a regularization method that prevents overfitting by randomly dropping units, assigning a weight of 0 to them during training. A 50% dropout is used in the developed architecture at different steps, as illustrated in Fig. (3).

- Dense: A fully connected dense layer with 26 neurons is utilized where each neuron is connected to all neurons.
from the last layer. Relu activation function is used for the dense layer.

• Output: A fully connected dense layer with 3 neurons and SoftMax activation function is used that predicts output as EI, IE, or N, whichever has the highest likelihood.

4. Assessing Performance. Model performance is assessed using the goodness of fit measures, including accuracy and f-score computed on train, validation, and test datasets. The model building process and assessment, including the random train/validation/test split, are performed recursively until desirable performance is achieved. Accuracy is a common goodness of fit measure for the classification of problems, and is defined as:

\[
\text{Accuracy} = \frac{\# \text{ Correctly Classified Cases}}{\# \text{ Total Cases}} \times 100
\]  

Generally, accuracy alone is not good enough to assess the model performance. Two other important evaluation measures are Precision and Recall, which are defined as

\[
\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \times 100
\]

\[
\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \times 100
\]

Where, for example in class EI, True Positive counts the number of cases correctly classified as EI, False Positive counts the number of cases incorrectly classified as EI, and False Negative counts the number of cases incorrectly classified as IE or N. The f-score calculated by (4) is a more accurate measure of a model’s performance by combining Precision and Recall, which is widely used to compare the performance of classification models. There are two more goodness of fit measures that are widely used in the literature to assess the performance of binary classifications. Area Under Receiver Operating Characteristic (auROC) measures the area under the plot of the False Positive rate (x-axis) versus the True Positive rate (y-axis) for a number of different candidate threshold values. Area Under Precision Recall Curve (auPR-C) measures the area under the plot of the Precision (y-axis) versus the Recall (x-axis) for different thresholds. Both of these measures provide values between 0 and 1; if the value is closer to 1, the better is the model. For multiclass classifications, such as the developed model in this article, these measures can be computed for one-on-one comparisons and averaged to provide a macro comparison tool [26].

\[
f\text{-score} = \frac{2 \times (\text{Precision} \times \text{Recall})}{\text{Precision} + \text{Recall}}
\]

5. Reporting Finalized Model. A final model is selected by investigating the different number of layers, different type of layers, and different number of units for each layer. The best results are achieved with the explained architecture which is reproducible after several runs of the codes. Other compiling options are batch size of 50, Adam optimizer, and sparse categorical cross-entropy loss function.

Fig. (4). Distribution of accuracy and f-score in test set for 50 runs of the developed LSTM model (in blue) and a CNN model for comparison (in gray). (A higher resolution / colour version of this figure is available in the electronic copy of the article).
3. RESULTS

After running the finalized model for 50 runs with random train/test splits, accuracy and f-score were computed on the test set. Fig. (4) shows the distribution of accuracy and f-score for these 50 runs along with their first (lower dashed lines), second (inner dashed line), and third quartile (upper dashed line). The developed LSTM model could achieve an accuracy of 91.34% on average over 50 runs with an average f-score of 91.36%. The three quartiles for accuracy were 90.78%, 91.22%, and 91.93%, while f-score quartiles were 90.85%, 91.25%, and 91.97%. It took approximately 112 seconds on average for the developed model to train over 100 epochs and generate predictions on a single run of the model. Moreover, to provide consistent performance measures for comparison, 10-fold cross-validation has been utilized; the average goodness of fit measures are provided in Table 2. Results showed average accuracy and f-score of 91.31% and 91.27%, respectively, which are roughly the same as average values computed over 50 runs. Average auROC and average auPRC for one-on-one comparisons were 0.9820 and 0.9649, respectively, demonstrating competitive values to recent CNN models.

Results have been compared with a recent CNN model developed by Wang et al. (2019) over 50 runs on dataset described in section 2. Wang et al. (2019) reported 90.25% accuracy for their algorithm but their model could achieve an average accuracy of 92.48% with an average f-score of 92.50% in our runs. Fig. (4) shows the distribution of accuracy and f-score for this CNN model (in grey) along with three computed quartiles (dashed lines). Comparisons showed that this CNN model is slightly over 1% more accurate than our developed model. The average runtime for this model was 89 seconds which is also slightly better.

