Diplomová práce

České
vysoké
učení technické
v Praze

F3

Fakulta elektrotechnická
Katedra počítačů

Hledání sekvenčních motivů v mRNA selektovaných vazbou na translační iniciační faktory z rodiny eIF4E

Jan Holčák

Vedoucí: RNDr. Martin Pospíšek, Ph.D.
Obor: Otevřená informatika
Studijní program: Kybernetická bezpečnost
Srpen 2020

Poděkování

Rád bych touto cestou vyjádřil poděkování RNDr. Martinu Pospíškovi, Ph.D. za jeho cenné rady a návrh velmi zajímavého tématu diplomové práce. Současně bych chtěl poděkovat také všem, kteří mě při tvorbě této práce podpořili, a bez jejichž pomoci by nebylo možné práci dokončit.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze, 14. srpna 2020

Abstrakt

Výběr vhodného nástroje hledajícího neobjevené motivy v RNA sekvencích je obtížný. Náročná instalace programů může vést k problémům s nasazením. Dostupné sady nástrojů nejsou připraveny na hromadné výpočty. Nástroje schopné integrovat výstup většího množství programů jsou zatím jen ve fázi prototypů.

Uvedené problémy jsou řešeny sestavením množství Docker kontejnerů kompatibilních se Singularity. Realizován je izolovaný, paralelní běh programů bez potřeby znalosti specifických parametrů. Vyřešeno je také převedení výsledků do MEME formátu vhodného pro další zpracování. Zakomponován je vylepšený program integrující objevené motivy. Navržena je příprava dat generováním FASTA souborů uplatňujících rozdílů mezi sekvenačními daty a referenčním genomem.

Zjednodušeno je nasazení programů hledajících motivy nezávisle na zvolené platformě. Implementované hromadné spouštění umožňuje výrazně rychlejší hledání a následné zpracování nalezených motivů. Zvolené řešení umožňuje také velmi rychlou změnu verze nebo modifikaci podporovaných nástrojů.

Klíčová slova: hledání nových motivů, RNA-seq, UTR, DMD, RNA vázající proteiny

Vedoucí: RNDr. Martin Pospíšek, Ph.D.
Genetics and Microbiology,
Viničná 1965/5,
Praha 128 43

Abstract

Choosing the right de-novo motif search tool for RNA sequences can be tough. Difficult tool installation can lead to later deployment issues. There is no toolkit combining motif discovery tools properly implemented with bulk data processing in mind. Tools capable of integrating the output of a larger number of programs are currently only in the prototyping phase.

These issues are addressed by building a number of Singularity-compatible Docker containers. An isolated, parallel running of programs is realized without the need for knowledge of specific parameters. The conversion of results into MEME format suitable for further processing is also solved. A program integrating the discovered motifs is included. Data preparation by generating FASTA files with applied differences observed between sequencing data and reference genome is proposed.

The deployment of motif discovery tools, regardless of the chosen platform, is significantly simplified. Implemented bulk excution allows significantly faster search and subsequent processing of discovered motifs. The chosen solution also allows a very fast version change or modification of supported tools.

Keywords: de-novo motif discovery, RNA-seq, UTR, DMD, RBP

Title translation: Search for sequence motifs in mRNAs selected by binding of translation initiation factors from the eIF4E family

Obsah

Obrázky

Tabulky

Kapitola 1

Úvod

Při zkoumání nukleotidových sekvencí se můžeme setkat s opakujícími se vzory tzv. motivy. Výskyt těchto motivů může signalizovat interakci příslušných oblastí nukleových kyselin s regulačními bílkovinami. Oblast výskytu motivu tak hraje důležitou roli v regulačních procesech a mezi-buněčné komunikaci.

Velký biologický význam sekvenčních motivů učinil jejich hledání často řešeným problémem. Protože se jedná o problém NP-úplný, řeší se hledání na základě nejrůznějších heuristik. Provedená rešerše ukázala na nepřeberné množství různých přístupů, které již byly vyzkoušeny. Při pohledu na stovky publikací lze usoudit, že vytvoření nového nástroje nepřinese pravděpodobně příliš velký užitek. Pozornost proto bude zaměřena převážně na zjednodušení distribuce implementovaných programů.

Rozmanitost dat vycházejících z různých biologických experimentů a specifické požadavky laboratoří vedly ke vzniku velkého množství jednoúčelových nástrojů, jejichž společným cílem je hledání signifikantně se vyskytujících motivů. Nástroje vytvořené za účelem zpracování jednoho experimentu často nejsou příliš obecné a jejich nepřeberné množství činí výběr vhodného nástroje velmi obtížným. Další překážkou je očekávaná vysoká zdatnost uživatelů při nasazení zvoleného programu na požadovanou platformu.

Cílem této práce je řešení zmíněných problémů navržením uceleného postupu pro vyhledávání a analýzu sekvenčních motivů v datové sadě získané metodou RNAseq NGS. Obsaženo je také testování vhodnosti nástrojů pro data pocházející ze sekvenace RNAseq se zaměřením na úseky nepřekládaných oblastí UTR.

Při hledání musíme mimo formát vstupních dat, zohlednit také postup, kterým byla data získána. Při práci s nukleotidovými sekvencemi se běžně můžeme setkat s různými metodami sekvenace a jejich unikátními vlastnostmi dat za použití stejného formátu.

Samotnému hledání motivů bude předcházet příprava dat do běžně podporovaného formátu. Navázáno bude na již částečně zpracovaná data sekvenace a provedenou analýzu.

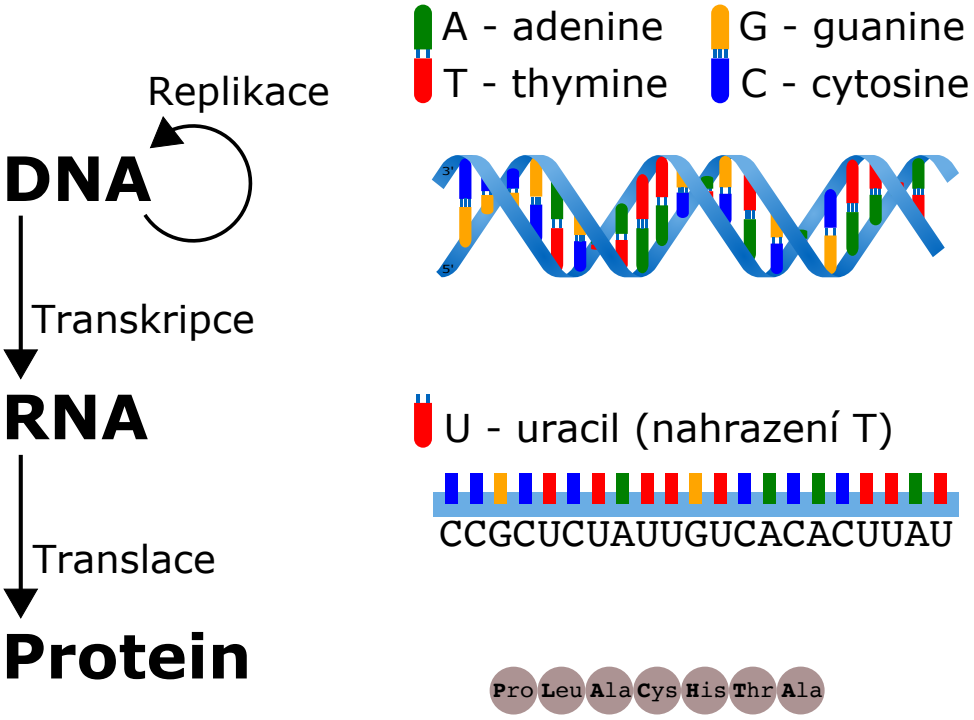
K volbě tohoto bioinformatického tématu autora přivedl očekávaný zvyšující se význam těchto technik z důvodu postupného snižování nákladů na sekvenování nukleových kyselin. Dalším důvodem je množství zajímavých problémů, které autoři bioinformatického softwaru často řeší velmi kreativním způsobem.

Kapitola 2

Problematika hledání sekvenčních motivů

S vývojem a zlevněním sekvenovacích technologií došlo ke zvýšení nároků zpracování velkých objemů sekvenačních dat. Důvodem je velká komplexita a rozmanitost biologických systémů obsahujících značné množství informace. Vysoká je také rychlost proměnlivosti informace se kterou tyto systémy pracují.

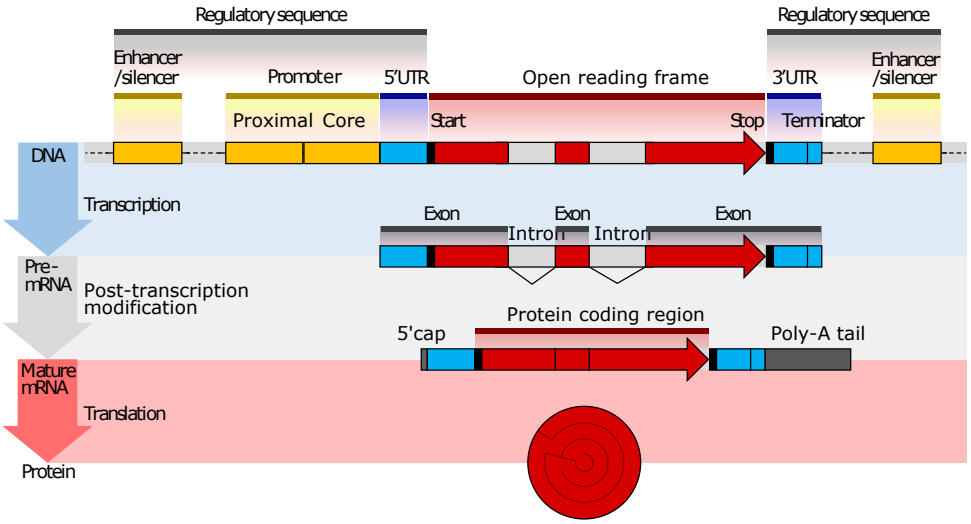
Při zpracovávání dat z bioinformatických experimentů se setkáváme s informacemi uchovávanými v biologické podobě prostřednictvím DNA, RNA a bílkovin. V živých buňkách existují tři hlavní způsoby přenosu informace mezi těmito biologickými strukturami. U DNA se můžeme setkat s replikací a transkripcí na RNA, podle které se při translaci vytvoří řetězec aminokyselin tvořící výsledný protein.



Obrázek 2.1: Centrální dogma

Informace v DNA je uchovávána v podobě dvojité šroubovice řetězců nukleotidů. Při zpracování DNA sekvencí využíváme předpokladu, že jsou řetězce nukleotidů propojeny podle základních watson-crickovských pravidel komplementarity [Wik20b], tzn. báze adenin (A) páruje s thyminem (T) a báze guanin (G) páruje s cytosinem (C). Běžně je tedy uložen pouze jeden ze sekvenovaných řetězců s kódováním v abecedě znaků A, C, G a T. Sekvence RNA tvořená jediným řetězcem obsahujícím nukleotid uracilu U zastupující funkci nukleotidu T poté kódujeme abecedou A,C,G a U. Pro značení délky sekvence tvořené komplementárním párem bází (base pair) se používá ustálená zkratka bp.

V nukleotidových sekvencích se často vyskytují sekvence nesoucí podobný vzor. Opakování podobných úseků sekvencí nukleotidů nazýváme sekvenčním motivem. U jedno-řetězcových nukleových kyselin (například mRNA) mohou díky nepárovaným bázím vznikat také různé sekundární struktury tvořené párováním volných bází stejného řetězce. Častý výskyt kombinace sekvence a její sekundární struktury nazýváme tzv. strukturním motivem. Sekvenční motiv někdy označujeme také jako 1D motiv a strukturní zobrazovaný ve dvou osách jako 2D motiv [GR14].



Obrázek 2.2: Rozdíly v oblastech mezi sekvencí DNA a RNA [Wik19]

Abecedy sekvencí DNA i RNA jsou mezi sebou lehce převeditelné záměnou písmen a RNA sekvence jsou proto někdy ukládány ve stejné abecedě jako DNA. Podobnost abeced sekvencí DNA i RNA také často umožňuje použít stejné algoritmy hledající sekvenční motivy. Důležité je ale věnovat pozornost určitým odlišnostem v obsažených úsecích obou typů sekvencí. Zatímco u DNA často zkoumáme motivy z důvodu hledání transkripčních faktorů regulujících transkripci, u sekvencí RNA je pozornost soustředěna převážně na nepřekládané oblasti 5' a 3' UTR (untranslated region).

Dalšími odlišnostmi může být například jiné zastoupení pravděpodobností výskytu nukleotidů nebo rozdílná délka zkoumaných sekvencí. Průměrná délka

mRNA transkriptu lidského genomu je 3392 bp, ale při zkoumání kratších úseků UTR narazíme na sekvence průměrné délky 259 bp pro 5' UTR a 1470 bp pro 3' UTR [PCA+16]. Dále se můžeme při přímém zpracování sekvenačních dat v závislosti na použité technologii setkat s výrazně rozdílnou délkou celistvých sekvencí tzv. čtení (reads). Tyto odlišnosti mohou způsobit problémy při použití nástroje, který nebyl na množství rozdílných parametrů navržen.

Velké množství nástrojů se specializuje na zpracování dat získaných různými metodami sekvenace. Rozdíly mezi jednotlivými metodami se nachází převážně v dodatečných informacích získaných při sekvenaci. Dále se nástroje zaměřují také na rozdílné biologické funkce motivů.

Podstatná část této práce je zaměřena na vyhledávání a analýzu sekvenčních motivů v nepřekládaných oblastech mRNA. Pozornost je převážně zaměřena na sekvenční motivy specificky interagující s lidskými translačními iniciačními faktory z rodiny eIF4E.

2.1 Reprezentace sekvenčních motivů

Hledání motivů v biologických sekvencích (de-novo motif search) se zabývá vzory, které se v těchto sekvencích zvýšeně vyskytují a nejsou doposud obsaženy v databázích popsaných motivů. Pro tento účel bylo vyvinuto nepřeberné množství nástrojů hledajících sekvenční nebo strukturní motivy. Většina programů přišla s formátem motivů vhodným pro danou aplikaci. Naštěstí je většina proprietárních formátů založená na podobných principech.

1. GAGGTAAA
2. TCCGTAAG
3. CAGGTTGG
4. ACAGTCAG
5. TAGGTCAT
6. TAGGTACT
7. ATGGTAAC
8. CAGGTATA
9. TGTGTGAG
10. AAGGTAAG

List 2.1: Ukázka motivu vyjádřeného nalezenými sekvencemi.

Záznam motivů můžeme rozdělit do dvou kategorií. V prvním případě se jedná o záznam podřetězců nalezených v prohledávaných sekvencích 1. Tato varianta je často vylepšena doplněním pozice výskytu uváděné vzhledem k prohledávané sekvenci.

Další častou variantou je využití matice, kde jeden z rozměrů je tvořen velikostí abecedy a druhý odpovídá délce motivu. Matice pak ještě můžeme dále rozdělit na tři základní typy a to position frequency matrix (PFM), position probability matrix (PPM) a position weight matrix (PWM). Vezmeme-li množinu S obsahující N zarovnaných sekvencí $s1, \ldots, sN \in S$ z předchozího případu 1, pak příslušnou matici PFM vytvoříme součtem stejných symbolů zvlášť pro každý sloupec zarovnaných sekvencí o délce L.

$$PFM_{i,j} = \sum_{k=1}^{N} I(S_{k,j} = i)$$

$$PFM = \begin{array}{c|cccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ A & 3 & 6 & 1 & 0 & 0 & 6 & 7 & 2 \\ C & 2 & 2 & 1 & 0 & 0 & 2 & 1 & 1 \\ G & 1 & 1 & 7 & 10 & 0 & 1 & 1 & 5 \\ T & 4 & 1 & 1 & 0 & 10 & 1 & 1 & 2 \end{array}$$

$$PPMi, j = \frac{1}{N} PFM_{i,j}$$

$$PPM = \begin{array}{c|cccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ A & 0{,}3 & 0{,}6 & 0{,}1 & 0{,}0 & 0{,}0 & 0{,}6 & 0{,}7 & 0{,}2 \\ C & 0{,}2 & 0{,}2 & 0{,}1 & 0{,}0 & 0{,}0 & 0{,}2 & 0{,}1 & 0{,}1 \\ G & 0{,}1 & 0{,}1 & 0{,}7 & 1{,}0 & 0{,}0 & 0{,}1 & 0{,}1 & 0{,}5 \\ T & 0{,}4 & 0{,}1 & 0{,}1 & 0{,}0 & 1{,}0 & 0{,}1 & 0{,}1 & 0{,}2 \end{array}$$

$$PWM_{i,j} = \log_2 \left(PPM_{i,j} / b_i \right)$$

$$PWM = \begin{array}{c|cccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ A & 0{,}26 & 1{,}26 & -1{,}32 & -\infty & -\infty & 1{,}26 & 1{,}49 & -0{,}32 \\ C & -0{,}32 & -0{,}32 & -1{,}32 & -\infty & -\infty & -0{,}32 & -1{,}32 & -1{,}32 \\ G & -1{,}32 & -1{,}32 & 1{,}49 & 2{,}0 & -\infty & -1{,}32 & -1{,}32 & 1{,}0 \\ T & 0{,}68 & -1{,}32 & -1{,}32 & -\infty & 2{,}0 & -1{,}32 & -1{,}32 & -0{,}32 \end{array}$$

kde: $i \in \{A, C, G, T\}, \quad j \in (1, ..., L), \quad b_i \in B$ [Gui03]

Model pozadí B (background) vyjadřuje pravděpodobnosti výskytu symbolů v celé prohledávané sekvenci. V ukázce se předpokládalo rovnoměrné rozdělení všech čtyřech symbolů $1/4 = 0,25$ [Gui03].

2.1.1 Konsenzuální sekvence

Při zápisu motivů písmeny anglické abecedy se užívá hned několik formátů, snažících se o zaznamenání nejistoty výskytu symbolů. Nejčastěji se můžeme setkat s konsenzuální sekvencí užívající IUPAC symbolů definovaných stejnojmennou společností [CB85]. Drobnou nevýhodou tohoto formátu je nutná znalost významu jednotlivých symbolů.

1. KCTTTTWV
2. KCTTTTAR
3. CTAAAGKS

List 2.2: Ukázka motivů využívajících abecedy IUPAC

V ukázce 2.1.1 vyskytující se symbol K například značí výskyt nukleotidů G nebo T. Dalším příkladem je symbol V, který značí možný výskyt tří nukleotidů A, C nebo G. Tento systém se používá také pro RNA, kde je používán symbol T jako zástupce symbolu U.

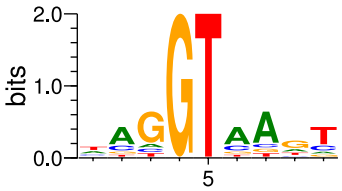
Některé programy namísto implementace IUPAC abecedy zobrazují jen symbol s nejčastějším výskytem. Dále se lze setkat také s užitím regulárních výrazů. Použití přináší snadnou čitelnost, ale značnou nevýhodou je horší porovnatelnost delších sekvencí způsobená velkými délkovými rozdíly a ztrátou zarovnání.

1. [GT]CTTTT[AT][ACG]
2. [GT]CTTTTA[AG]
3. CTAAAG[GT][CG]

List 2.3: Ukázka motivů využívajících regulárních výrazů.

2.1.2 Formát WebLogo

V případě grafického výstupů, který je často zprostředkován generováním webové stránky se využívá grafického znázornění WebLogo generovaného podle četnosti výskytu symbolu. Zmíněný formát má několik variant užívaných podle kontextu, ve kterém je motiv uváděn. U programů hledajících motivy se setkáváme s variantou udávající rozdíl mezi maximální možnou a sledovanou entropií rozdělení výskytu symbolů [CHCB04].



Obrázek 2.3: Ukázka zobrazení motivu v grafickém formátu WebLogo [web].

Kapitola 3

Nástroje hledající motivy v nukleotidových sekvencích

Hledání motivů je oblíbený problém v bioinformatice, existuje proto velké množství nástrojů, které tento problém řeší. Nástroje jsou často specializovány pro hledání motivů v experimentu pro který byly navrženy.

Před započetím práce proběhla rešerše aktuálně dostupných nástrojů pro hledání sekvenčních motivů. Nalezené nástroje byly zhodnoceny a některé poté otestovány. Programy bylo zapotřebí otestovat aby se ukázalo, zda umožňují řešení požadovaného problému.

Vybrány byly programy umožňující hledat nové motivy bez využití databáze (de-novo motif search). Vítanou vlastností bylo hledání motivů obohacených pouze v jedné ze dvou datových sad DMD (discriminative motif discovery).

Největší skupinu nástrojů vyhledávajících sekvenční motivy tvoří nástroje navržené pro hledání motivů vázajících transkripční faktory (TFBM). Tyto proteiny kontrolující míru transkripce se váží v oblasti promotoru, která se nachází pouze v sekvencích DNA. Některé nástroje jsou úzce specializovány na vyhledáváni motivů tohoto typu a nemohou být proto použity pro zkoumaný dataset skládající se ze sekvencí RNA.

V porovnání s RNA dochází v oblasti zabývající se TFBM k rychlejšímu vývoji. Rychlý vývoj a zaměření na hledání sekvenčních motivů v této oblasti vedlo k prozkoumání některých nástrojů také z tohoto odvětví. Otestovány byly nástroje u kterých byla očekávána dostatečná obecnost umožňující zpracování zadaného vstupního datasetu.

Při rešerši bylo zjištěno, že se pro hledání TFBM v datasetech ChIP-seq pro to určených využívá také nástrojů, které na tuto činnost nejsou primárně navrženy a zároveň podávají dobré výsledky [BvH18].

Většina programů umožňuje vyhledávat při zadání jednoho až dvou souborů sekvencí. První soubor s obsahem motivů a druhý bez výskytu pro vytvoření modelu pozadí datové sady. Některé programy umožňují vyhledávat také podle zadaných oblastí zvoleného genomu. Tato funkce bývá dostupná hlavně

ve větších nástrojích, ale nebývá dobře podporována.

Při hledání motivů se můžeme setkat s rozdílným způsobem definice délky hledaného motivu. Některé programy neumožňují nastavit hledanou délku a výstupem jsou různě dlouhé motivy. Další skupina programů umožňuje hledat podle několika zadaných délek. Někdy je ale nastavení omezeno pouze na jedinou volitelnou délku. U těchto programů se pro hledání různě dlouhých motivů využívá vícenásobného spuštění s inkrementací argumentu délky.

Prohledávání souboru sekvencí je založeno na jednom ze tří modelů výskytů motivů v prohledávaných sekvencích. První předpokládá, že každá sekvence obsahuje jeden motif. Model bývá označován jako OOPS (one occurrence per sequence). Druhý povoluje i sekvence bez motivu s označením ZOOPS (zero or more occurrences per sequence). Poslední model počítá s libovolným množstvím motivů v sekvenci. U programu MEME a programů z něj vycházejících se tento model dříve nazýval TCM (two-component mixture). [HMAA19] Novější programy včetně MEME již označují zmíněný model jako ANR (Any NumbeR) [LMPT13].

Podoba výsledků hledání je mezi většinou programů velmi rozdílná. Můžeme se setkat s programy, které umí zobrazit výsledky pouze na standardní výstup. V případě ukládání výsledků bývá běžně využíván proprietární formát navržený autorem nebo v horším případě dochází k ukládání ve formě pouhé matice motivu bez dodatečných dat. Proprietární formát se může skládat z textových souborů, souborů webových stránek HTML nebo také obrázků motivů nejčastěji formátu WebLogo.

Výsledky se skládají nejen z několika nalezených motivů, ale také dalších souvisejících hodnot. Často je zobrazeno skóre podle kterého daný program vyhodnocuje signifikantnost. Setkat se můžeme také s údajem o počtu výskytů v prohledávaných sekvencí nebo s přesnou polohou všech výskytu vztažených k příslušné sekvenci ve které došlo k nálezu.

3.0.1 Použití frameworku pro hledání motivů v sekvencích

Problém s výběrem vhodného nástroje řeší několik projektů zabývajících se integrací množství nástrojů do jednotného rámce, ve kterém jsou programy spouštěny. Výhodou je jednotný výstup zajištěný převodem a zpracováním výsledků jednotlivých nástrojů. Pro porovnání byly otestovány 4 projekty.

- **GimmeMotifs for transcription factor motif analysis [BvH18]**
- **MCAT: Motif Combining and Association Tool [YRG+19]**
- **DynaMIT, the Dynamic Motif Integration Toolkit [DQ15]**
- **EMD Ensemble Motif Discovery [HYK06]**

Program	Gimme	DynaMIT	MCAT	EMD	SUM
AlignACE	-	-	-	✓	1
AMD	✓	-	-	-	1
BioProspector	✓	-	✓	✓	3
ChIPMunk	✓	-	-	-	1
CMF	-	-	✓	-	1
CMfinder	-	✓	-	-	1
DECOD	-	-	✓	-	1
DiNAMO	✓	-	-	-	1
DREME	✓	-	-	-	1
GADEM	✓	-	-	-	1
Gibbs	-	✓	-	-	1
GLAM2	-	✓	-	-	1
GraphProt	-	✓	-	-	1
HMS	✓	-	-	-	1
HOMER	✓	✓	-	-	2
Improbizer	✓	-	-	-	1
MDmodule	✓	-	-	-	1
MDscan	-	✓	-	✓	2
MEME	✓	✓	✓	✓	4
MEMERIS	-	✓	-	-	1
MotifSampler	✓	-	-	✓	2
Posmo	✓	-	-	-	1
ProSampler	✓	-	-	-	1
RNAforester	-	✓	-	-	1
RNAhybrid	-	✓	-	-	1
RNAprofile	-	✓	-	-	1
Trawler	✓	-	-	-	1
Weeder	✓	✓	✓	-	3
XXmotif	✓	-	✓	-	2
YAMDA	✓*	-	-	-	1
RPMCMC	✓*	-	-	-	1
Celkem	**17+2**	**12**	**6**	**5**	

Tabulka 3.1: Programy hledající motivy podporované v jednotlivých sadách nástrojů. *Pouze podpora, distribuce bez nainstalovaného programu.

3.0.2 Framework GimmeMotifs

GimmeMotifs je framework sdružující množství nástrojů se zaměřením na hledání transkripčních faktorů v datových sadách experimentů ChIP-seq [BvH18]. Jedná se o framework obsahující 17 již nainstalovaných nástrojů. Podporovány jsou také další dva nástroje YAMDA a RPMCMC, které může uživatel doinstalovat bez potřeby úprav frameworku. Jedná se o nejpokročilejší a aktivně vyvíjený framework s možností instalace správcem balíčků bioconda. Dále je možné využít již připravený kontejner dodávaný komunitním projektem

BioContainers.

V rámci této práce byl GimmeMotifs otestován, aby se zjistilo, zda některý z nástrojů nedosahuje dostatečné obecnosti pro hledání motivů v datové sadě sekvencí zkoumané v této práci. K prozkoumání tohoto frameworku vedly velmi dobré výsledky některých nástrojů, které nebyly primárně navrženy pro zpracování ChIP-seq datových sad [BvH18].

Při použití nástroje GimmeMotifs se vyskytly problémy s využitím velkého množství operační paměti. I při volbě několika málo nástrojů dosahovaly požadavky na operační paměť v řádech desítek GB. Chyba se vyskytovala nepředvídatelně a při nedostatku volné paměti došlo k zamrznutí aplikace. Výskyt tohoto problému učinily framework nepoužitelným pro hromadné výpočty na výpočetním clusteru.

Problém s nepřiměřeným využitím paměti vedl ke spouštění samostatných nástrojů bez účasti frameworku. Využita je pouze malá část nepostižených funkcí volaných ve skriptech řešících konverzi výsledků spuštěných programů.

Pro zprovoznění většího množství funkcí byly provedeny další úpravy kontejneru, ve kterém je framework distribuován. Základní úpravou je aktualizace dvou hlavních komponent frameworku, ve kterých byly nalezeny chyby. Rychlá oprava provedená autorem frameworku tak nahradila vlastní řešení nalezeného problému. Dalším nalezeným problémem je špatné zpracování vstupních argumentů, které zapříčiňuje nespuštění některých programů bez výpisu chybové hlášky. Problém je vyřešen lepší kontrolou zadávaného vstupu v obalujícím skriptu.

3.0.3 Sada nástrojů MCAT

Projekt MCAT zahrnuje 6 vybraných nástrojů. Výsledky ve formátu sekvenčních motivů a jejich pozic jsou porovnány podle významnosti stanovené shodou použitých nástrojů. Poté je provedena shluková analýza maximalizující konsensus motivů. [YRG+19]

Volně přístupný zdrojový kód byl nahrán na Github bez návodu na instalaci. Uvedena je zde pouze část balíčků potřebných pro chod ukázkového skriptu. Zdrojový kód je jen velmi zřídka okomentován a obsahuje množství nepopsaných konstant. V připraveném skriptu jsou nastaveny hodnoty nevhodné pro zkoumanou datovou sadu. Upraveny byly parametry programu *weeder* změnou souboru frekvencí genomu. Dále došlo k odkomentování řádku kódu, obsahujícího volání programu DECOD. Tento program podle publikace [YRG+19] nepřinesl zlepšení výsledků na testovaných souborech. Dále došlo u programu DECOD ke snížení počtu iterací, aby byl zajištěn kratší čas běhu tohoto programu. Čas běhu v opačném případě mnohonásobně převyšuje čas hledání ostatních programů sady nástrojů MCAT.

Přestože je snahou práce spouštět nástroje tak, jak jsou dodávány, zdrojový kód této sady nástrojů musel projít úpravami zabraňujícími zápisu velkého

množství dočasných souborů. Docházelo k vytváření souborů na různých místech v adresáři zdrojového kódu. Projekt obsahuje také další chyby týkající se shlukování. I tyto chyby musely být opraveny, aby nedocházelo k pádům celé sady nástrojů. Z uvedených důvodů by bylo vhodné nástroj alespoň částečně přepsat. Přidání většího množství parametrů volitelných při spouštění nástroje by zamezilo nutným úpravám hodnot přímo ve zdrojovém kódu.

3.0.4 Framework EMD

Tento projekt shlukuje výsledky nejlepších programů v roce 2006. Poslední aktualizace bohužel proběhla v roce 2009 a nástroj není nadále vyvíjen. Zdrojový kód s postupem instalace je zveřejněn na stránce projektu. Framework byl vybrán za účelem lepšího zakomponování programu AlignACE a srovnání výsledků různých verzí programů.

3.0.5 Framework DynaMIT

Projekt DynaMIT se vyznačuje obsaženým množstvím nástrojů pro integraci motivů [Das18]. Zaměřuje se na problematiku zpracování různorodého výstupu při použití několika velmi rozdílných programů. Celý postup hledání motivů je rozdělen na tři kroky nazvané Search, Integrate a Print. V prvním kroku se podle zvolených programů spustí vyhledávání a po dokončení jsou výsledky převedeny do jednotného formátu. Vybírat lze libovolnou kombinaci programů uvedených v tabulce 3.3. Framework nezávisle na zvolených vyhledávacích programech integruje nalezené motivy podle uživatelem zvolené integrační strategie. Integrovaný souhrn motivů vygenerovaný v předešlém kroku je poté převeden do lehce čitelné podoby volbou jednoho nebo více generátorů uživatelsky přívětivého výstupu.

Strategie	Informace pro shlukování nebo vytvoření consensu
Alignment	provede párový alignment, vypočítá alignment skóre
Biclusterin	použije funkci spectral biclustering algorithm na motivy a sekvence
CoOccurrence	vypočítá “co-occurrence score” pro dvojice motivů na stejné sekvenci
Jaccard	vypočítá “Jaccard similarity score” pro páry motivů nacházející se na stejné pozici v sekvenci
MI	vypočítá “mutual information score” pro páry motivů nacházející se na stejné pozici v sekvenci
PCA	provede redukci pomocí PCA do dvou komponent
Proximity	vypočítá skóre na základě množství případů, kdy dochází k výskytu dvojice motivů v určité vzdálenosti

Tabulka 3.2: Tabulka integračních strategií. Vytvořeno na základě dokumentace [Das18] a publikace [DQ15].

Všechny kroky proběhnou nezávisle na zvolených nástrojích. Při volbě nespolupracujících nástrojů dojde pouze k zobrazení menšího množství užitečných informací. K nastavení nejzákladnějších parametrů je připraven jednoduchý nástroj s GUI, tvořící konfigurační soubor s parametry pro všechny tři zmíněné kroky. Lze volit také z již přednastavených profilů hledání v datových sadách DNA, ChIP, CLIP a RNA.

Volně přístupný repozitář se zdrojovým kódem a manuálem je k dispozici na serveru Bitbucket. Zamýšlený způsob distribuce je realizován vytvořením virtuálního disku, zakládajícím se na již nepodporovaném operačním systému Ubuntu 14.10. Dodávaný systém obsahuje připravené všechny podporované nástroje a jejich závislosti. Tento způsob distribuce naráží na potřebu správy nejen samotného frameworku, ale také zvoleného operačního systému. Při použití poskytovaného virtuálního disku se vyskytly problémy s kompatibilitou u stávající verze Virtualboxu 6.1 a následnou instalací balíčku VirtualBox Extension Pack. Tento balíček je důležitý pro zajištění uživatelské přívětivosti při používání operačního systému v nástroji Virtualbox. Samotná instalace není příliš uživatelsky přívětivá z důvodu použití disku formátu VDI (Virtual Disk Image) namísto běžně používaného balíčku OVA (Open Virtual Appliance).

Další možností je instalace s použitím správce balíčků pip (repozitář PyPI). Instalace touto cestou vyžaduje instalaci jednotlivých programů hledajících motivy. V obsáhlém manuálu jsou uvedeny odkazy ke stažení jednotlivých programů. Seznam ale bohužel neobsahuje podporované verze, což je zvláště problematické z důvodu nedostupnosti velké části odkazů.

Drobnou nevýhodou tohoto projektu je nemožnost odděleného spuštění jednotlivých kroků. Framework naštěstí nabízí možnost navázat na již vypočtené kroky, ale v neupraveném projektu nelze jednoduše rozpoznat, kdy dochází k pádu programu hledajícího motivy a kdy dochází k chybě při zpracování výsledků hledání.

3.0.6 Nakládání s parametry

Jednotlivé frameworky přistupují k míře zapouzdření rozdílně. Například Gimme Motifs se kompletně stará o nastavení parametrů zvolených programů a uživatel tak nemusí studovat množství manuálů. Velkou výhodou tohoto řešení je zamezení opakovaného spouštění u programů hledajících motiv pevné délky. Při hledání motivů v rozsahu hodnot jsou programy automaticky spouštěny s postupně se zvyšující délkou hledaných motivů až do prohledání celého rozsahu. Hledání různých délek umožňuje dále také EMD, který má rozsahy nastaveny v konfiguračním souboru. Sada nástrojů MCAT umožňuje hledat pouze motivy jediné délky. Při vynechání parametru nastavujícího délku hledaného motivu je využita výchozí hodnota 12.

Framework DynaMIT přenechává celý proces volby parametrů na uživateli. Spouštění programů se řídí konfiguračním souborem, který musí uživatel

vytvořit. Hledání motivů různé šířky lze docílit vložením množství řádků obsahujících hledání pro rozdílnou délku motivu.

Program	Využit	Komentář	DMD
AlignACE	✓		-
AMD	-	Malá signifikantnost hledaného motivu (ChIP)	-
BioProspector	✓	Velmi dobré výsledky v krátkém čase	-
ChIPMunk	✗	Vyžaduje peak soubor (pouze ChIP)	-
CMF	✓	Nestabilní běh programu.	-
CMfinder	✓	Pouze RNA 2D ale integrace dynamit	-
DECOD	✓	Hledání výpočetně náročné.	✓
DiNAMO	✓	Nízký obsah informace v motivu (ChIP)	✓
DREME	✓	Pouze kratké motivy se signifikantností (ChIP)	✓
GADEM	✓	Velmi dlouhý běh	-
Gibbs	✓		-
GLAM2	✓	Velké rozmezí délky motivů až od jediného nukleotidu	-
GraphProt	✓	Využívá skrukturu ale výstupem 1D	✓
HMS	✗	Vyžaduje peak soubor (pouze ChIP)	-
HOMER	✓	Dobré výsledky pro DMD (ChIP)	✓
Improbizer	✗	Neuspokojivé výsledky	-
MDmodule	✗	Využívá biologických vlastností TF	-
MDscan	✗	Zaměřeno na ChIP	-
MEME	✓	Velmi dobré výsledky v krátkém čase	✓
MEMERIS	✓	Využívá skrukturu sekvence	-
MotifSampler	✗	Neuspokojivé výsledky	-
Posmo	✗	Nestabilní běh	-
ProSampler	✗	Nezahrnut kvůli chybě v pipeline	-
RNAforester	✗	Pouze RNA 2D výstup	-
RNAhybrid	✗	Pouze RNA 2D výstup	-
RNAprofile	✗	Pouze RNA 2D výstup	-
Trawler	✗	Neuspokojivé výsledky	-
Weeder	-	Zachován v mcat (slabé výsledky, ChIP-seq)	-
XXmotif	✓	Dlouhé motivy se signifikantností	✓
YAMDA	✗	Zaměřeno pouze na ChIP-seq. Nutné manuální ladění parametrů.	-
RPMCMC	✗	Neuspokojivé výsledky, možné dlouhé motivy	-

Tabulka 3.3: Přehled programů a jejich zařazení do hledání motivů. Pouze programy obsažené v použitých frameworcích.

Duplicita programů

Použitím několika frameforků došlo k výskytu duplicit v seznamu dostupných programů. Tento stav není nežádoucí, ale naopak posloužil ke kontrole výkonu aplikací různých verzí navíc instalovaných v odlišných prostředích. Například program Weeder testovaný v DynaMIT, ve frameworku GimmeMotifs a zároveň také jako samostatný nástroj ukázal, že pomocné nástroje tohoto programu obsažené v dodávaném disku projektu DynaMIT pomáhají výrazně zlepšit výsledky hledání.