However, our developed model outperforms other alternative LSTM models in the literature. The most recent LSTM model was developed by Lee et al. (2020) that utilizes one-hot and generates distinct spatio-temporal features from sequences around the splice site to feed into one LSTM layer. After experimenting with a number of units in the LSTM layer, they reached an 86% f-score using multiple Homo sapiens genome datasets [9]. We were not able to make their open codes work and relied on their published results. Moreover, Zhang et al. (2018) also investigated an LSTM model on the HS3D dataset, and their model achieved an auROC score of 0.960 (0.942) on donor (acceptor) splice site classification and an auPRC score of 0.803 (0.721) on donor (acceptor) splice site classification. Clearly, our results outperformed these measures.

Table 1. Summary of recent splice junction identification models for Homo sapiens in literature.

| Article | Method | Data | Results |
|---------|--------|------|---------|
| Mapleson et al. (2018) | An RNA sequence mapping process called Portcullis | ~76 million simulated Human training dataset [13], and combined real data from PRJEB4208 [14] | Up to 98.17% f-score for Homo sapiens |
| Zhang et al. (2018) | DeepSplice using Convolutional Neural Networks | 2,880 (28,800) true (false) acceptor sites; 2,796 (27,960) true (false) donor sites for Homo sapiens from HS3D [15] | auROC of 0.983 (0.974) on donor (acceptor) splice sites and auPRC of 0.863 (0.800) on donor (acceptor) splice sites |
| Zuallaert et al. (2018) | SpliceRover using Convolutional Neural Networks | 1,324 (5,553) true (false) acceptor sites; 1,324 (4,922) true (false) donor sites from NN269 [16] | 96.12% (95.35%) accuracy, 0.9899 (0.9829) au-ROC, 93.96% (93.31%) f-score for detecting acceptor (donor) sites |
| Van Moerbeke et al. (2018) | A linear mixed model, Random Effects for the Identification of Differential Splicing (REIDS) | In total, 33,516 genes were measured using 298,281 exons and 249,475 junctions, each represented by eight probes on average from HJAY [17] | REIDS analytical framework detected between 65–77% of the validated exon probe sets |
| Zhao et al. (2019) | Assembling Splice Junctions Analysis (ASIA) | RNA-seq datasets from twelve normal tissues, seven cancerous tissues and seven matched adjacent tissues from GEO were used [18], making up a total of 322,675 linear junctions and 81,484 Back-splice junctions. | The sensitivity of ASIA known linear junctions is 97.3%. For novel linear junctions, the sensitivity is 89.8% comparing the known splice of 2-pass without annotation with gold standard |
| Wang et al. (2019) | SpliceFinder using Convolutional Neural Networks | 10,000 donor sites, 10,000 acceptor sites, and 10,000 non-splice-site which were randomly selected from Ensembl [19] | Up to 90.25% accuracy |
| Lee et al. (2020) | LSTM and Gated Recurrent Unit (GRU) | Dataset contains consolidated epigenomes from the Roadmap Epigenomics Consortium and the ENCODE Consortium [20] | Up to 86% f-score; above 80% precision-recall curve metric |
| Amilpur et al. (2020) | EDeepSSP using Convolutional Neural Networks | 2,880 (238,431) true (false) acceptor sites; 2,796 (180,975) true (false) donor sites from HS3D dataset [15] | 0.9870 (0.9887) on auPRC and 0.9873 (0.9891) on auROC for acceptor (donor) site detection |
| Albaradei et al. (2020) | Splice2Deep using Ensemble of Convolutional Neural Networks | A total of 250,400 (250,400) true (false) acceptor sites; 248,150 (248,150) true (false) donor sites for Homo sapiens [21]. | Accuracy (f-score) of 96.91% (96.91%) for acceptor site detections, 97.38% (96.38%) for donor site detection |
| Dasari et al. (2020) | InterSSPP using Convolutional Neural Networks | 2,880 (238,431) true (false) acceptor sites; 2,796 (180,975) true (false) donor sites from HS3D [15]. 1,324 (5,553) true (false) acceptor sites; 1,324 (4,922) true (false) donor sites from NN269 [16]. | HS3D: 0.9946 (0.9945) on auPRC and 0.9947 (0.9981) on auROC for acceptor (donor) site detection. NN269: 0.9922 (0.9891) on auPRC and 0.9923 (0.9894) on auROC for acceptor (donor) site detection |
Table 2. Performance comparison with recent LSTM models.