Vyřazení programů využívající peak soubory

V experimentu ChIP-seq se využívá zvýšeného obohacení v oblastech s navázaným proteinem označovaných jako peak [KTP08]. Pro hledání těchto oblastí se využívá programů generujících soubory obsahující nalezené Peaky. [Fej08] Nástroje úzce zaměřené na datové sady ChIP-seq využívající tyto soubory byly vyřazeny. Prozkoumána byla také možnost hledání Peaků v datech zkoumané datové sady, ale postup tímto směrem nebyl dále rozšiřován z důvodu náročnosti procedury a nutné hlubší znalosti postupu přípravy dat pro sekvenaci. Hledání Peaku bez znalosti některých parametrů může vést k velkému množství falešně pozitivních nálezů. [AHS+18]

3.0.7 Nástroje frameworků

AlignACE

Program využívá Gibbs sampling a pro shlukování velkého množství motivů je využíváno algoritmu CompareACE [HETC00].

AMD

Založeno na hledání IUPAC motivů s možností mezery. Motivy jsou degenerovány, prodlužovány a padesát nejlepších motivů je poté převedeno na PWM s následným odstraněním redundance. [SYC+11]

BioProspector

Vylepšení Gibbs sampling pro model očekávající žádný nebo mnoho motivů v sekvenci s rozšířením pro motivy obsahující mezery. Bioprospector využívá markovovy řetězce k tvorbě background modelu. Kvalita motivů je posuzována metodou Monte Carlo. [LBL01] Program je společně s programy MEME a DREME nastaven jako výchozí sestava pro vyhledávání motivů. Mimo velmi

dobré výsledky je výhodou také obecnost návrhu nástroje, který exceluje i na jiných než zamýšlených datových sadách [BvH18].

CMF

Slouží k hledání kompozitních motivů formulací problému jako combinatorial groups [LMPT13].

CMfinder

Hledání zaměřeno na nekódující oblast RNA. Využívá EM algoritmus a kovarianční modely pro RNA struktury. [YWR05] Nástroj byl zařazen do hledání, přestože je výstupem strukturní motiv. Důvodem je snadná integrace využitím DynaMIT MI integrační strategie.

DECOD

Hledání využívá extrakcí počtu všech K-merů pozitívních i negativních sekvencí. Následně je hledána diskriminační PWM pro množství K-merů v pozitivní sekvenci, ale naopak malé v negativní. Korekce výběru stejných posunutých K-merů dekonvolucí. Vyhledání vhodné PWM využívá gradientního algoritmu [HZS+11].

DiNAMO

Řeší problémy s hledáním vzácných motivů hrubou silou a efektivním algoritmem pro hledání IUPAC motivů. Prohledá všechny K-mery, poté sestaví mřížku motivů, kterou redukuje podle MI. K filtraci využívá Fisherův exaktní test. [SNR+18]

DREME

Využito regulárních výrazu k vyhledání motivů. V každé iteraci se zamaskuje nejlepší nalezený motiv a hledání se opakuje. Signifikantnost je testována pro obě sady sekvencí použitím Fisherova exaktního testu [Bai11], který je podle [GSC18] velmi náchylný na nekvalitní negativní sekvence. Program je uzpůsoben na hledání motivů do 8bp bez kratkých inzercí a delecí (INDEL).

GADEM

Kombinuje genetický algoritmus s EM [Li09]. Program vyžaduje nadprůměrnou dobu běhu oproti ostatním testovaným programům.

Gibbs

Množství programů využívá Gibbs sampling proto je duležité uvést celý název programu Gibbs Centroid Sampler obsaženého ve frameworku DynaMIT [Das18]. Jedná se o vylepšenou verzi programu Gibbs Recursive Sampler, která řeší problém s lokálními optimy algoritmů maximalizujících pravděpodobnostní skóre [TNC+07].

GraphProt

Zaměřeno na datové sady CLIP-seq zpracované metodou graph-kernel využívající kombinaci RNA struktur kódovaných jako graf a naučení modelu algoritmem rodiny Support Vector Machine (SVM) [MLCB14]. Program umožňuje vytvoření modelu ze vstupních sekvencí a zadaných parametrů. Hledání parametru vestavěným nástrojem trvalo několik hodin bez známky postupu a proto je při spouštění programu v této práci použito pouze základních parametrů.

HOMER

Program počítá s rozdílným zastoupením GC párů a provádí proto několik normalizačních kroků. Hledat lze motivy různých délek zadaných při startu programu. Není navržen na hledání motivů delších než 16bp, prakticky ale funguje při zadání délky desítek bp [Lab]. Pro uložení výsledků je použit proprietární formát obsahující také skóre log P-value obohacení motivu. Homer optimalizuje motivy na základě hypergeometrického nebo binomického rozdělení.

MDmodule

Implementace konceptu rozděl a panuj rozdělením sekvence na 4 pod-sekvence podle symbolů abecedy A, C, G, T. V rozdělených sekvencích je potom rozhodováno na základě příslušného symbolu zda se jedná o motiv. Algoritmus volí písmeno G s největší vahou z důvodu častého zastoupení v TF. [AA]

MDscan

Prohledávání slov určité délky na obohacená slova spojené s aktualizací PWM. [LBL02]

MEME

Program využívá algoritmus expectation maximization (EM) v modelech konečných směsí pojmenovaný MM [BE94]. Dále je využito heuristik a vícenásobného spuštění algoritmu, umožňující hledání motivů v zadaném rozmezí. Od verze 5.1 je program rozšířen o možnost hledání v módu DMD s využitím přímo background sekvencí namísto jejich Markovovských řetězců jako dříve. Žádný z uvedených frameworku neobsahuje dostatečně novou verzi tohoto nástroje, aby umožňovaly hledat motivy tímto způsobem. Použití nové verze dává značnou výhodu při hledání diferenciálně obohacených motivů.

Před spuštěním nástroje musí být ošetřeny sekvence kratší 8bp. V opačném případě dochází k pádu programu. Chybová hláška uvádí, že postačuje nastavit parametr minimální délky, ale po opětovném spuštění s tímto parametrem došlo k opětovnému pádu programu. Mimo ošetření krátkých sekvencí je vhodné také generovat markovovský model pozadí. Pro generování se využívá program dodávaný společně s MEME spouštěný před hledáním motivů.

MEMERIS

MEMERIS je rozšíření programu MEME pro hledání v sekvencích RNA integrací vypočtené informace o struktuře sekvence. Program využívá nástroje RNAfold pro výpočet struktury. Na rozdíl od MEME, kde je pravděpodobnost výskytu motivu nezávislá na pozici v sekvenci, u programu MEMERIS je pravděpodobnost závislá na vypočtené struktuře. [HPBB06] Využití strukturní informace slouží k nalezení motivů nacházejících se v oblastech, kde struktura RNA napomáhá navázat protein.

MotifSampler

Vylepšení algoritmu Gibbs sampling rozšířením pro data obsahující šum. Rozšíření spočívá v nasazení Markovova modelu pozadí vyššího řádu pro různé organismy [TLM+01]. V kontejneru GimmeMotifs je dodáván generátor, který je při spouštění v rámci této práce využíván.

ProSampler

Prohledá všechny K-mery (výchozí K=8) v obou sadách sekvencí. Identifikuje významné K-mery výpočtem z-score a kombinuje se všemi podobnými méně významnými. Poté je z K-merů sestaven graf podobnosti, nad kterým probíhá Gibbs sampling. [LNZ+18]

RPMCMC

K hledání využívá algoritmus Repulsive parallel Markov chain Monte Carlo (MCMC), který využívá interagující paralelně běžící Gibbs sampling s funkcí zabraňující hledání ve společném lokálním minimu. Výstupem je množství podobných motivů, které jsou shlukovány do nepodobných množin. [IY15] Umí vyhledávat motivy v zadaném rozsahu, který je ve výchozím stavu nastaven na rozmezí 6-14bp s možným maximem až 30bp.

Weeder

Hledání implementací Sufixového stromu [PMMP04]. Program je dodáván s frekvenčními soubory několika organismů.

XXmotif

Hledání nejprve využívá hrubou silou vyhodnocované P-value počátečních motivů. Signifikantní motivy jsou následně prodlouženy paprskovým prohledáváním [HGS+12]. Program jako jeden z mála umožňuje zadat startovní matici s krátkým motivem pro přeskočení prvního stupně hledání a rozšíření poskytnutého motivu. Vyhledávat lze teoreticky motivy dlouhé až 26bp, ale doporučeno je hledání motivů do délky 17bp [HGS+]. Stanovit lze pouze horní hranici délky motivu. Výstup programu v některých případech obsahuje stovky nesignifikantních motivů.

3.0.8 Samostatné nástroje

Nevýhodou využití nástrojů distribuovaných skrze některý z frameworků je méně častá aktualizace těchto nástrojů. Využití novějších verzí je jedním z důvodů proč došlo ke stažení již obsažených nástrojů a jejich spouštění samostatně.

Problémy se spuštěním

Některé aplikace nebylo možné otestovat z důvodu výskytu chyb. Nástroj SSMART produkoval chybu segmentation fault již na ukázkových souborech. Program RNAcontext a od něj odvozený RCK padal při segmentation fault z neznámého důvodu. Framework EMD na testovacích datech produkuje mírně odlišný výstup než je popsáno v manuálu. Vyřazeny byly také nástroje, které vyžadují náročnější úpravy vstupních dat. K testovaní nedošlo u programu RNAmotifs2, který je součástí většího projektu a využívá formát dat, jehož příprava by byla příliš náročná. Velmi odlišná vstupní data využívá také nástroj Teiser. Nástroje RNAcompete a catRapid nebyly posouzeny z

Program	Využit	Komentář	DMD
BaMMmotif	✓	Silně upřednostňuje dlouhé motivy	-
Discrover	✓		✓
RNAcontext a RCK	✗	Chyba segmentation fault	-
rnamotifs2	✗	Zahrnuta	-
sshmm	✗	Hledá motivy kombinující sekvenci i strukturu, singularity nekompatibilní	-
Zagros	✓	Hledá sekvenční motivy s využitím struktury	-

Tabulka 3.4: Zařazení samostatných programů hledajících motivy.

licenčních důvodů. Další desítky nástrojů nebyly hlouběji zkoumány z důvodu malého povědomí o těchto nástrojích v kombinaci s časovou náročnosti procesu kompilace a plnění závislostí programů.

BaMM!motif

Sada nástrojů BaMM!motif obsahuje program PEnGmotif hledající motivy obohacené ve vstupní datové sadě v porovnání s očekávanou hodnotou v sekvencích pozadí. Hledání využívá seznamu sekvencí v IUPAC abecedě optimálních v podmínce, že změna jakéhokoli symbolu sekvence povede ke snížení obohacení vzhledem k modelu pozadí. Nalezené optimální sekvence jsou poté převedeny na PWM, která je dále optimalizována EM algoritmem. [KRG+18]

Discrover

Program reagující na zvětšující se velikost datových sad vylepšuje hledání s využitím skrytých Markovových modelů. Inicializace probíhá skrze množství seedů ve formě IUPAC. Nástroj je navržen na hledání motivů jak v DNA tak i v RNA sekvencích. [MR14]

MDS2

Hledání probíhá nad orientovaným grafem uzlů tvořených dvěma nukleotidy. Motivy jsou tvořeny hledáním cest v grafu a takto nalezené signifikantní k-mery jsou ukládány v pomocné tabulce. Nakonec jsou motivy shlukovány podle Pearsonova korelačního koeficientu. [GSC18] Program je zaměřen na hledání protein-RNA vazebných míst, které mohou být na rozdíl od transkripčních faktorů mnohem kratší a to okolo 3-5bp. Určen převážně na krátké RNA sekvence. Na dlouhých sekvencích došlo k nálezu velkého množství

motivů v některých případech dosahujícím jednotek tisíců. Mimo výsledné motivy program poskytoval pouze nulové hodnoty P-value. Jako jeden z mála programů neumožňuje nastavit mimo dvě sekvence a délky hledaných motivů žádné dodatečné parametry.

ssHMM

Využívá Markovovy skryté modely kombinující sekvenci a strukturu. Pro dodatečné kroky při zpracování zadané struktury je využit Gibbs sampler. [HKO+17] Výstupem je velmi originální grafické zobrazení naučeného modelu poskytujícího informaci zároveň o struktuře a sekvenci. Zajímavostí je podpora dvou nástrojů používaným k výpočtům struktury.

Zagros

Program umí hledat motivy v sekvencích s využitím nebo bez využití souboru struktur. Strukturní data mohou být vygenerována ze sekvencí nástrojem thermo využívajícím McCaskill algoritmus přímo importovaný z RNA Vienna balíčku. Implementace vlastního obalu namísto použití RNAfold je zdůvodněno snížením náročnosti výpočtů vynecháním nepotřebných informací. [BSPSU14b] Hledání motivů funguje na podobném principu jako u programu MEME s EM algoritmem rozšířeným o strukturní informaci. [BSPSU14a] Program neumožňuje DMD v sekvencích a hledá pouze motivy maximálně 12bp dlouhé ať už s využitím struktury nebo bez. Zadat lze hledání motivů jediné délky, která je ve výchozím stavu nastavena na hodnotu 6.

Kapitola 4

Konverze formátů sekvenčních motivů

4.1 Implementace nástroje lead2gold

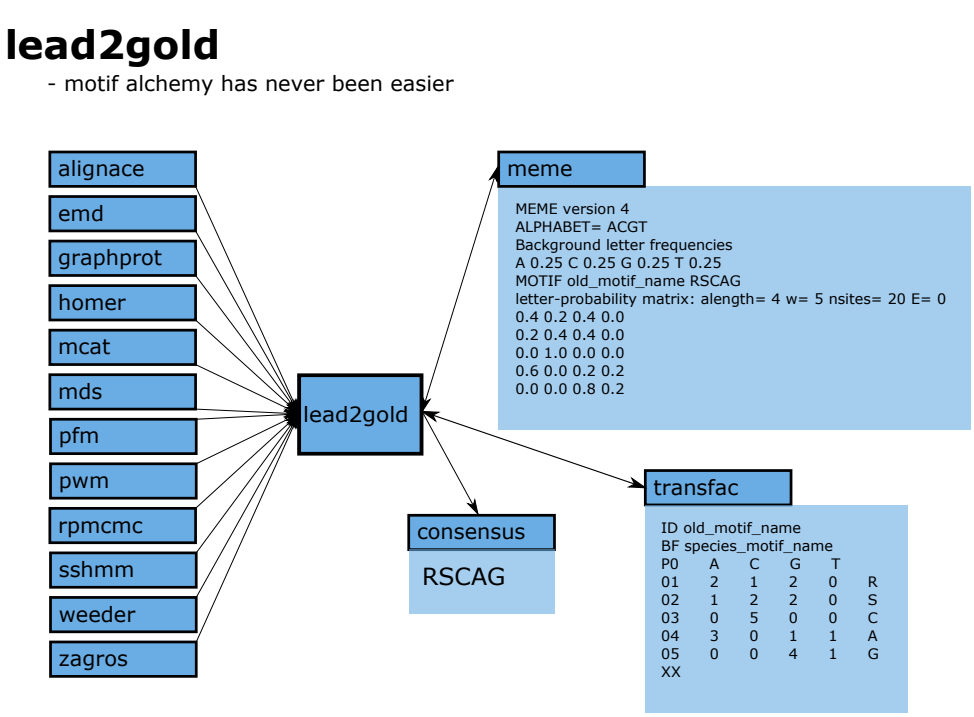
Práce s množstvím programů vyžaduje časté převody sekvenčních motivů pro jejich porovnání a další zpracování. Pro konverzi je hojně využíváno také funkcí implementovaných uvnitř GimmeMotifs. Závislost převodu motivů na tomto frameworku ale zapříčinila nutné úpravy po nálezu několika chyb vedoucím k nerozpoznání motivů. Dále chybí některé programy, které byly v této práci samostatně testovány. Z těchto důvodů došlo k implementaci vlastního nástroje pro převod motivů.

Na rozdíl od projektu GimmeMotifs, který integruje převod několika nástrojů do stejné převodní funkce, lead2gold přistupuje ke každému programu individuálně. Přístup na bázi nástroje namísto podobnosti formátu má za cíl extrakci většího množství dat převáděného motivu. Pozornost je soustředěna převážně na programy produkující určité skóre.

Před implementací byla zvážena integrace kontejneru nástroje universalmotif, který umožňuje manipulaci s množstvím běžně používaných formátu [JM18]. Přednost ale byla dána vlastní implementaci z důvodu snadného převodu motivů méně známých nástrojů a zbavení se závislosti na jazyce R, který v kombinaci s jazykem Python používaným napříč projektem zapříčiňuje razantní navýšení velikosti kontejneru.

Výsledný program implementovaný v jazyce Python převádí motivy využívané v rámci práce na formáty běžně používané. Program úmyslně vyžaduje jen velmi malý počet knihoven. Cílem je usnadnění implementace v množství projektů potýkajících se s potřebou převodu nalezených motivů. V případě použití programu v kontejneru je dosaženo značné úspory místa zmenšením prostředí na méně než desetinu objemu dat v porovnání s kontejnerem *biocontainers/bioconductor-universalmotif*.

Při převodu lze pomocí přepínače zvolit, zda se má opravit motiv obsahující nulový součet výskytů mezi nukleotidy přidáním pseudocount hodnoty. Motivy



Obrázek 4.1: Převodník známých formátů sekvenčních motivů. Znázornění podpory převodu mezi různými formáty nástrojů.

s touto vlastností byly pozorovány při zpracování výsledků ve framoworku DynaMIT s výskytem mimo koncové nukleotidy motivu.

Motivy nesoucí informaci umožňující řazení lze seřadit a vybrat omezený počet motivů. Pokud není k dispozici informace, podle které by bylo možné seřadit motivy, dochází k výběru motivu z počátku seznamu. Výsledky programů bývají obvykle řazeny od nejlepších po nejhorší.

Převodník umožňuje výstup jen v několika málo možných formátech z důvodu malého využití vstupních proprietárních formátů. Program může být zařazen do linuxové pipeline nastavením vstupního/výstupního souboru jako standardní vstup/výstup.

Sekvenční konsenzus využívající IUPAC kódy nebyl exaktně definován [CB85]. Existuje proto více implementací a také rozšíření definované abecedy [Joh10]. Pro sjednocení generovaného výstupu při převodu do motivu tvořeného IUPAC kódy byla do projektu zakomponována konsensus generující funkce z projektu GimmeMotifs [BvH18].

Kapitola 5

Příprava dat pro nástroje hledající motivy

5.1 Biologický kontext motivů

Nástroje hledající motivy jsou specializovány jen na samotné vyhledávání motivů bez podpory přípravy dat do podoby vhodné pro zpracování. Při použití sekvenačních dat tak vzniká krok se značnou volnosti v celém procesu, který poté udává důležitý kontext nalezených motivů. Zmíněným chybějícím krokem je zpracování obsáhlých sekvenačních dat v závislosti na informaci o které budou tyto motivy vypovídat. Výstupem je sada sekvencí vztahujících se k požadované vlastnosti ve formátu podporovaném hledajícím nástrojem.