| Article            | Data                                                                 | Results                                                                 |
|--------------------|----------------------------------------------------------------------|------------------------------------------------------------------------|
| Developed model    | 13,666 cases with 3,470 instances (25.39%) of class EI, 3,550 instances (25.98%) of class IE, and 6,646 instances (48.63%) of class N from HSID [15] and UCI [22]. | With 10-fold-CV, average accuracy (f-score) of 91.31% (91.27%), average auROC (auPRC) for one-on-one comparisons was 0.9820 (0.9649). |
| Wang et al. (2019) | 10,000 donor sites, 10,000 acceptor sites, and 10,000 non-splice-sites which were randomly selected have been used from Ensembl [19]. | auROC score of 0.960 (0.942) on donor (acceptor) splice site classification and an auPRC score of 0.803 (0.721) on donor (acceptor) splice site classification. |
| Lee et al. (2020)  | Dataset contains consolidated epigenomes from the Roadmap Epigenomics Consortium and the ENCODE Consortium [20]. | Up to 86% f-score; above 80% precision-recall curve metric. |

4. DISCUSSION

Deep learning models are gaining popularity in various fields, especially bioinformatics. LSTM architecture is a relatively new deep learning design capable of learning and memorizing patterns through a sequence. This method is especially important in speech recognition, text mining, and natural language processing, where dependencies among entries in a sequence exist. Such chain dependencies do exist in mRNA sequences too, thus making the LSTM model feasible. The proposed LSTM design in this research successfully detected splice junctions with high accuracy and could be applied to the discovery of other classification tasks for sequence problems. Various combinations of LSTM, Dropout, and Dense layers with different neurons were investigated to reach the proposed finalized model.

The computations were performed using Python 3.7 with TensorFlow 2.3.0 package on a Windows 10 OS with Core i7-10750H CPU and 32 GB RAM; training took 112 seconds on average for 100 epochs. All data and codes are available to researchers: https://github.com/kr0401/LSTM_Splice

Designing LSTM models that capture RNA sequences’ traits is challenging. We overcame this challenge by processing sequences in 3-letter groups and utilizing an embedding layer that encodes these groups into real-valued vectors. This process not only reduces the dimension of input sequences but also is optimized using input sequences. This is also a logical choice since mRNAs are decoded in groups of three. We investigated processing sequences in 2-letter groups and letter by letter but could not achieve more than 60% accuracy. Considering a classification task with three categories, this accuracy is higher than random predictions but is not good enough.

CONCLUSION

Comparisons showed a highly competitive model to recent Convolutional Neural Network structures. The proposed LSTM model achieves the highest accuracy and f-score among published alternative LSTM structures. Mixing Convolutional layers with LSTM layers could improve detection accuracy but designing a structure that works with both CNN and LSTM layers is challenging.

REFERENCES

[1] Lu, Z.X.; Jiang, P.; Xing, Y. Genetic variation of pre-mRNA alternative splicing in human populations. *Wiley Interdiscip. Rev. RNA*, 2012, 3(4), 581-592. http://dx.doi.org/10.1002/wrna.120 PMID: 22095823

[2] Ding, L.; Rath, E.; Bai, Y. Comparison of alternative splicing junction detection tools using RNA-seq data. *Curr. Genomics*, 2017, 18(3), 268-277. http://dx.doi.org/10.2174/1389202918666170215125048 PMID: 28659722

[3] Mapleson, D.; Venturini, L.; Kaithakottil, G.; Swarbreck, D. Efficient and accurate detection of splice junctions from RNA-seq with Portcullis. *Gigascience*, 2018, 7(12), 1-11. http://dx.doi.org/10.1093/gigascience/giy131 PMID: 30418570

[4] Zhang, Y.; Liu, X.; MacLeod, J.; Liu, J. Discerning novel splice junctions derived from RNA-seq alignment: a deep learning approach. *BMC Genomics*, 2018, 19(1), 971. http://dx.doi.org/10.1186/s12864-018-5350-1 PMID: 30591034

[5] Zuallaert, J.; Godin, F.; Kim, M.; Soete, A.; Saey, Y.; De Neve, W. SpliceRover: interpretable convolutional neural networks for
improved splice site prediction. *Bioinformatics, 2018, 34(24), 4180-4188.
http://dx.doi.org/10.1093/bioinformatics/bty497 PMID: 29931149

Van Moerbeke, M.; Kasim, A.; Shkedy, Z. The usage of exon-exon
splice junctions for the detection of alternative splicing using the
REIDS model. *Scientific Reports, 2016, 8, 8331.