Často sledovanou informací je například rozdíl v expresi oproti očekávané hodnotě. Vybírány jsou v tomto případě sekvence se vztahem k pozorovanému rozdílu exprese, které mohou být dále roztříděny do skupin pro hledání rozdílně obohacených motivů. Dva soubory tvořené rozdělením sekvencí do dvou množin jsou dále nazývány jako primární (pozitivní) a kontrolní (negativní).

1. Biologický experiment
2. Sekvenace
3. Analýza
4. **Příprava dat**
5. Hledání motivu

Dodaná zkoumaná datová sada pochází z experimentu zabývajícím se translačními iniciačními faktory. Součástí dodaných dat je také analýza obsahující informace o expresi jednotlivých genů v sekvenačních datech. Na základě těchto dat jsou v této práci vyrobeny soubory sekvencí, ve kterých následně probíhá hledání.

Podle obsažené analýzy jsou připraveny sekvence oblasti UTR vztahující se k rozdílům v expresi udávané analýzou. Očekává se, že tímto způsobem vytvořené sekvence budou obsahovat sekvenční motivy vztahující se specificky k funkci translačních iniciačních faktorů z rodiny eIF4E. Pro přípravu popi-

sovaných sekvencí nebyl nalezen již hotový postup. Neexistující hotové řešení vedlo k implementaci dvou nezávislých postupů, kterými lze cílové sekvence generovat.

5.2 Formáty využívané při zpracování sekvencí

5.2.1 Formát FASTQ

Sekvence pocházející ze sekvenace jsou dodávány ve formátu FASTQ společně s kvalitou čtení pro každý nukleotid čteného řetězce. Soubory FASTQ neobsahují genomové souřadnice sekvencí a nejsou proto vhodným formátem pro další přímé zpracování.

5.2.2 Formát BAM

Pro doplnění informace o genomových souřadnicích přečtené sekvence se po sekvenaci provádí tzv. mapping, který najde alignment ve zvoleném referenčním genomu organismu. Takto mapované sekvence jsou již zahrnuty v dodané datové sadě. Soubory pochází z mapovaní vykonaného nástrojem BWA. Sekvence společně s jejich oblastí ve známém genomu jsou uloženy v běžně používaném formátu BAM. Jedná se o binární a komprimovanou verzi ekvivalentního textového formátu SAM (Sequence Alignment Map) [src20].

5.2.3 Formát FASTA

Při práci se sekvencemi je dále často využíván formát FASTA. Textová data lze snadno číst díky jednoduché podobě tohoto formátu. Řádky obsahující znak '>' značí název a popis sekvence. Následující řádky obsahují příslušnou sekvenci RNA, DNA nebo proteinu [Wik20a]. Soubory FASTA běžně neobsahují informace o souřadnicích ani kvalitě čtení.

Listing 5.1: Ukázka obsahu souboru FASTA

```
>SEKVENCE_ID1
GGGACCAGAGCGAGAAGCGGGGACC
>SEKVENCE_ID2
TATCTCAGAGATGTTAACTGTCCT
```

5.2.4 Formát BED

Některé nástroje místo souborů sekvencí pracují na základě souřadnic úseků v referenční sekvenci. Pro záznam úseků se využívá mimo jiné formát BED.

Záznamy jsou ukládány na samostatné řádky. Každý obsahuje minimálně 3 sloupce označující chromozom, začáteční pozici a koncovou pozici. Dále je specifikováno až 9 dalších sloupců pro dodatečné informace. [beda] Soubory obsahující všechny definované sloupce jsou označovány jako formát BED12 [bedb].

5.3 Uměle vytvořená datová sada

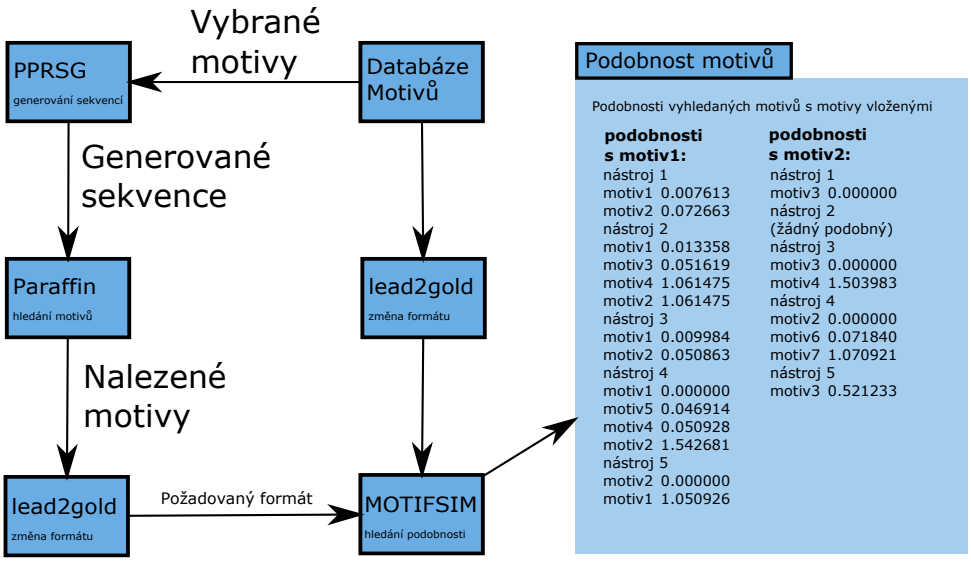
Při spouštění velkého množství nástrojů se ukázalo, že rozdíly v nalezených motivech jsou větší než se očekávalo. Hledání s použitím soustavy zvolených nástrojů by vedlo k velmi obtížně interpretovatelným výsledkům. Problém s posouzením kvality a chování programů mohl být vyřešen použitím datové sady určené pro benchmarking těchto nástrojů. Testování by se ale v případě použití velkých souborů stalo výpočetně velmi náročným procesem, který by zároveň vyžadoval různé datové sady pro otestování všech zkoumaných parametrů.

Za účelem otestování pouhé základní funkčnosti nalezených nástrojů byl vytvořen dataset obsahující pseudonáhodné sekvence a známý počet vložených motivů. Pro vytvoření tohoto testovacího datasetu bylo implementováno vlastní řešení generující pseudonáhodné sekvence. Výsledný program nazvaný PPRSG umožňuje specifikovat několik parametrů. Zvolit lze například délku sekvencí nebo vkládaný motiv ve formátu MEME. Důležitou funkcí je generování sekvencí podle dodaného modelu obsahujícího pravděpodobnosti výskytu jednotlivých nukleotidů. Pro naučení tohoto modelu program využívá Markovovy řetězce volitelného řádu použité na sekvence zadaného souboru ve formátu FASTA. Pro datové sady používané v této práci došlo k vytvoření modelu založeném na souboru obsahujícím referenční sekvence UTR. Naučený model poté umožňuje generovat množství sekvencí různě zvolených délek. Dále lze generovat také sekvence různých délek pozorovaných při učení na zadaném souboru. Vytvořeno tak bylo několik souborů obsahujících sekvence parametrů nesoucích některé vlastnosti typické právě pro zkoumanou oblast 5’ UTR nebo 3’ UTR.

Naučený model nultého řádu zachycuje zvýšený podíl GC/AT pro UTR 5’ oblasti s hodnotou 60,4% GC. Naopak u 3’ UTR modelu je zastoupení nižší a to 42,8% GC. Naučené parametry se tedy velmi blíží k popisovaným hodnotám 60,6±12% GC a 42,4±11% GC u těchto oblastí [ZKCB04]. Drobná odchylka je pravděpodobně způsobena filtrací krátkých úseků. Při specifikaci souboru obsahujícího několik motivů lze lehce specifikovat četnost výskytu jednotlivých motivů odděleně pro primární i kontrolní sekvenci.

5.3.1 Postup testování aplikací

Pro zrychlení postupu testování byla sestavena pipeline testující schopnost nástrojů odhalit motiv v pseudonáhodné sekvenci. Tento krok se ukázal jako velmi důležitý pro odhalení řady chyb skládajících se převážně ze špatně nastavených parametrů spouštěných programů.



Obrázek 5.1: Vývojový diagram postupu testování aplikací hledajících motivy.

Popisovaná pipeline Obr. 5.1 je sestavena z několika nástrojů implementovaných v rámci této práce a nástroje MOTIFSIM [TH18]. V prvním kroku dochází ke spuštění PPRSG, který na základě zvolených parametrů a motivů z databáze sestaví požadované soubory pozitivních a negativních sekvencí. Po dokončení generování je pomocí nástroje paraffin paralelně spuštěno libovolné množství nástrojů. Po doběhnutí posledního hledajícího programu dochází k převodu nalezených motivů na formát MEME. Z důvodu pochybné podpory tohoto formátu nástrojem MOTIFSIM ale dochází k dalšímu převodu nalezených i databázových motivů do formátu, který je zmíněným nástrojem stabilně podporován. V konečném kroku jsou motivy zvlášť pro každý program porovnány a seřazeny. Tento proces je realizován vytvořením dvou datových sad, kdy jedna obsahuje motiv z databáze a druhá všechny vyhledané motivy. Takto vytvořené datasety MOTIFSIM vyhodnotí přiřazením skóre podobnosti každé mezi-datasetové dvojici motivů.

Celý postup vedoucí k výpočtu skóre probíhá pouze s využitím sekvenční informace. Přidání strukturních motivů do generovaných sekvencí nebylo pro vysokou časovou náročnost a stávající formulaci zadání práce realizováno. Chybějící smysluplná struktura ale znevýhodňuje programy hledající motivy, které této informace využívají. Při výběru nástrojů byl tento fakt zohledněn

a vypočtené skóre bylo v této skupině využito pouze pro ověření správného výpočtu background modelu.

5.3.2 Soubory sekvencí testovacího datasetu

Při instalaci a používání nástrojů hledajících motivy bylo zpozorováno neočekávané chování a několik možných problémů, které by mohly negativně ovlivnit výsledky hledání. Soubory testovacích sekvencí jsou proto navrženy tak, aby upozornily na programy, které trpí některým z pozorovaných problémů. Častým problémem je například výskyt krátkých sekvencí, který v některých případech vede až k pádu aplikace.

Množství problémů zřejmých už při základním seznámení s některými programy hledajícími sekvenční motivy vedlo k vytvoření několika testovacích souborů sekvencí. Účelem je otestování programů na základní požadované funkce a zahrnutí do výpočtů na zkoumané datové sadě jen při překonání určité úrovně. Tímto je zajištěno ušetření zdrojů výpočetního clusteru a redukce produkovaných výsledků nízké kvality, které by bylo nutné dále zpracovat.

Vygenerované datasety pomohly otestovat nastavení velkého množství spravovaných programů. Častým nedostatkem byl chybějící argument, což by na reálných datech bylo mnohem hůře odhalitelné pochybení. Dalším častým nedostatkem byl chybný převod motivů implementovaný v testovaných sadách nástrojů DynaMIT a GimmeMotifs.

	5' UTR	CDS	3' UTR
Median délky	203 bp	1278 bp	938 bp
Průměrná délka	259 bp	1663 bp	1470 bp
Standardní odchylka	228 bp	1901 bp	1620 bp

Tabulka 5.1: Délky oblastí mRNA lidského genomu. [PCA+16]

Sekvence délky 30bp

Dataset s krátkými sekvencemi byl vytvořen z důvodu návrhu některých nástrojů na sekvence krátkých čtení. Tento dataset slouží pouze pro ověření základní funkčnosti testovaných programů. Soubor obsahuje 700 sekvencí délky 30bp a 50 vložených motivů.

Sekvence délky 210 bp

Dataset obsahující soubor 100 sekvencí délky 210bp a 50 vložených motivů pro porovnání s krátkými sekvencemi. Dále dataset obsahuje soubor stejného počtu sekvencí, ale se zvýšeným počtem motivů na 200. Délka byla inspirována

tabulkou obsahující průměrnou délku 5' UTR, které je v této práci přikládána největší pozornost Tab. 5.1.

V průběhu testování se ukázalo, že nástroj Weeder obsažený ve třech projektech shlukujících nástroje nenalezl žádný vložený motiv. K špatným výsledkům mohlo částečně přispět použití předem generovaných frekvenčních souborů, které neodpovídají sekvencím zkoumané datové sady. Z důvodu velmi špatných výsledků a primárního účelu tohoto programu nebyla dále zkoumána možnost generování vlastních frekvenčních souborů [pav].

Test diskriminačního hledání

Pro hledání diskriminační analýzou bylo zapotřebí ověřit, že program hledá motivy obohacené pouze v požadované datové sadě. Dataset obsahuje sekvence podobné jako dataset předchozí, ale navíc jsou vloženy také motivy spadající zároveň do souboru pozitivních i negativních sekvencí.

S použitím vytvořené pipeline Obr. 5.1 byla ověřena schopnost hledat motivy obohacené pouze v primární datové sadě. Sledován byl *Motiv1* nacházející se pouze v pozitivních sekvencích a *Motiv2* obsažený v pozitivních ale zároveň také v negativních sekvencích. Programy byly hodnoceny na základě skóre podobnosti ke sledovaným motivům. V případě že program našel motiv vyskytující se v obou sekvencích došlo k jeho penalizaci, naopak nález motivu vyskytující se pouze v pozitivní sekvenci znamenal zlepšení výsledku programu.

Výsledné hodnoty v tabulce Tab. 5.2 potvrdily správnost seznamu DMD nástrojů Tab. 3.3 Tab. 3.4 vytvořeného při rešerši. Neodpovídal pouze výsledek programu HOMER. Na základě této nesrovnalosti došlo k odhalení chybně uvedeného parametru v nápovědě frameworku DynaMIT. Po nápravě této chyby došlo k dosažení nejlepšího možného výsledku tohoto testu a vyřešení zmíněné nesrovnalosti.

Na základě výsledků všech proběhlých testů bylo rozhodnuto o vyřazení několika nástrojů. Program RPMCMC produkoval příliš velké množství motivů, kde žádný nepřesahoval podobnostní skóre programu MEME. Navíc hledání zabralo déle než minutu oproti pěti vteřinám u konkurenčního programu MEME. V testu s background sekvencemi dopadl nejhůře program improbizer, který vyhledal motiv vyskytující se v obou sadách a obohacený motiv vůbec nezpozoroval.

Překvapivě dobrý výsledek ale podal nástroj Bioprospector, který není typickým nástrojem využívaným pro DMD.

Program	Motiv1	Motiv2	Výsledek
dynamit/memeris	0,007	0	0,007
dynamit/mdscan	0,009	1	−0,990
dynamit/meme	0,013	0	0,013
dynamit/homer	0	0,042	−0,042*
dynamit/graphprot	0	1	−1
dynamit/glam2	0	0,515	−0,515
dynamit/previous	0	0	0
gimme/mdmodule	0,034	0,5	−0,465
gimme/meme	1	0	1
gimme/dreme	0	1	−1
gimme/gadem	0	1	−1
gimme/weeder	0,509	0,013	0,496
gimme/motifsampler	1	0,041	0,958
gimme/improbizer	1	0	1
gimme/bioprospector	0	0,517	−0,517
gimme/AMD	0,576	0,565	0,011
gimme/homer	0,011	0,063	−0,052*
gimme/xxmotif	0,005	0,040	−0,03
gimme/dinamo	0,5	1	−0,5
mcat	0,007	0,004	0,002
bamm	0	1	−1
discrover	0,029	1	−0,970
dreme	0	1	−1
EMD	0	1	−1
graphprot	0	1	−1
homer	0,011	0,046	−0,035*
meme	0	1	−1
MDS2	0,01	0,043	−0,030
rpmcmc	0,053	0,049	0,003
sshmm	0,014	0	0,014
xxmotif	0	1	−1
zagros	0	0	0
weeder2	0,509	0,013	0,496

Tabulka 5.2: Výstup testu diskriminačního hledání. U sloupce Motiv1 a Výsledek menší hodnoty znamenají lepší splnění problému. Sloupec Motiv1 a Motiv2 obsahuje hodnoty podobnosti pro nejpodobnější motiv. Vysoká hodnota podobnosti značí nepodobné motivy. Sloupec výsledku vyjadřuje nakolik program splnil definovaný problém. *Chyba v argumentu.

Sekvence různých délek

Některé nástroje (např. XXmotif) jsou navrženy pro práci se sekvencemi stejné délky. Tento set sekvencí testuje zda nástroje podporují sekvence různých délek. Vygenerované soubory obsahují zároveň krátké i velmi dlouhé sekvence, které způsobují pád několika zkoumaných programů. Ukázkou je například pád i velmi známého programu MEME a nebo od něj odvozeného RPMCMC.

Rozdílné množství sekvencí

Programy byly testovány také na schopnosti zpracování velkého počtu sekvencí. V případě výpočtů trvajících jednotky hodin došlo v některých případech k vyřazení. Potvrdil se například příliš dlouhý výpočetní čas programu DECOD obsaženého v sadě nástrojů MCAT [YRG+19].

Sekvence s dlouhými motivy

Jedním z cílů bylo nalézt co nejdelší signifikantní motivy. Některé nástroje mají limity v maximální délce motivu, který mohou vyhledat. Cílem tohoto datasetu bylo ukázat na schopnosti programů v hledání dlouhých motivů.

5.4 Využití zkoumané datové sady

Po výběru nástrojů splňujících požadované parametry proběhlo hledání motivů ve zkoumané datové sadě. Dodaná data neobsahují sekvence ve správném formátu a není je tedy možné rovnou využít pro hledání motivů. Z tohoto důvodu je zde popsáno mimo jiné také zpracování sekvencí do formátu pro hledání běžně používaného. Dále popsané kroky jsou realizovány implementací několika skriptů zabývajících se hromadnou přípravou množství souborů. Výsledkem jsou sekvence vytvořené ze souborů mapovaných čtení sekvenační metody RNA-seq NG s využitím dodané analýzy provedené nad těmito daty. Další prozkoumaná metoda se zabývá dotazováním na sekvence podle poskytnuté analýzy z veřejně přístupného API.

5.4.1 Zkoumaná datová sada a další zdroje dat

Data sekvenace provedených experimentů

Hledání je zaměřeno na data, která jsou výsledkem 7 experimentů a jejich opakování ve třech biologických replikátech. Celkem tedy datová sada obsahuje 21 souborů sekvencí, které jsou výstupem jednotlivých sekvenací. Získané sekvence jsou uloženy v podobě komprimovaných souborů FASTQ.

Tato surová data nacházející se v souborech s příponou .txt.gz byla namapována na referenční genom a výsledné soubory jsou také součástí zkoumané datové sady. K namapovaní sekvencí na aktuálně běžně používaný genom GRCh38 byl použit program BWA a výstupní namapovaná data jsou uložena v binárním souboru mapovaných sekvencí BAM. Pro každý soubor ve formátu FASTQ tak můžeme v datasetu nalézt odpovídající soubor ve formátu BAM.

Neočekávanou vlastností specifickou pro soubory BAM obsažené v dodané datové sadě je chybějící předpona při označování chromozomů. Běžně je využito předpony chr, která v těchto souborech chybí a namísto označení *chrN* se tedy setkáme s pouhým číselným označením. Tento problém se vyskytuje při použití referenčního souboru neobsahujícího předponu. Použit byl pravděpodobně referenční soubor stažený ze serveru Ensembl [Mm].