Zhao, J.; Li, Q.; Li, Y.; He, X.; Zheng, Q.; Huang, S. ASIA: A pro-
gram for assembling splice junctions analysis. *Comput. Struct.
Biotechnol. J., 2019, 17, 1143-1150.
http://dx.doi.org/10.1016/j.csbj.2019.08.001 PMID: 31462970

Wang, R.; Wang, Z.; Wang, J.; Li, S. SpliceFinder: ab initio pre-
diction of splice sites using convolutional neural network. *BMC
Bioinformatics, 2019, 20(23)(Suppl. 23), 652.
http://dx.doi.org/10.1186/s12859-019-3036-3 PMID: 31881982

Lee, D.; Zhang, J.; Liu, J.; Gerstein, M. Epigenosome-based splicing
prediction using a recurrent neural network. *PLoS Comput. Biol.,
2020, 16(6), e1008006.
http://dx.doi.org/10.1371/journal.pcbi.1008006 PMID: 32584815

Amilapur, S.; Bhukya, R. EDeepSSP: Explainable deep neural net-
works for exact splice sites prediction. *J. Inform Comput.
Bio., 2020, 18(4), 2050024.
http://dx.doi.org/10.1142/S0219720020500249 PMID: 32696716

Albaradei, S.; Magana-Mora, A.; Thafar, M.; Uludag, M.; Bajic,
V.B.; Gojobori, T.; Esack, M.; Jankovic, B.R. Splice2Deep: An
ensemble of deep convolutional neural networks for improved
splice site prediction in genomic DNA. *Gene, 2020, 5, 100035.
http://dx.doi.org/10.1016/j.gene.2020.100035 PMID: 32550561

Dasari, C.M.; Bhukya, R. InterSSP: Investigating patterns through
interpretable deep neural networks for accurate splice sig-
nal prediction. *Chemom. Intell. Lab. Syst., 2020, 206, 104144.
http://dx.doi.org/10.1016/j.chemolab.2020.104144

Mapleson, D.; Venturini, L.; Kaithakottil, G.; Swarbreck, D. Sup-
porting data for “Efficient and accurate detection of splice junc-
tions from RNA-Seq with Portcullis.”

Pollastro, P.; Rampone, S. HS3D, a dataset of non-coding RNA sequences.
European Nucleotide Archive. Available from: www.ebi.ac.uk/ena/browser/view/PRJEB4208

Pollastro, P.; Rampone, S. HS3D, a dataset of non-coding RNA sequences.
European Nucleotide Archive. Available from: www.ebi.ac.uk/ena/browser/view/PRJEB4208

Pollastri, G.; Ramponi, S. HS3D, a dataset of Homo sapiens splice regions, and its extraction procedure from a major public database. *Int. J. Mol. Biol. 2002, 13(8), 1105-1117.
http://dx.doi.org/10.1471/mol.2002.13.8.1105

Reese, M.G.; Eeckman, F.H.; Kulp, D.; Haussler, D. Improved splice site detection in Genie. *J. Comput. Biol., 1997, 4(3), 311-322.
http://dx.doi.org/10.1089/cmb.1997.4.311 PMID: 9278062

Seok, J.; Xu, W.; Davis, R.W.; Xiao, W. RASA: Robust alterna-
tive splicing analysis for human transcriptome arrays. *Sci. Rep.,
2015, 5, 11917.
http://dx.doi.org/10.1038/srep11917 PMID: 26145443

Zheng, Q.; Bao, C.; Guo, W.; Li, S.; Chen, J.; Chen, B.; Luo, Y.;
Lyu, D.; Li, Y.; Shi, G.; Liang, L.; Gu, J.; He, X.; Huang, S. Circu-
lar RNA profiling reveals an abundant circHIPK3 that regulates
cell growth by sponging multiple miRNAs. *Nat. Commun., 2016,
7, 12115.
http://dx.doi.org/10.1038/ncomms12115 PMID: 27050392

Hubbard, T.; Barker, D.; Birney, E.; Cameron, G.; Chen, Y.;