Analýza a hodnoty exprese

Nad sekvenovanými daty proběhla také analýza zkoumající obohacení genů v jednotlivých experimentech. Změny výskytu byly získány nástrojem DESeq2 a jsou uloženy v souboru formátu TSV. Výsledek analýzy celkem obsahuje 19 870 řádků, kde každý řádek náleží unikátnímu genu specifikovanému názvem a identifikátorem v prvním sloupci. Dalších šest sloupců uchovává změny exprese oproti prvnímu experimentu vyjádřené jako log2 fold change. Použití logaritmu na násobky změny exprese rozprostřelo hodnoty symetricky okolo nuly pro jednodušší interpretaci a následné zpracování. Dále lze v souboru najít sloupec obsahující hodnotu FDR.

Referenční genom GRCh38

V případě že využíváme namísto samotných sekvencí jejich genomické souřadnice, musíme pro získání konkrétní sekvence znát přesnou podobu použitého genomu. K tomuto účelu se využívá dohodnutý referenční genom zkoumaného organismu. Lidský referenční genom obsahuje kompletní sekvence všech chromozomů. O množství těchto referenčních genomů a jejich aktualizace se stará Genome Reference Consortium [Gena].

Volba genomu pro hledáni motivu byla stanovena na, v době psaní této práce, nejnovější minor verzi GRCh38.p13. Důvodem pro volbu posledního vydaného genomu je neznámá verze použitá pro mapování ve zkoumané datové sadě. Referenční soubor GRCh38 (UCSC značení hg38) je aktualizován systémem opravných záplat tzv. patches. Při aktualizaci referenčního genomu nedochází ke změně souřadnic, protože nové úseky jsou přidávány jako speciální úseky mimo chromozomy sestavené při vydání hlavní verze [Genb]. Při použití nejnovějšího referenčního souboru proto nehrozí problémy z důvodu rozdílných souřadnic a zároveň soubory BAM nemohou obsahovat sekvence namapované na místo, které v referenci není obsaženo.

Anotační soubor referenčního genomu

Existence referenčního genomu umožňuje úsporný popis oblastí s využitím pouhých tří sloupců obsahujících chromozom, start pozici a stop pozici. Přidáním dalších sloupců vyjadřujících informaci o zaznamenaných oblastech vzniklo několik podobných formátů jako například GTF, GFF a BED. Při převodu některého z jmenovaných formátu nebo jejich zpracování je vhodné namísto formátování textu zvolit prověřené nástroje. Rozdíly se například vyskytují i v definici startovních a koncových hodnot [Tyn16].

S těmito formáty se můžeme setkat při práci s anotačními soubory obsahujícími známé pozice různých typů oblastí sekvencí. Soubory běžně obsahující pozice genů, transkriptů a kódujících sekvencí dodává hned několik organizací. Pro práci se zkoumanými oblastmi UTR bylo využito anotačního souboru spravovaného projektem GENCODE. Tento anotační soubor obsahuje nejvíce anotovaných oblastí společně s více sloupci poskytujícími doplňující informace F.1.

5.4.2 Výběr zkoumaných oblastí

Analýza genové exprese na úrovni mRNA v buněčných liniích HEK293 s rozdílnou hladinou produkce jednotlivých translačních iniciačních faktorů z rodiny eIF4E vedla k výběru genů, jejichž exprese se liší vzhledem ke kontrolnímu souboru. Kritériem pro zařazení konkrétních záznamů do další analýzy byla hodnota FDR nižší než 0,01. Filtrací tímto kritériem bylo vybráno 5 045 genů.

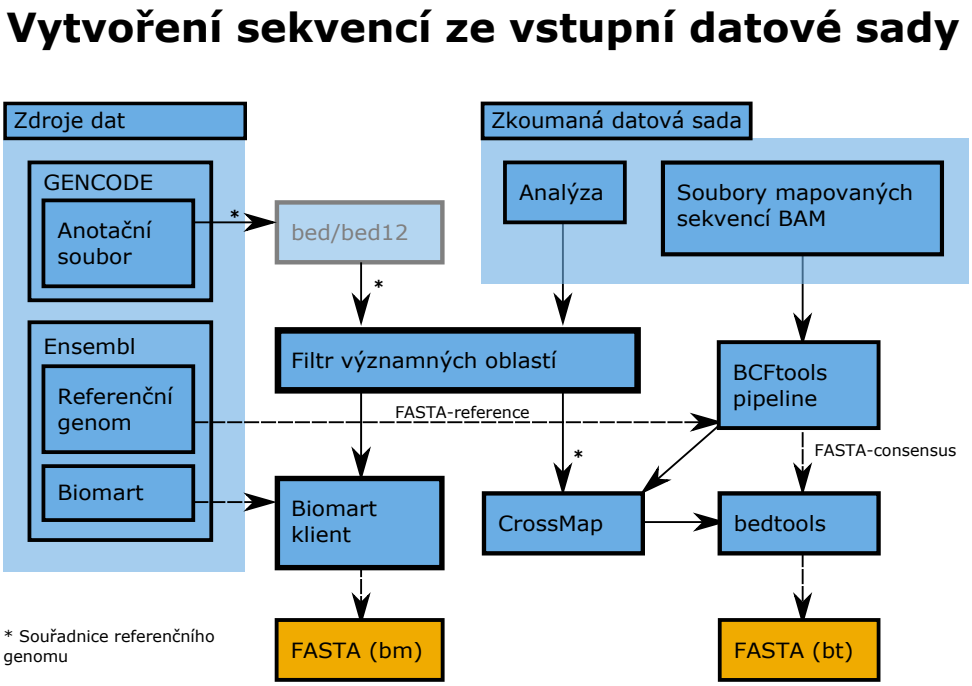
Geny zařazené do dalšího kroku výběru byly seřazeny podle hodnoty násobku změny exprese mRNA vzhledem ke kontrolní buněčné linii. Tímto vznikl seřazený seznam genů pro každou zkoumanou buněčnou linii. Dále bylo potřeba stanovit počet genů, které budou dále zkoumány a z toho vyplývající velikost primárního souboru určeného k hledání motivů. Namísto stanovení pevně dané velikosti byly vytvořeny různě velké soubory, obsahující N genů, kde $N \in 8, 16, 32, 64, 128, 256$. Pro každé N tak byl vytvořen soubor obsahující sekvence N genů vybraných ze začátku seřazeného seznamu. Tímto způsobem bylo vytvořeno 6 souborů genů s pozitivně zvýšenou hladinou exprese.

V dalším kroku došlo k obrácení řazení seznamu genů a při využití stejného postupu generování 6 souborů se sníženou hladinou exprese. Vytvořeno tak bylo 6 datových sad tvořených celkem 12 soubory, kde každá dvojice datové sady odpovídá stejnému počtu vybraných genů.

Každá taková datová sada vytvořená implementovaným skriptem byla označena identifikátorem. Tento identifikátor vznikl na základě parametrů zadaných při volání implementovaného skriptu. Konkrétně se jedná o název, počet genů v každém souboru a zvolené FDR.

Sekvence v souborech datových sad jsou uloženy ve formátu FASTA. Z důvodu přidání různých zdrojů sekvencí byl identifikátor rozšířen také o zdroj dat a zvolenou oblast sekvence. Mezi podporované zdroje dat patří *biomart*, *bt_reference* pro soubor referenčního genomu a *bt_bams* pro sekvence vycházející z dat sekvenace. Volitelné cílové oblasti se liší pro každý zdroj dat. Podporované oblasti pro každý zdroj dat je uveden v příloze F.1. Tímto způsobem připravená datová sada je dále zpracovatelná v implementovaném programu hledajícím motivy zadáním příslušného identifikátoru.

5.4.3 Příprava sekvencí ve formátu FASTA



Obrázek 5.2: Vytvoření sekvencí ze vstupní datové sady.

Stav v oblasti hledání podle zadaných souřadnic

Jen velmi malé množství nástrojů hledajících motivy umožňuje specifikovat požadované geny pro diferenční analýzu. Jedním z mála je například framework GimmeMotifs, kde byla tato funkce otestována. Bohužel se ukázalo, že program stahující genom potřebný pro tuto funkci nefunguje pro použité prostředí a specifikace vlastního genomu obsahuje několik chyb, které bylo třeba opravit přidáním výpočtu délek sekvencí v genomu.

Nestabilita tohoto přístupu ukazuje na malou četnost využívání souřadnic namísto sekvencí při hledání motivů i u nástrojů, které tuto metodu podporují. Z uvedených důvodu byl tento systém opuštěn ve prospěch generování souborů obsahujících sekvence.

Sekvence Ensembl BioMart

Snadno přístupným zdrojem množství popsaných tzv. anotovaných sekvencí je veřejně přístupné API Ensembl BioMart [KKH+11]. Webové API umožňuje stažení sekvencí pro požadovaný seznam genů s možností volby oblasti sekvence. Webový klient umožňuje jednoduše sestavit požadavek na požadovanou oblast 5' UTR, 3' UTR libovolného počtu genů. Z důvodu nutnosti automatizace stahování dat z RESTful API bylo využito doporučeného BioMart PERL API scriptu [Ens].

Před použitím dat z nástroje BioMart je vhodné provést alespoň vizuální kontrolu získaných sekvencí. Jedním z problémů, na které lze snadno narazit, jsou duplicitní názvy obdržených sekvencí nebo sekvence obsahující chybová hlášení. V rámci práce byl sestaven kontejner pro použitý skript a implementovány skripty řešící nejběžnější problémy, které se vyskytovaly při hromadném stahování.

Automatizace procesu stahování umožnila vytvořit množství datových sad podle různě zvolených parametrů. Vytvořeny byly datové sady pro různé počty genů, požadované oblasti nebo volbou jiného experimentu. Tento proces získávání sekvencí byl velmi rychlý. Limitující faktorem byla pouze rychlost připojení k síti internetu.

Sekvence genů získané z referenčního genomu

Rychlejší alternativou k BioMart API bez nutnosti přístupu k síti je dolování sekvencí z referenční sekvence podle známých anotovaných oblastí. Na rozdíl od předchozí metody s využitím API je nutné před dolováním sekvencí ještě převést seznam vybraných genů na souřadnice odpovídající požadovaným oblastem. Převod zahrnuje filtraci a konverzi formátu anotačního souboru.

První krok převodu byl implementován nástrojem grep, který umožňuje filtrovat využívaný anotační soubor seznamem zvolených genů. Ještě před filtrací na základě genů dochází k rozdělení anotačního souboru podle oblastí genů pro urychlení opakované filtrace. Dalším krokem je převedení anotovaných úseků do formátu BED. Pro zajištění kvalitní konverze byl zvolen nástroj gff2bed [NKR+12] obsažený v rozsáhlém projektu zabývajícím se analýzou genomu BEDOPS. Konvertovaná data byla poté nástrojem *sed* zbavena předpony *chr* pro zajištění případné kompatibility s daty dodaného datasetu.

Referenční genom a převedený soubor požadovaných úseků nyní ve formátu BED dostačují k volání funkce getfasta nástroje bedtools [QKb]. Tento nástroj byl zvolen pro velké množství obsažených funkcí sekvenčních manipulací [QKa]. Nástroj je velmi stabilní a dobře dokumentován. Volání funkce getfasta velmi rychle navrátí sekvence odpovídající cílovým oblastem vybraných genů.

Sekvence ze souborů BAM

Sekvence referenčního genomu se v některých úsecích liší od dat získaných sekvenací. Z tohoto důvodu byl implementován postup pro generování sekvencí jejichž zdrojem je právě soubor mapovaných sekvencí ve zkoumané datové sadě. Zvolený postup využívá skript s nástrojem bedtools implementovaný pro výběr sekvencí z referenčního genomu. Rozdíl spočívá v pouhé náhradě referenční sekvence za sekvenci konsenzuální. Tato konsensuální sekvence je vytvořena uplatněním pozorovaných významných změn v sekvenačních datech na referenční sekvenci genomu.

K vytvoření konsensuální sekvence byla použita sada nástrojů bcftools obsahující množství funkcí zaměřených na variant calling a manipulaci souvisejících souborů. Vytvoření sekvence využívá řadu po sobě jdoucích funkcí mpileup, call, filter, consensus (bcftools pipeline Obr. 5.2) inspirovaných oficiálním návodem [rc] a manuálem [bcf].

Variant calling je proces detekující SNV (jedno-nukleotidové varianty) a InDel (malé inzerce a delece) [KSD19]. Soubory obsahující detekované změny jsou uloženy ve formátu VCF (Variant Call Format) nebo binárním ekvivalentu BCF.

Vytvoření sekvence proběhlo ve více krocích z důvodu vysoké výpočetní náročnosti postupu a potřeby ladění parametrů v pozdějších krocích. Výpočetně nejnáročnější částí je samotný variant calling. V dalším kroku jsou data filtrována podle kvality a hloubky čtení. Filtrovaná data jsou poté zpracována funkcí consensus, která vytvoří chain soubor a finální sekvenci podobnou referenčnímu genomu.

Při vytváření konsensuální sekvence jsou uplatňovány detekované odlišnosti z kroku variant calling. Uplatnění delece nebo inzerce a související posun nukleotidů znamená ztrátu možnosti výběru sekvence podle souřadnic referenčního genomu. Nástroj bcftools s tímto jevem počítá a pomocí přepínače funkce consensus umožňuje vytvořit soubor formátu chain [cha], který uchovává informace o mezerách mezi dvěma zpracovávanými genomy.

Soubor chain lze použít pro převod souřadnic mezi sestavenými genomy. Tento proces opravy souřadnic se nazývá liftover, stejně jako jeden z programů řešící tento problém. Pro převod anotovaných oblastí Obr. 5.2 byl do skriptů zakomponován již delší dobu aktivně vyvíjený nástroj CrossMap. Výhodou je jistá [BSS] podpora souboru formátu chain, protože byl pro tento formát navržen [ZSW+13].

Po převedení souřadnic je možné využít stejný postup jako při dolování sekvencí z referenčního genomu. Pro ověření správnosti postupu byla použita kombinace nástrojů diff, bcftools view a také prohlížeče genetických sekvencí s grafickým prostředím IGV. Kontrolou bylo potvrzeno, že sekvence pocházející z reference a souborů BAM se liší pouze v očekávaných úsecích.

5.4.4 Hledání motivů v získaných sekvencích

Po vygenerování sekvencí z API Biomart, reference i souboru BAM proběhlo hledání motivů ve všech třech vytvořených datových sadách. Výsledné motivy ukázaly že rozdíly mezi sekvencemi ze souboru BAM a reference nejsou dostatečně časté na to aby ovlivnily výsledné motivy. Neočekávaně velké změny ale nastaly mezi sekvencemi staženými z API Biomart oproti dvěma získanými nástrojem bedtools. Zdrojem velkých změn signifikantností některých motivů bylo rozdělení exonů v sekvencích generovaných nástrojem bedtools. K rozdělení dochází z důvodu použití formátu BED namísto BED12 požadovaného pro vytvoření sekvencí s návazností. Jedná se o chybějící část postupu Obr. 5.2, která může být v budoucnu vylepšena.

Z důvodu změn způsobených rozdělením exonů a malých rozdílů konsensuální a referenční sekvence byla pro další výpočty zvolena pouze datová sada získaná z API Biomart. Rozdíly v signifikantnostech motivů upozornily na důležitost oblastí předělu exonů. Při budoucím rozšiřování projektu bude brán větší důraz na volbu transkriptů namísto dosud využívaných genů. Dále by bylo vhodné vylepšit zpracování anotačního souboru některým z existujících pokročilejších převodníků [Dai], které umožní použít soubory sekvenace u projektů s větším množstvím rozdílů oproti referenčnímu souboru.

Kapitola 6

Nasazení Docker kontejnerů

Instalace velkého množství aplikací je náročná z různých úhlů pohledu. Hlavním důvodem je rozmanitost používaných prostředí, která přináší požadavky na ošetření množství rozdílných verzí závislostí. Programy hledající motivy jsou často distribuovány bez využití správce balíčků a ve většině případů chybí kompletní seznam požadovaných závislostí.

Dalším problémem jsou vysoké nároky na ověření správné funkce, kvality a bezpečnosti běhu programů mimo jiné z důvodu implementace za použití velkého množství programovacích jazyků. Běžný uživatel tedy nemůže snadně ověřit použitý program a riskuje tak neočekávané změny v hostujícím operačním systému.

V případě využití sdílených výpočetních prostředí je uživatel také často omezen úzkou oblastí instalovaných programů. V tomto případě je splnění všech potřebných závislostí ještě mnohem náročnější než v případě instalace na osobním počítači.

Zmíněné problémy řeší právě použití kontejnerů, které umožňují reprodukovat stejné izolované prostředí sestavené podle souboru Dockerfile. Kontejner a obsažené nástroje lze stáhnout z některého hostujícího serveru nebo v případě zveřejněného souboru Dockerfile a potřebných zdrojových kódů sestavit kontejner jedním příkazem.

6.0.1 Srovnání s nástrojem Virtualbox

Hlavní výhodou použití kontejnerů naproti virtualizaci operačního systému je větší podpora u poskytovatelů hostujících náročné výpočty. Například RCI cluster [iP] i MetaCentrum [wc] umožňují běh kontejnerů vytvořených na základě konfiguračních souborů Docker. Další výhodou je menší spotřeba zdrojů, která umožňuje běh většího množství kontejnerů paralelně. Kontejnery oproti virtuálním obrazům operačních systémů mohou být vytvořeny a spuštěny mnohonásobně rychleji. Kontejnery také umožňují snadnou modifikaci prostředí. Změny jako je například výměna operačního systému mohou být

uskutečněny úpravou jediné řádky konfiguračního souboru. Naopak nevýhodou použití kontejnerů je menší izolace od hostujícího operačního systému způsobující mírné snížení bezpečnosti.

6.0.2 Výhody nasazení kontejnerů při hromadných výpočtech

Bezpečnější běh neověřeného softwaru

Zdrojové kódy a binární soubory pocházející od velkého množství autorů mohou skrývat nečekané bezpečnostní hrozby. Problém představují převážně programy, které byly vytvořeny za účelem jednoho použití a nejsou aktivně vyvíjeny. Autoři v těchto případech nekladou příliš velký důraz na bezpečnost. Některé programy jsou distribuovány ve formě spustitelných souborů a zdrojový kód není k dispozici. V jiných případech je zdrojový kód špatně čitelný a nelze snadno ověřit vykonávaná činnost. Zdrojové kódy nebyly zkoumány z pohledu bezpečnosti, přesto je možné narazit na problematické části implementace. Ukázkou je například použití funkce eval na vstupní argument zadávaný uživatelem (MDS2). Dalším exemplářem je nastavení oprávnění adresářů (775 RNAmotifs). Nepříjemné důsledky může vyvolat také neočekávané smazání uživatelem zvoleného adresáře. Přestože se na kontejner nelze z pohledu bezpečnosti plně spoléhat, lze s jeho využitím zamezit většině problémů, způsobených často implementační chybou.

Vyšší reprodukovatelnost

U programů využívaných jen velmi malým množstvím uživatelů je často velkým problémem nízká přenositelnost aplikací způsobená mimo jiné nedostatečným testováním. Běh některých programů nebyl úspěšný ani s použitím přibalených testovacích dat. Méně závažným příkladem je dosažení výsledků nižší kvality než je prezentováno v návodu instalace. Velmi problematické jsou skryté závislosti, které se projevují až v případě použití určité funkce. Výskyt chyby bez řádného odchycení výjimky a bez doprovodného varování je velmi špatně odhalitelný a může vést k mylnému vyhodnocení na základě takto dosažených výsledků.

Použití kontejnerů výrazně zredukovalo množství pozorovaných chyb při přesunu nástrojů na nový systém. Také již sestavené kontejnery bylo možné připravit k použití za zlomek času, oproti nástrojům nevyužívajícím správce balíčků.

Oproštění od náročné instalace

Již sestavené kontejnery lze využívat nezávisle na různých platformách. Tohoto chovaní je velmi těžké docílit i při použití správce balíčků. V případě

vlastnictví souboru Dockerfile lze kontejnery také snadno konfigurovat. Sestavený kontejner neohrožují ani aktualizace balíčku v balíčkovacím systému. Množství bioinformatických programů má příliš volně definované verze knihoven. To se negativně projevuje rozbitím takto instalovaných aplikací. V době psaní této práce došlo k ukončení podpory jazyka Python ve verzi 2 a mnoho nových knihoven již není kompatibilních se staršími balíčky.

Izolace prostredi

Při běhu kontejneru lze využít izolovaného systému souborů. Izolace má značnou výhodu při paralelním spouštění, kdy je zabezpečeno bezpečné oddělení adresářů dočasných souborů. Programy často zapisují dočasné soubory do složky obsahující zdrojový kód programu. Někdy také dochází k zápisu do pracovního adresáře. Obě varianty bez použití náhodného podadresáře vedou ke sdílení stejné složky pro paralelní běh. Toto chování vede v lepším případě k pádu aplikace.

Nevýhodou může být zpomalení přístupu k souborům hostujícího operačního systému. U operačního systému MacOS bylo pozorováno zdvojnásobení doby přenosu velkých souborů oproti stejnému řešení využívajícího Virtualbox.

Některé programy vyžadují úpravy z důvodu pevně stanovené výstupní složky. Takovou složku někdy nelze snadno připojit do souborového systému hostujícího OS z důvodu snahy programu o odstranění složky před zápisem výsledků. Vybízí se zde řešení kopírovat výsledky před ukončením kontejneru, které selhává při nasazení na platformě využívající izolovaný souborový systém bez oprávnění k zápisu.

6.0.3 Volba lokální instalace namísto online rozhraní

Reakcí na velmi náročnou instalaci některých nástrojů je vytvoření online prostředí a zpřístupnění webserveru široké komunitě. Toto řešení je vhodné pro základní otestování programu a řešení problémů menší náročnosti. Online prostředí obvykle obsahují limity na čas běhu a velikost vstupní sady dat. Pozor je potřeba dát také na různá omezení prohledávaného prostoru a nemožnost plné kontroly vstupních parametrů. Nástroj implementovaný v této práci se zaměřuje především na běh ve výpočetním clusteru a online prostředí proto nebylo realizováno.

6.0.4 Použití již sestavených kontejneru

Ke zpracování dat bylo využito také sestavených kontejneru. Několik známých bioinformatickych nástrojů vydává oficiální již sestavený kontejner. Dále lze využít kontejnerů dodávaných komunitou Biocontainers. Komunita využívá

kombinaci manuálně vytvořených souborů Dockerfile a kontejnery vytvořené podle receptů kanálu BioConda ve správci balíčků Conda [dVLGAA+17]. Můžeme tak najít kontejner pro libovolný nástroj obsažený v tomto kanálu.

Namísto instalace správcem balíčků tak byly plně využívány kontejnery a to převážně z důvodu sjednocení procesu získávání nástrojů. Množství automaticky tvořených kontejnerů bohužel neřeší důležitý problém s instalací méně udržovaných programů. Pokud byl nalezen fungující již sestavený kontejner tak došlo k použití tohoto kontejneru přímo nebo jeho rozšířením D.

6.0.5 Sestavení kontejneru

Pro účely hledání motivů bylo sestaveno celkem 16 hlavních základních kontejnerů. Nad těmito kontejnery byly dále prováděny další modifikace pro potřeby výpočtů specificky zaměřených na zkoumanou datovou sadu.

Listing 6.1: Ukázka modifikace sestaveného kontejneru

```
# Kontejner rozšíří již existující kontejner obsahující nástroj MCAT
FROM plachta11b/mcat:0.1

# definice proměnné cestou ke skriptu
ENV PIPELINE /packages/MCAT/orange_pipeline_refine.py

# Odkomentování řádků týkajících se programu DECOD
RUN sed -i '/^#.* runDECOD/ s/^#//' $PIPELINE
RUN sed -i '/^#.* parseDECOD/ s/^#//' $PIPELINE
```

Listing 6.2: Sestavení modifikovaného kontejneru

```
# Docker sestaví kontejner podle souboru Dockerfile
docker build -t modified-mcat:0.1.1 - < Dockerfile
# Zobrazení nápovědy modifikovaného nástroje
docker run modified-mcat:0.1.1 orange_pipeline_refine.py -h
```

Ukázka 6.1 prezentuje možnost rychlé úpravy zdrojového kódu. V sadě nástrojů MCAT je zakomentován program DECOD a jeho spuštění není možné bez vymazání znaku komentáře. Pro tento účel stačí kód jednoduše modifikovat nástrojem **sed** s parametrem obsahujícím vzor pro nahrazení textu. Sestavení kontejneru z uvedeného souboru Dockerfile tak probíhá pouze s použitím platformy Docker a uživatel tak nepotřebuje žádný další software pro časté drobné úpravy nástrojů.

Jako základní kontejner byl namísto kontejneru *biocontainers/biocontainers* využívaného komunitou Biocontainers použit *ubuntu* různých verzí z důvodu mnohonásobně menší velikosti. Teoreticky dojde ke stažení základního kontejneru pouze jednou ale v případě ladění a potřeby výměny sady nástrojů bylo použito větší množství verzí OS. Důsledkem by bylo jen malé využití sdílení základního kontejneru.

Velikost kontejnerů může být problematická mimo delší čas stahování také při použití v platformě Singularity. Nevýhodou velkých kontejnerů je delší čas sestavení, který rychle převáží výhody základního vybavení většího kontejneru. Kontejner *biocontainers/biocontainers* například obsahuje i kompilátor a související nástroje. Chybějící malý kontejner jde proti doporučením pro sestavování kontejnerů s využitím běžně používaného vícefázového sestavení [VAS19].

Kapitola 7

Výpočetní cluster RCI

Hledání motivů je závislé na velkém množství parametrů. Společně s velkým množstvím testovacích dat došlo k navýšení času potřebného na zpracování celého datasetu. Hledání motivů z tohoto důvodu probíhalo za použití ČVUT RCI clusteru. Přístup do výpočetního clusteru je umožněn studentům ČVUT po odeslání žádosti vyplněním formuláře na webu clusteru.[Vecb]

RCI cluster umožňuje spouštět všechny potřebné kontejnery s využitím platformy Singularity. Tato platforma nevyužívá ke správě kontejnerů klienta Docker přesto ale umožňuje automaticky importovat kontejnery vytvořené ze souboru Dockerfile. Všechny Docker kontejnery bylo možné sestavit ale toto řešení má oproti běžnému Docker serveru množství omezení. Uživatel v kontejneru nemůže být root a při sestavování kontejneru dochází k modifikaci některých adresářů. Například u staršího Ubuntu16.04 dochází k přepsání /dev/random na prázdný soubor. Dalším problémem na který je potřeba dát pozor je změna pracovního adresáře.

Všechny kontejnery byly spouštěny s argumentem pro izolaci prostředí. Toto opatření zabraňuje zápisům na neočekávaná místa převážně v jinak připojeném domovském adresáři.

7.0.1 Omezení kontejnerů platformy Singularity

Pro přenositelnost na platformu Singularity je potřeba zajistit dodatečné podmínky pro úspěšný běh programu. Singularity využívá jiný bezpečnostní model, kde uživatel v kontejneru není administrátorem, tak jako je tomu u platformy Docker. Toto opatření má za následek nepřístupnost adresářů s omezenými oprávněními ke čtení pouze administrátorským účtem jako je například adresář textit/root. Změny prostředí způsobují občasné problémy. Na omezení čtení narazil například oficiální kontejner programu ssHMM, který má zdrojové kódy vlastněné právě administrátorským účtem bez oprávnění čtení dalšími uživateli.

Větším problémem při použití Singularity kontejneru na clusteru RCI

je nemožnost zápisu do obsažených adresářů. Docker umožňuje s každým spuštěním vytvoření nového kontejneru a následné smazání po ukončení procesu. Zápis dočasných souborů tak nemůže ovlivnit další běh, protože nový kontejner vlastní opět čistý souborový systém. Konfigurace použitá na výpočetním clusteru RCI ale zápis dočasných souborů nedovoluje, což vede často k pádu aplikace a nutnosti úprav.

Problematické je v kontejneru převážně použití proměnné prostředí *$HOME*, která se může na této platformě měnit podle uživatele spouštějícího kontejner. U některých programů dochází k zápisu na velké množství míst, což je nepříjemné hlavně z důvodu následných mnohočetných úprav zdrojového kódu. Problém s zápisem, který nebylo možné ovlivnit vstupními parametry byl vyřešen přepsáním zdrojového kódu několika aplikací, tak aby braly v potaz proměnou prostředí *$TMPDIR*.

Kontejnery mohou byt také sestaveny jako zapisovatelné (–writeable). Tato volba je ale nežádoucí při použití na výpočetním clusteru z důvodu nutných administrátorských oprávnění.

7.0.2 Analýza výkonu aplikace

Doba běhu programu byla měřena nástrojem perf. Pro případ, že zvolená platforma neumožňuje běh tohoto programu, je ve skriptech připraveno také měření doby běhu použitím vestavěné funkce použitého jazyku time. Pro porovnání tak bylo pro každý FASTA soubor spuštěno hledání motivů a zároveň byla sledována doba běhu testovaného kontejneru.

Poskytovaný operační systém využívá linuxové jádro verze 3.10 s podporou zvoleného profilovacího nástroje [GMdB]. Program je běžně využíván pro měření uplynulého času ale taky k profilování, které je možné pomocnými nástroji převést do grafické podoby. Pro kvalitní přehled o využívaných funkcích je zapotřebí překompilovat všechny programy s informacemi pro ladění. Tato oblast nebyla rozvíjena z důvodu velkého množství použitých nástrojů s podstatnou částí distribuovanou binárními soubory.

Projekt využívá vlastní binární soubor programu perf se staticky linkovanými knihovnami. Záznam doby běhu probíhal zvlášť pro každý kontejner. Běh na výpočetním uzlu byl spouštěn se zařazením do skupiny CPUFAST, kde nehrozí chybný výpočet času z důvodu uspání kontejneru.[Veca]

Sledování délky běhu samotných nástrojů bez příspěvku startování frameworku mělo sloužit k porovnání doby hledání motivů. Tento údaj ale neměl z důvodu finálního návrhu projektu a různorodosti nalezených výsledků příliš velkou vypovídající hodnotu. Údaj byl nakonec použit pouze pro ladění parametrů u programů s velkým rozdílem v času běhu, tak aby byl běh všech paralelně hledajících programů přibližně stejný.

Kapitola 8

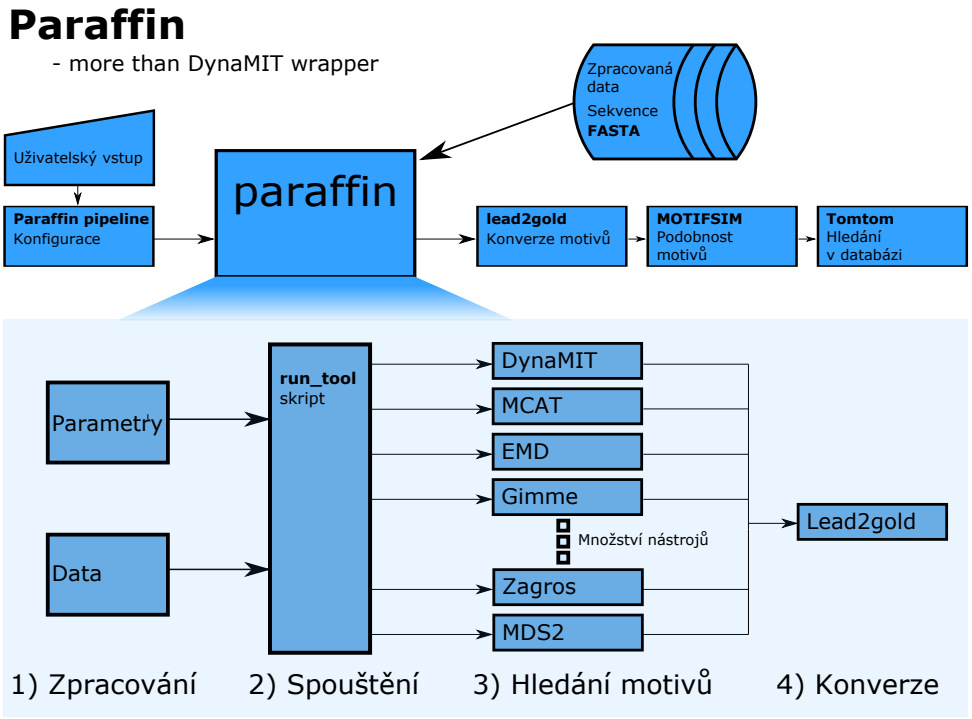
Vytvoření nástroje pro obsluhu programů hledajících motivy

Testování velkého množství nástrojů vedlo k nutnosti jejich integrace do jednotného rámce zabraňujícího nárůstu komplexity projektu. Vytvořen byl proto nástroj paraffin, starající se o přípravu dat, nastavení parametrů a následné spouštění množství kontejnerů obsahujících převážně programy hledající motivy. Pro každý kontejner je připraveno několik skriptů řešících obvyklé kroky specifické pro obsažený program. Vyřešeno je zde například učení modelů nebo filtrace sekvencí nevyhovujících pro použití u některých programů.

Nástroj byl implementován v jazyce Bash s podporou verze 3 a vyšší. Jazyk Bash byl zvolen pro svou širokou podporu mezi operačními systémy (OS). Pro základní zpracování dat využívá běžně dostupných GNU a BSD nástrojů. Paraffin je určen převážně k vykonání základních úkonů a složitější výpočty jsou prováděny využitím příslušné služby implementované v Docker kontejneru. Cílem tohoto návrhu je redukce množství instalovaných balíčků a jejich závislostí na minimum.

V prvním kroku nástroj zpracuje uživatelem zadané parametry jako je minimální a maximální délka motivu, zvolené nástroje, maximum motivů s možností přidání dalších nastavení specifických pro zvolený program. Zpracované parametry jsou poté převedeny a využity při spouštění jednotlivých programů. Uživatel tak nemusí, přestože má možnost, zadávat specifické parametry zvlášť pro každý zvolený program. Soustava skriptů a souvisejících kontejnerů tak umožňuje pouhou záměnou názvu programu spustit stejné hledání vykonávané programem nově zvoleným.

Návrh využívající izolace kontejnerů umožňuje spuštění a běh množství paralelně běžících programů. Každý kontejner je v příslušném skriptu veden pod svým identifikátorem, který slouží k výběru kontejneru z repozitáře. Repozitář uchovává odkazy na kontejnery projektu a umožňuje jednoduchou výměnu verze nebo aktualizaci zdroje. Tento přístup vyžaduje aby se skripty projektu nenacházely uvnitř kontejneru. Další podmínkou je stejné



Obrázek 8.1: Framework pro hledání motivů paraffin

pojmenování binárních souborů uvedených v proměnné prostředí $PATH. Tato možnost byla v průběhu hledání motivů několikrát využita pro udržení projektu v aktuálním stavu. Vyřešen je tak problém s rychlým zastaráváním projektů, které nejsou často po vydání již nadále aktualizovány.

Po vzoru frameworku GimmeMotifs umí paraffin spouštět programy hledající motivy ve velmi zjednodušené formě. Uživatel je oproštěn od nastavování veškerých cest, ať už se jedná o cesty ke zdroji sekvenčních dat nebo o cesty ke spustitelným souborům programů. Nástroj řeší také hledání motivů různých délek, kdy dochází k automatickému opakovanému spouštění programů pro daný rozsah.

Na obrázku 8.1 můžeme vidět celý proces hledání motivů v připravených datech.

Výsledkem je nástroj, který kombinuje výhody uživatelské přívětivosti podobně jako u GimmeMotifs a možnost vyhledávání vzorů typických pro RNA integrací sady nástrojů DynaMIT. Nástroj umožňuje paralelní spuštění obsažených programů a oddělením kroku výpočtů a zpracování výsledků je zabezpečeno, že pád nebo zamrznutí programu hledajícího motivy neovlivní pozdější zpracování ostatních výsledků.

Kapitola 9

Zpracování nalezených motivů

Volba většího množství nástrojů pro hledání motivů působí obtíže při následném zpracování velkého množství výstupních dat. Všechny popsané frameworky hledající motivy obsahují vlastní řešení tohoto problému. Použití některého z frameworku pro pouhé porovnání motivů bylo zavrženo z důvodu složitějšího převodu dat do formátu daného zvoleným frameworkem. Další nevýhodou je nutnost implementace obálky pro část zvoleného frameworku, která pravděpodobně nebude znovupoužitelná při jeho aktualizaci. Tyto důvody vedly k volbě jednoho z nástrojů zaměřených primárně na zpracování výsledných motivů.

9.0.1 Nástroj pro detekci podobnosti motivů MOTIFSIM

Volba tohoto nástroje byla podložena úspěšným použitím v projektu MODSIDE [TH18] využívajícím velmi podobného návrhu zamýšlenému zpracování dat. Nástroj byl také v nedávné době aktualizován a rozšiřován o nové funkce.

Přestože nástroj deklaruje podporu množství formátů motivů, nelze se na zpracování spolehnout. V případě použití formátu MEME lze narazit na tiché přeskočení některých motivů. Nástroj umožňuje přijímat jen malou podmnožinu motivů formátu MEME, přestože motivy dodržují specifikaci. Další nepříjemností je nerozpoznání jména při načítaní motivů v maticovém formátu. Úspěšně rozpoznány byly motivy TRANSFAC-like, které jsou modifikovány pravidlem vyžadujícím prázdnou řádku za každým motivem.

Zprovoznění nástroje pro použití ve skriptech bylo velmi obtížné. Nástroj očekává uživatelem bezchybně zadané parametry skrze standardní vstup. V případě chyby dochází k zacyklení. Program také trpí množstvím dalších chyb jako je například chybné zpracování cest, pevně stanovené cesty k závislostem, výpis části kódu místo dat, atd. Z těchto důvodů došlo ve zdrojovém kódu k množství změn a zároveň byla vytvořena obálka kontrolující vstupní parametry. Pravděpodobněji lepší alternativou k MOTIFSIM by bylo využití projektu STAMP, který není aktivně vyvíjen ale v repositáři není nahlášen žádný výskyt nevyřešených chyb [MB].

MOTIFSIM umožňuje nastavit čtyři typy výstupů a to text, PDF, HTML nebo kombinace všech uvedených. Při testování programů byl zvolen pouze textový výstup který byl automaticky převeden do formátu ze kterého lze snadno získat název generujícího programu, název motivu, skóre podobnosti a další informace o datové sadě. Další nevýhodou programu je nemožnost nastavení parametrů při hledání podobných motivů mimo samotnou podobnost. Při párovém porovnávání se porovnávají také reverzně komplementární motivy které v kombinaci s možností posunu motivu nemají příliš velkou vypovídací hodnotu.

9.0.2 Nástroj pro porovnání motivů Tomtom

Další fází po vyhledání signifikantního motivu je jeho zařazení. Při hledání motivů můžeme narazit na již známé motivy zapsané v některé z databází. Veřejně přístupné databáze běžně nabízí ke stažení množství popsaných matic sekvenčních motivů. Po převedení matic do formátu MEME lze využít nastroj porovnávající motivy Tomtom. Tento nástroj umožňuje rychle prohledat vložené motivy oproti databázi popsaných motivů a výsledky dodat mimo jiné také v uživatelsky přívětivém formátu html. Tento nástroj je vhodný při hledání v databázi, která není součástí projektu paraffin.

Při zpracování zkoumané datové sady byl využíván soubor obsahující kombinaci databází projektu AURA [DRL+14] dále byly přidány motivy ze souboru Ray2013 Homo Sapiens obsaženého v balíčku MEME. Nalezené motivy tak byly porovnány s motivy kombinace databází. Pro zjednodušení procesu byl využíván pro porovnání motivů s databází také program MOTIFSIM, který porovnání provádí jako součást analýzy motivů. Zdrojový kód byl upraven tak, aby bylo možné přidat vlastní databázi.

9.0.3 Významnosti výsledných motivů

Při shlukování výsledků není využíváno statistik jednotlivých programů, ale využívá se skóre počítané nástrojem až po shlukování. Využívá se zde skutečnosti, že jednotlivé programy mohou hodnotit motivy různými technikami a shoda různých metod může signalizovat kvalitnější motiv.

Shlukovaní a až následný výpočet p-value provádí pouze frameworky MCAT a GimmeMotifs. GimmeMotifs ale obsahuje chybu neumožňující spouštět framework na clusteru a MCAT nebyl zařazen do konečného zpracování z důvodu nestability programu CMF. Nestabilita zapříčinila malý průniku zbylých obsažených nástrojů s nástroji samostatně spouštěnými.

Při finálním zpracování clusterů motivů vytvořených nástrojem MOTIFSIM byly motivy posuzovány podle statistik nástrojů ze kterých pochází.

9.0.4 Datové sady

Pro 6 biologických experimentu byly vytvořeny různě velké datové sady cílových oblasti 3UTR, 5UTR. U oblasti CDS byla vytvořena pouze jediná datová sada. Datové sady cílových oblastí byly na clusteru zpracovány a výsledky ve formátu MEME staženy pro další zpracování.

9.0.5 Nalezené motivy

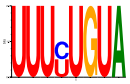
Analýza motivů nástrojem ukázala na časté zastoupení motivů programu GraphProt jako motivy best-matches. Tato statistika sleduje podobnost zvlášť každého sekvenčního motivu se všemi nejpodobnějšími motivy ostatních datasetů. Z tohoto výsledku lze usoudit, že programy hledající pouze sekvenční motivy se pravděpodobně zaměřují na místo s určitou strukturou ale nedokáží jej s přesností lokalizovat. Tyto sekvenční motivy jsou poté v některé části velmi podobné krátkému motivu programu GraphProt, který tímto získá nejvyšší skóre.

Dále je uveden nejmenší testovaný dataset tvořen sekvencemi oblasti 3' UTR. Vybírány byly motivy s vysokou shodou programů hledajících sekvenční motivy.

(a) : GraphProt

(b) : HOMER $ln(\text{Pval}) = -22{,}34$

Obrázek 9.1: Experiment 02 UTR3

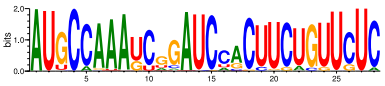


(a) : GraphProt

(b) : MEME $ln(\text{Eval}) = -3{,}14$

Obrázek 9.2: Experiment 03 UTR3

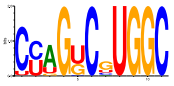
(a) : GraphProt



(b) : BaMM $ln(\text{Pval}) = -138{,}43$

Obrázek 9.3: Experiment 04 UTR3

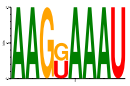
(a) : GraphProt



(b) : BaMM $ln(\text{Pval}) = -25{,}37$

Obrázek 9.4: Experiment 05 UTR3

(a) : GraphProt



(b) : DREME $ln(\text{Eval}) = -7{,}14$

Obrázek 9.5: Experiment 06 UTR3

(a) : GraphProt

(b) : DREME $ln(\text{Eval}) = -5{,}22$

Obrázek 9.6: Experiment 07 UTR3

#	IUPAC	ln(Sig.)	Shoda
02	GAGTGG	P −22,34	BaMM, Discrover, HOMER, DREME
03	ACAGAA	E −3,14	MEME
04	ATGCCAAA...	P −138,43	BaMM
05	CYAGKCGTGGC	P −25,37	BaMM
06	AAGKAAAT	E −7,14	DREME, HOMER, BaMM, Discrover
07	GGGCCS	E −5,22	DREME, Discrover, BaMM

Tabulka 9.1: Ukázka sekvenčních motivů. Dataset 8 genů oblast UTR3. Symbol E značí E-value a P značí Pvalue. 7 experimentů rozdílných buněčných linií.

Kapitola 10

Závěr

Výstupem této práce je kompletní řešení umožňující hledat motivy v datové sadě RNA-seq s provedenou analýzou exprese. Pro přípravu sekvencí na základě dodané analýzy jsou v rámci práce implementovány skripty umožňující stahovat sekvence z veřejně přístupného API Biomart. Alternativní možností je generování sekvencí obsahující rozdíly obsažené v sekvenačních datech v porovnání s referenčním genomem. Ve vygenerovaných nebo stažených sekvencích lze vyhledávat motivy využitím implementovaného nástroje umožňujícího paralelně spouštět velké množství programů hledajících motivy. Vyhledané motivy lze zpracovat upraveným nástrojem MOTIFSIM do uživatelsky přívětivé podoby. Celý proces je řešen využitím kontejnerů pro zbavení se množství závislostí obsažených programů. Implementované skripty starající se o spouštění kontejnerů vyžadují pouze běžné linuxové nástroje a aplikaci Docker. Celé řešení je možné využívat také na výpočetním clusteru s použitím kontejnerů Singularity.

Před samotným stahováním sekvencí dochází ke zpracování informací z dodané analýzy. Stahování sekvencí podle zpracovaných informací využívá vytvořeného kontejneru obsahujícího oficiální aplikaci komunikující s veřejným API Biomart. Implementovány jsou také skripty řešící některé problémy při stahování dat z jmenovaného API. Generování sekvencí ze sekvenovaných dat je realizováno skripty, které se opírající o kontejnery kvalitních sad nástrojů bcftools, bedops, bedtools, seqtk a nástroje crossmap. Skripty řeší celý postup přípravy dat ze souboru mapovaných sekvencí. Výstupem jsou sekvence v běžně podporovaném formátu FASTA.

Za účelem hledání motivů byl implementován nástroj paraffin vycházející z informací provedené rešerše. Implementace využívá integraci množství testovaných nástrojů hledajících sekvenční motivy. Zahrnuto bylo také několik nástrojů využívajících strukturní informace připravených sekvencí. Pozornost byla zaměřena převážně na nástroje umožňující discriminační hledání motivů. Tímto způsobem lze hledat motivy obohacené pouze v jedné z prohledávaných datových sad.

Nástroj parrafin využívá kontejnerizaci všech použitých nástrojů. Tento

návrh umožňuje jejich bezpečné paralelní spuštění. Nástroj se stará také o přípravu všech potřebných parametrů a souborů pro spuštění všech dostupných nástrojů jediným příkazem. Vyřešena je také následná konverze všech nalezených motivů do nejpoužívanějších formátů MEME, TRANSFAC a konsensuální sekvence implementací nástroje pro převod formátů motivů lead2gold. Takto sjednocené motivy mohou být dále převedeny do grafického formátu Weblogo nebo dále zpracovány hledáním motivů se shodou u více nástrojů. Hledání podobnosti ve výsledcích hledání řeší pro tento projekt opravený program MOTIFSIM integrovaný do nástroje paraffin.

Celý postup využívající skriptů pro přípravu dat nástroje paraffin a lead2gold byl použit pro hledání motivů v dodané datové sadě. Zvolený postup úspěšně vyhledal signifikantně obohacené motivy související se zvýšenou expresí v několika buněčných liniích.

Budoucí vývoj se bude zabývat především opravou nynějších nedostatků. Jako slabý článek uceleného postupu se ukázalo použití programu MOTIFSIM, který i přes provedená vylepšení stále nedosahuje očekávané úrovně. Výstup ve formátu HTML umožňuje pouze statické zobrazení výsledků, které je přehledné, ale nenabízí možnost filtrace nebo alternativního seřazení výsledků.

Programy instalované v samostatném kontejneru podávaly stejné nebo lepší výsledky než programy integrované v kontejneru frameworku. Na základě tohoto pozorování budou postupně z implementovaného nástroje paraffin použité kontejnery jiných frameworků vyřazeny.

Projekt výrazně zjednodušuje hledání motivů v RNA datových sadách odstraněním kroku ladění a kompilace použitých programů. Program paraffin ale není dostupný uživateli neovládajícímu prostředí příkazové řádky. Návrh počítá s rozšířením o kontejner obsahující webový server sloužící jako interaktivní webové GUI.

Postup generující sekvence je natolik komplexní, že dojde k oddělení do samostatného projektu. Zachováno bude pouze předávání zpracovaných dat. Pro naplnění plného potenciálu této části projektu potřeba v kroku stahování anotačního souboru převést stažený anotační soubor do formátu BED12. Vyřeší se tím problém s rozdělenými exony v datech generovaných touto metodou.

Množství nástrojů nebylo ani zdaleka připraveno pro nasazení na výpočetní cluster. Nástroje které potřebovaly pro svůj běh interaktivní vstup nebo grafické prostředí byly obaleny do skriptů řešících možnost neinteraktivního běhu. Množství upravených a opravených projektů bude uveřejněno.

Příloha A

Literatura

[AA] Mohammed Alshalalfa and Reda Alhajj, *Motif location prediction by divide and conquer*, Communications in Computer and Information Science, Springer Berlin Heidelberg, pp. 102–113.

[AHS+18] Xian Adiconis, Adam L. Haber, Sean K. Simmons, Ami Levy Moonshine, Zhe Ji, Michele A. Busby, Xi Shi, Justin Jacques, Madeline A. Lancaster, Jen Q. Pan, Aviv Regev, and Joshua Z. Levin, *Comprehensive comparative analysis of 5′-end RNA-sequencing methods*, Nature Methods **15** (2018), no. 7, 505–511.

[Bai11] Timothy L. Bailey, *DREME: motif discovery in transcription factor ChIP-seq data*, Bioinformatics **27** (2011), no. 12, 1653–1659.

[bcf] *bcftools utilities for variant calling and manipulating vcfs and bcfs*, `http://www.htslib.org/doc/bcftools.html`, [Online; accessed 6-June-2020].

[BE94] T. L. Bailey and C. Elkan, *Fitting a mixture model by expectation maximization to discover motifs in biopolymers*, Proc Int Conf Intell Syst Mol Biol **2** (1994), 28–36.

[beda] *Bed file format - definition and supported options*, `https://m.ensembl.org/info/website/upload/bed.html`.

[bedb] *Frequently Asked Questions: Data File Formats*, `https://genome.ucsc.edu/FAQ/FAQformat.html`, [Online; accessed 11-Aug-2020].

[BSPSU14a] Emad Bahrami-Samani, Luiz O.F. Penalva, Andrew D. Smith, and Philip J. Uren, *Leveraging cross-link modification events in CLIP-seq for motif discovery*, Nucleic Acids Research **43** (2014), no. 1, 95–103.

[BSPSU14b] _____, *Leveraging cross-link modification events in CLIP-seq for motif discovery – supplementary*, Nucleic Acids Research **43** (2014), no. 1, 95–103.

[BSS] Karel Brinda, Greg Slodkowicz, and Daniel Standage, *Remapping genomic coordinates to account for indels*, `https://bioinformatics.stackexchange.com/a/286`, [Online; accessed 25-July-2020].

[BvH18] Niklas Bruse and Simon J. van Heeringen, *GimmeMotifs: an analysis framework for transcription factor motif analysis.*

[CB85] Athel Cornish-Bowden, *Nomenclature for incompletely specified bases in nucleic acid sequences: rcommendations 1984*, Nucleic Acids Research **13** (1985), no. 9, 3021–3030.

[cha] *Chain format*, `https://genome.ucsc.edu/goldenPath/help/chain.html`, [Online; accessed 25-July-2020].

[CHCB04] Gavin E. Crooks1, Gary Hon1, John-Marc Chandonia2, and Steven E. Brenner, *WebLogo: A sequence logo generator*, Genome Research **14** (2004), no. 6, 1188–1190.

[Dai] Jacques Dainat, *GFF to BED conversion - review of the main conversion tools*, `https://github.com/NBISweden/GAAS/blob/master/annotation/knowledge/gff_to_bed.md`, [Online; accessed 11-Aug-2020].

[Das18] Erik Dassi, *DynaMIT: the dynamic motif integration toolkit*, `https://bitbucket.org/erikdassi/dynamit/wiki/Home`, 2018.

[DQ15] Erik Dassi and Alessandro Quattrone, *DynaMIT: the dynamic motif integration toolkit*, Nucleic Acids Research **44** (2015), no. 1, e2–e2.

[DRL+14] Erik Dassi, Angela Re, Sara Leo, Toma Tebaldi, Luigi Pasini, Daniele Peroni, and Alessandro Quattrone, *AURA 2 empowering discovery of post-transcriptional networks.*, Translation **2** (2014), no. 1, e27738.

[dVLGAA+17] Felipe da Veiga Leprevost, Björn A Grüning, Saulo Alves Aflitos, Hannes L Röst, Julian Uszkoreit, Harald Barsnes, Marc Vaudel, Pablo Moreno, Laurent Gatto, Jonas Weber, Mingze Bai, Rafael C Jimenez, Timo Sachsenberg, Julianus Pfeuffer, Roberto Vera Alvarez, Johannes Griss, Alexey I Nesvizhskii, and Yasset Perez-Riverol, *BioContainers: an open-source and community-driven framework for software standardization*, Bioinformatics **33** (2017), no. 16, 2580–2582.

[Ens] Ensembl, *Biomart perl api*, `http://Apr2020.archive.ensembl.org/info/data/biomart/biomart_perl_api.html`, [Online; accessed 11-Aug-2020].

[Fej08] Anthony P. Fejes, *FindPeaks 3.1.9.2 Manual*, `https://www.bcgsc.ca/platform/bioinfo/software/findpeaks/releases/3.1.9.2/findpeaks3-1-9-2-tar.gz`, September 2008.

[Gena] Genome Reference Consortium, *Human genome overview*, `https://www.ncbi.nlm.nih.gov/grc/human`, [Online; accessed 29-June-2020].

[Genb] ———, *Introduction to patches*, `https://www.ncbi.nlm.nih.gov/grc/help/patches`, [Online; accessed 29-June-2020].

[GMdB] Eric Gouriou, Tipp Moseley, and Willem de Bruijn, *Linux kernel profiling with perf*.

[GR14] Jan Gorodkin and Walter L. Ruzzo (eds.), *RNA sequence, structure, and function: Computational and bioinformatic methods*, Humana Press, 2014.

[GSC18] Tian Gao, Jiang Shu, and Juan Cui, *A systematic approach to RNA-associated motif discovery*, BMC Genomics **19** (2018), no. 1.

[Gui03] Roderic Guigo, *An introduction to position specific scoring matrices*, `http://bioinformatica.upf.edu/T12/MakeProfile.html`, March 2003, [Online; accessed 6-June-2020].

[HETC00] Jason D Hughes, Preston W Estep, Saeed Tavazoie, and George M Church, *Computational identification of cis -regulatory elements associated with groups of functionally related genes in saccharomyces cerevisiae 1 1edited by f. e. cohen*, Journal of Molecular Biology **296** (2000), no. 5, 1205–1214.

[HGS+] H. Hartmann, E. W. Guthohrlein, M. Siebert, S. Luehr, and J. Soding, *XXmotif: sourcecode*, `https://github.com/soedinglab/xxmotif`, [Online; accessed 6-June-2020].

[HGS+12] ———, *P-value-based regulatory motif discovery using positional weight matrices*, Genome Research **23** (2012), no. 1, 181–194.

[HKO+17] David Heller, Ralf Krestel, Uwe Ohler, Martin Vingron, and Annalisa Marsico, *ssHMM: extracting intuitive sequence-structure motifs from high-throughput RNA-binding protein data*, Nucleic Acids Research **45** (2017), no. 19, 11004–11018.

[HMAA19] F. A. Hashim, M. S. Mabrouk, and W. Al-Atabany, *Review of Different Sequence Motif Finding Algorithms*, Avicenna J Med Biotechnol **11** (2019), no. 2, 130–148.

[HPBB06] Michael Hiller, Rainer Pudimat, Anke Busch, and Rolf Backofen, *Using RNA secondary structures to guide sequence motif finding towards single-stranded regions*, Nucleic Acids Research **34** (2006), no. 17, e117–e117.

[HYK06] Jianjun Hu, Yifeng D Yang, and Daisuke Kihara, *EMD: an ensemble algorithm for discovering regulatory motifs in dna sequences*, BMC Bioinformatics **7** (2006), no. 1, 342.

[HZS+11] Peter Huggins, Shan Zhong, Idit Shiff, Rachel Beckerman, Oleg Laptenko, Carol Prives, Marcel H. Schulz, Itamar Simon, and Ziv Bar-Joseph, *DECOD: fast and accurate discriminative DNA motif finding*, Bioinformatics **27** (2011), no. 17, 2361–2367.

[iP] CTU in Prague, *Rci cluster delivery to fel Čvut in prague*, `https://www.mcomputers.cz/en/2019/04/16/dodavka-rci-clusteru-na-fel-cvut/`.

[IY15] Hisaki Ikebata and Ryo Yoshida, *Repulsive parallel MCMC algorithm for discovering diverse motifs from large sequence sets*, Bioinformatics **31** (2015), no. 10, 1561–1568.

[JM18] Benjamin Jean-Marie, *universalmotif*, 2018.

[Joh10] A. D. Johnson, *An extended IUPAC nomenclature code for polymorphic nucleic acids*, Bioinformatics **26** (2010), no. 10, 1386–1389.

[KKH+11] R. J. Kinsella, A. Kahari, S. Haider, J. Zamora, G. Proctor, G. Spudich, J. Almeida-King, D. Staines, P. Derwent, A. Kerhornou, P. Kersey, and P. Flicek, *Ensembl BioMarts: a hub for data retrieval across taxonomic space*, Database **2011** (2011), no. 0, bar030–bar030.

[KRG+18] Anja Kiesel, Christian Roth, Wanwan Ge, Maximilian Wess, Markus Meier, and Johannes Söding, *The BaMM web server for de-novo motif discovery and regulatory sequence analysis*, Nucleic Acids Research **46** (2018), no. W1, W215–W220.

[KSD19] Manojkumar Kumaran, Umadevi Subramanian, and Bharanidharan Devarajan, *Performance assessment of variant calling pipelines using human whole exome sequencing and simulated data*, BMC Bioinformatics **20** (2019), no. 1.

[KTP08] P. V. Kharchenko, M. Y. Tolstorukov, and P. J. Park, *Design and analysis of ChIP-seq experiments for DNA-binding proteins*, Nat. Biotechnol. **26** (2008), no. 12, 1351–1359.

[Lab] Benner Lab, *Practical tips to motif finding with homer*, `http://homer.ucsd.edu/homer/motif/practicalTips.html`.

[LBL01] X. Liu, D. L. Brutlag, and J. S. Liu, *BioProspector: discovering conserved DNA motifs in upstream regulatory regions of co-expressed genes*, Pac Symp Biocomput (2001), 127–138.

[LBL02] X. Shirley Liu, Douglas L. Brutlag, and Jun S. Liu, *An algorithm for finding protein–DNA binding sites with applications to chromatin- immunoprecipitation microarray experiments*, Nature Biotechnology **20** (2002), no. 8, 835–839.

[Li09] L. Li, *GADEM: a genetic algorithm guided formation of spaced dyads coupled with an EM algorithm for motif discovery*, J. Comput. Biol. **16** (2009), no. 2, 317–329.

[LMPT13] Mauro Leoncini, Manuela Montangero, Marco Pellegrini, and Karina Panucia Tillán, *CMF: A combinatorial tool to find composite motifs*, Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2013, pp. 196–208.

[LNZ+18] Yang Li, Pengyu Ni, Shaoqiang Zhang, Guojun Li, and Zhengchang Su, *Ultra-fast and accurate motif finding in large ChIP-seq datasets reveals transcription factor binding patterns.*

[MB] Shaun Mahony and Panayiotis V Benos, *Stamp: a web tool for exploring dna-binding motif similarities*, `https://github.com/seqcode/stamp`, [Online; accessed 1-August-2020].

[MLCB14] Daniel Maticzka, Sita J Lange, Fabrizio Costa, and Rolf Backofen, *GraphProt: modeling binding preferences of RNA-binding proteins*, Genome Biology **15** (2014), no. 1, R17.

[Mm] Max Masnick and mangfu100, *Question: Human dna reference file with no prefix 'chr'.*

[MR14] Jonas Maaskola and Nikolaus Rajewsky, *Binding site discovery from nucleic acid sequences by discriminative learning of hidden markov models*, Nucleic Acids Research **42** (2014), no. 21, 12995–13011.

[NKR+12] Shane Neph, M. Scott Kuehn, Alex P. Reynolds, Eric Haugen, Robert E. Thurman, Audra K. Johnson, Eric Rynes, Matthew T. Maurano, Jeff Vierstra, Sean Thomas, Richard

Sandstrom, Richard Humbert, and John A. Stamatoyannopoulos, *BEDOPS: high-performance genomic feature operations*, Bioinformatics **28** (2012), no. 14, 1919–1920.

[pav] Giuliano pavesi, *How to build a new frequency file for weeder 2.0*, `http://159.149.160.88/modtools/`, [Online; accessed 10-June-2020].

[PCA+16] Allison Piovesan, Maria Caracausi, Francesca Antonaros, Maria Chiara Pelleri, and Lorenza Vitale, *GeneBase 1.1: a tool to summarize data from NCBI gene datasets and its application to an update of human gene statistics*, Database **2016** (2016), baw153.

[PMMP04] G. Pavesi, P. Mereghetti, G. Mauri, and G. Pesole, *Weeder web: discovery of transcription factor binding sites in a set of sequences from co-regulated genes*, Nucleic Acids Research **32** (2004), no. Web Server, W199–W203.

[QKa] Aaron R. Quinlan and Neil Kindlon, *bedtools: a powerful toolset for genome arithmetic.*

[QKb] ———, *Linux kernel profiling with perf.*

[rc] Samtools repository contributors, *Bcftools howto*, `https://samtools.github.io/bcftools/howtos/consensus-sequence.html`, [Online; accessed 6-June-2020].

[SNR+18] Chadi Saad, Laurent Noé, Hugues Richard, Julie Leclerc, Marie-Pierre Buisine, Hélène Touzet, and Martin Figeac, *DiNAMO: highly sensitive DNA motif discovery in high-throughput sequencing data*, BMC Bioinformatics **19** (2018), no. 1.

[src20] samtools repository contributors, *Sequence Alignment/-Map Format Specificationl*, `https://github.com/samtools/hts-specs`, August 2020.

[SYC+11] Jiantao Shi, Wentao Yang, Mingjie Chen, Yanzhi Du, Ji Zhang, and Kankan Wang, *AMD, an automated motif discovery tool using stepwise refinement of gapped consensuses*, PLoS ONE **6** (2011), no. 9, e24576.

[TH18] Ngoc Tam L. Tran and Chun-Hsi Huang, *MODSIDE: a motif discovery pipeline and similarity detector*, BMC Genomics **19** (2018), no. 1.

[TLM+01] G. Thijs, M. Lescot, K. Marchal, S. Rombauts, B. De Moor, P. Rouze, and Y. Moreau, *A higher-order background model*

improves the detection of promoter regulatory elements by gibbs sampling, Bioinformatics **17** (2001), no. 12, 1113–1122.

[TNC+07] W. A. Thompson, L. A. Newberg, S. Conlan, L. A. McCue, and C. E. Lawrence, *The gibbs centroid sampler*, Nucleic Acids Research **35** (2007), no. Web Server, W232–W237.

[Tyn16] Cath Tyner, *The ucsc genome browser coordinate counting systems*, `http://genome.ucsc.edu/blog/the-ucsc-genome-browser-coordinate-counting-systems/`, 2016, [Online; accessed 11-Aug-2020].

[VAS19] TIBOR VASS, *Intro guide to dockerfile best practices*, `https://www.docker.com/blog/intro-guide-to-dockerfile-best-practices/`, July 2019.

[Veca] Vecerka, *Changes in rci cluster scheduler from july 2020*.

[Vecb] ———, *Rci cluster intro*.

[wc] Metacentrum wiki contributors, *Singularity*, `https://wiki.metacentrum.cz/w/index.php?title=Singularity&oldid=9305`.

[web] *WebLogo: A sequence logo generator*, `http://weblogo.threeplusone.com/create.cgi`, [Online; accessed 6-June-2020].

[Wik19] Wikiversity, *Wikijournal of medicine/eukaryotic and prokaryotic gene structure — wikiversity,*, 2019, [Online; accessed 28-January-2019].

[Wik20a] Wikipedia contributors, *Fasta format — Wikipedia, the free encyclopedia*, `https://en.wikipedia.org/w/index.php?title=FASTA_format&oldid=956134292`, 2020, [Online; accessed 20-June-2020].

[Wik20b] Wikipedie, *Párování bází — wikipedie: Otevřená encyklopedie*, 2020, [Online; navštíveno 18. 07. 2020].

[YRG+19] Yanshen Yang, Jeffrey A. Robertson, Zhen Guo, Jake Martinez, Christy Coghlan, and Lenwood S. Heath, *MCAT: Motif combining and association tool*, Journal of Computational Biology **26** (2019), no. 1, 1–15.

[YWR05] Z. Yao, Z. Weinberg, and W. L. Ruzzo, *CMfinder–a covariance model based RNA motif finding algorithm*, Bioinformatics **22** (2005), no. 4, 445–452.

[ZKCB04] L. Zhang, S. Kasif, C. R. Cantor, and N. E. Broude, *GC/AT-content spikes as genomic punctuation marks*, Proceedings of the National Academy of Sciences **101** (2004), no. 48, 16855–16860.

[ZSW+13] Hao Zhao, Zhifu Sun, Jing Wang, Haojie Huang, Jean-Pierre Kocher, and Liguo Wang, *CrossMap: a versatile tool for coordinate conversion between genome assemblies*, Bioinformatics **30** (2013), no. 7, 1006–1007.

Příloha B

Soubory na CD

```
Project files directory
├── thesis.pdf
├── project_source_code
│   ├── bioplachta_containers
│   ├── find_motifs
│   │   ├── tools
│   │   ├── similarity
│   │   └── pipelines
│   ├── generate_fasta
│   │   ├── bam_to_bcf
│   │   ├── bcf_to_cons
│   │   ├── biomart
│   │   ├── cons_to_fasta
│   │   ├── make_filter
│   │   ├── pipelines
│   │   └── PPRSG
│   ├── lead2gold
│   └── motif_description
├── Motifs
└── Docs
```

Příloha C

Některé příkazy použité při datových manipulacích

Příloha D

List použitých kontejnerů

```
# data preparation
# bvaldebenitom (fasta sequence does not mach REF allele)
# https://github.com/samtools/bcftools/issues/888
bcftools_broken biocontainers/bcftools:v1.9-1-deb_cv1
# fixed issue above
bcftools lifebitai/bcftools:1.10.2-51-ga205d5c
# important containers
agat quay.io/biocontainers/agat:0.4.0--pl526r35_0
bedops quay.io/biocontainers/bedops:2.4.39--hc9558a2_0
bedtools biocontainers/bedtools:v2.28.0_cv2
biomart plachta11b/biomart-xml-client:0.3-ensembl
crossmap crukcibioinformatics/crossmap
seqtk biocontainers/seqtk:v1.3-1-deb_cv1
# used but not required to get output (included in bedops)
gffread zavolab/gffread:0.11.7-slim
gffutils quay.io/biocontainers/gffutils:0.10.1--py_0

# motif search
alignace plachta11b/alignace:0.1
bamm soedinglab/bamm-suite:1.0.0
bash bash:4.4.23
bedtools quay.io/biocontainers/bedtools:2.29.2--hc088bd4_0
busybox busybox:1.32-glibc
cmfinder plachta11b/cmfinder:0.1
discrover plachta11b/discrover:0.1
dynamit plachta11b/dynamit:0.1-compute
emd plachta11b/emd:0.1
# extended container from biocontainers
gimme plachta11b/gimme_motifs:0.2-dev
graphprot plachta11b/graphprot:0.1
hellerd_sshmm hellerd/sshmm
homer biowardrobe2/homer:v0.0.2
latest_memesuite memesuite/memesuite:latest
lead2gold plachta11b/lead2gold:0.1
mcat plachta11b/mcat:0.2
mds2 plachta11b/mds2:0.1
memesuite biowardrobe2/memesuite:v0.0.1
```

```
motifsim plachta11b/motifsim:0.1
rck plachta11b/rck:0.1
rnacontext plachta11b/rnacontext:0.1
rpmcmc plachta11b/rpmcmc:0.1
# extended from hellerd/sshmm
sshmm plachta11b/sshmm:0.1
weeder2 quay.io/biocontainers/weeder:2.0--h6bb024c_3
xxmotif quay.io/biocontainers/xxmotif:1.6--h2d50403_2
zagros plachta11b/zagros:0.1
```

Příloha E

Soubory Dockerfile

E.0.1 GimmeMotifs Dockerfile - ukázka rozšíření kontejneru

```
FROM alpine/git:latest AS builder

WORKDIR /

# download version with fixed .sizes issue
RUN git clone https://github.com/vanheeringen-lab/genomepy.git
RUN git -C /genomepy checkout \
       b11f2f21cadf7d37a6b76c225c15a5f4f247f506

# download version with fixed cmds
#git clone --depth 1 git@github.com:VENDOR/REPO.git --branch 1.23.0
    --single-branch
RUN git clone https://github.com/vanheeringen-lab/gimmemotifs.git
RUN git -C /gimmemotifs checkout \
       d8eae732cb4eacb3dec6669386f79e23c5cb095c

FROM quay.io/biocontainers/gimmemotifs:0.14.4--py37h516909a_0

# genomepy install
COPY --from=builder /genomepy /genomepy
RUN python3 -m pip --no-cache-dir install /genomepy

# gimmemotifs install
COPY --from=builder /gimmemotifs /gimmemotifs
RUN opkg-install gcc libpthread
RUN ar -rc /usr/lib/libpthread.a
RUN python3 -m pip --no-cache-dir install /gimmemotifs

# dev
RUN opkg-install git git-http

# fix meme
ENV OMPI_MCA_plm_rsh_agent sh
```

```
# fix improbizer motif conversion
# uncomment if you need motif conversion
# RUN sed -i 's/m = p.search(line)/line=line.strip(); m = p.search(
    line)/' /usr/local/lib/python3.7/site-packages/gimmemotifs/tools/
    improbizer.py

# fix user privileges for non-root environments
RUN echo "gimmemotifs:x:1001:1001::/home/gimmemotifs:/bin/bash" \
>> /etc/passwd \
&& echo "gimmemotifs:!:1001:" >> /etc/group \
&& mkdir -p /home/gimmemotifs \
/home/gimmemotifs/.cache /home/gimmemotifs/.config \
&& chown -R gimmemotifs:gimmemotifs /home/gimmemotifs
USER gimmemotifs

# docker build -t plachta11b/gimme_motifs:dev - < Dockerfile
```

E.0.2 GimmeMotifs Dockerfile - ukázka vytvoření kontejneru

```
FROM ubuntu:20.04 AS builder

RUN apt-get update && apt-get install -y \
make \
g++ \
&& rm -rf /var/lib/apt/lists/*

# wget http://daweb.ism.ac.jp/yoshidalab/motif/rpmcmc-0.2.tar.gz
ADD ./rpmcmc-0.2.tar.gz /source
RUN mkdir /packages && mv /source/rpmcmc-0.2/ /packages/rpmcmc

WORKDIR /packages/rpmcmc/src
RUN make clean
RUN make
RUN make install

FROM ubuntu:20.04

COPY --from=builder /packages/ /packages/
COPY --from=builder /usr/lib/x86_64-linux-gnu/libgomp.so.1 \
       /usr/lib/x86_64-linux-gnu/libgomp.so.1

ENV PATH $PATH:/packages/rpmcmc/bin
ENV LD_LIBRARY_PATH $LD_LIBRARY_PATH:/packages/rpmcmc/lib

# (stack size) Before executing the application, it is necessary to
    increase the stack size as following command:
CMD ulimit -s unlimited && multi_motif_sampler
```

Příloha F

Anotační soubory

Listing F.1: Annotation files exploration

```
#UCSC annotation
cat ensGene.gff3 knownGene.gff3 ncbiRefSeq.gff3 refGene.gff3 \
| awk -F$'\t' '{print $3}' | sort | uniq
# > CDS,exon,transcript

#EBI GENCODE https://www.gencodegenes.org/human/
gencode="gencode.v34.chr_patch_hapl_scaff.annotation.gff3"
cat $gencode | awk -F$'\t' '{print $3}' | sort | uniq
# > CDS,exon,five_prime_UTR,gene,start_codon,stop_codon,
    stop_codon_redefined_as_selenocysteine,three_prime_UTR,transcript
cat $gencode | grep "three_prime_UTR" | head -n 1 | wc -c
# > 457 (WARNING: more content :)
cat $gencode | grep "five_prime_UTR" | wc -l
# > 166128
cat $gencode | grep "three_prime_UTR" | wc -l
# > 167820

# ENSEMBL ftp://ftp.ensembl.org/pub/release-100/gff3/homo_sapiens/
ensembl="Homo_sapiens.GRCh38.100.gff3"
cat $ensembl | awk -F$'\t' '{print $3}' | sort | uniq
# > CDS,C_gene_segment,D_gene_segment,J_gene_segment,V_gene_segment,
    biological_region,chromosome,exon,five_prime_UTR,gene,lnc_RNA,
    mRNA,miRNA,ncRNA,ncRNA_gene,pseudogene,pseudogenic_transcript,
    rRNA,scRNA,scaffold,snRNA,snoRNA,tRNA,three_prime_UTR,
    unconfirmed_transcript,vaultRNA_primary_transcript
cat $ensembl | grep "three_prime_UTR" | head -n 1 | wc -c
# > 77
cat $ensembl | grep "five_prime_UTR" | wc -l
# > 153607
cat $ensembl | grep "three_prime_UTR" | wc -l
# > 155815

# BIOMART
# 5utr,3utr,gene_exon,cdna,coding
```

ZADÁNÍ DIPLOMOVÉ PRÁCE

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Holčák** Jméno: **Jan** Osobní číslo: **439587**

Fakulta/ústav: **Fakulta elektrotechnická**

Zadávající katedra/ústav: **Katedra počítačů**

Studijní program: **Otevřená informatika**

Specializace: **Kybernetická bezpečnost**

II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

Hledání sekvenčních motivů v mRNA selektovaných vazbou na translační iniciační faktory z rodiny eIF4E

Název diplomové práce anglicky:

Search for sequence motifs in mRNAs selected by binding of translation initiation factors from the eIF4E family

Pokyny pro vypracování:

1) Zpracovat dostupné informace o softwarových řešeních používaných pro vyhledávání a analýzu sekvenčních motivů uvnitř nepřekládaných i překládaných oblastí mRNA.
2) Otestovat a porovnat dostupná softwarová řešení na souboru dat získaných metodou RNAseq NGS nebo na jiném modelovém souboru dat.
3) Vybrat nejvhodnější softwarová řešení, případně vyvinout vlastní, a navrhnout ucelený postup pro vyhledávání a analýzu sekvenčních motivů uvnitř rozsáhlých souborů dat získaných metodou RNAseq NGS. Postup by měl umožňovat samostatné vyhledávání a analýzu sekvenčních motivů v 5'; a 3'; nepřekládaných oblastech mRNA.
4) Použít vyvinutý postup pro nalezení sekvenčních motivů v mRNA specificky interagujících s translačními iniciačními faktory z rodiny eIF4E. Soubory relevantních dat byly získány v rámci společného výzkumu s Genomics Core Facility v Evropské molekulárně biologické laboratoři (EMBL, Heidelberg).

Seznam doporučené literatury:

[1] RNA Sequence, Structure, and Function: Computational and Bioinformatic Methods, Editors: Jan Gorodkin and Walter L. Ruzzo, Springer 2014
[2] Mrvová S. et al., Major splice variants and multiple polyadenylation site utilization in mRNAs encoding human translation initiation factors eIF4E1 and eIF4E3 regulate the translational regulators? Mol Genet Genomics. 2018 Feb;293(1):167-186. doi: 10.1007/s00438-017-1375-4.
[3] a mnoho dalších; vyhledání relevantní literatury je součástí zadání DP

Jméno a pracoviště vedoucí(ho) diplomové práce:

RNDr. Martin Pospíšek, Ph.D., katedra počítačů FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) diplomové práce:

Datum zadání diplomové práce: **06.02.2020** Termín odevzdání diplomové práce: _____

Platnost zadání diplomové práce: **30.09.2021**

RNDr. Martin Pospíšek, Ph.D. podpis vedoucí(ho) práce	podpis vedoucí(ho) ústavu/katedry	prof. Mgr. Petr Páta, Ph.D. podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Diplomant bere na vědomí, že je povinen vypracovat diplomovou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací.
Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v diplomové práci.

. _____
Datum převzetí zadání

_____
Podpis studenta

