RAPPORT DE STAGE

Comparaison des outils de détection de gènes de fusion sur ADN et agrégation des résultats

EAP Guillaume

Mai 2023 à Août 2023

Maîtres de stage : NDIAYE Aminata, BRAYET Jocelyn
Superviseur académique/Enseignant référant : SPADONI Jean-Louis
Établissement/Formation : Cnam Paris – Licence Professionnelle de Bio-informatique
Entreprise d'accueil : MOABI (AP-HP) – 33 boulevard de Picpus, 75012 Paris

REMERCIEMENTS

Je tiens tout d'abord à remercier mes maîtres de stage, Aminata Ndiaye et Jocelyn Brayet, pour m'avoir offert cette incroyable opportunité de travailler au sein de la plateforme bio-informatique MOABI de l'AP-HP.

Leur bienveillance, leur patience et leur disponibilité ont été d'une aide inestimable tout au long de cette période. Ils ont répondu à toutes mes interrogations, ce qui m'a permis de progresser sans jamais me sentir bloqué, favorisant ainsi une avancée fluide et efficace du projet.

Je tiens également à exprimer ma gratitude envers toute l'équipe de la plateforme MOABI et SeqOIA. Leur accueil chaleureux et leur bienveillance ont contribué à rendre mon intégration rapide et agréable. Chaque fois que j'ai eu besoin d'aide, j'ai pu compter sur leur soutien sans faille, dans les meilleures conditions possibles. C'est un environnement de travail que je souhaiterais avoir à l'avenir, lors de ma professionnalisation dans la bio-informatique.

Cette expérience a été extrêmement enrichissante, me permettant d'acquérir de nombreuses connaissances sur le fonctionnement d'une plateforme bio-informatique, l'utilisation d'outils, ainsi que la programmation en bash et Python. Je n'hésite pas à affirmer que ce stage a confirmé mon désir de poursuivre une carrière dans le domaine de la bio-informatique.

L'équipe MOABI et SeqOIA

Table des matières

Glossaire

ADN:	Acide désoxyribonucléique.
ARN :	Acide ribonucléique.
NGS :	Next Generation Sequencing/Séquençage de Nouvelles Génération.
CPU :	Central Processing Unit, puce électronique intégrée à la carte mère et exécute les instructions d'un programme.
RAM :	Random Access Memory, composante essentielle de la mémoire centrale d'un ordinateur
Pipeline :	Terme utilisé dans le domaine informatique et bio-informatique, qui définit une suite d'étapes qui sont exécutées pour atteindre un objectif précis.
Gène de fusion :	Gène issu de la fusion entre deux gènes indépendants.
Fastq :	Format de fichier qui stocke les séquences génétiques et leur score de qualité après un séquençage. Lorsque celui-ci est paired-end, deux fichiers fastq sont obtenus, chacun contenant les données pour un sens.
SAM :	Format de fichier qui stock les résultats d'un alignement des séquences contenues dans les fastq sur un génome de référence.
BAM :	Fichier SAM compressé (binaire).
Reads supporteurs:	Fragments courts d'ADN/ARN obtenus lors d'un séquençage qui mettent en évidence une fusion génique.
Script :	Ensemble d'instructions ou de commandes écrites dans un langage de programmation spécifique.
Parsing :	Processus d'analyse de données complexes pour en extraire et organiser des éléments significatifs.

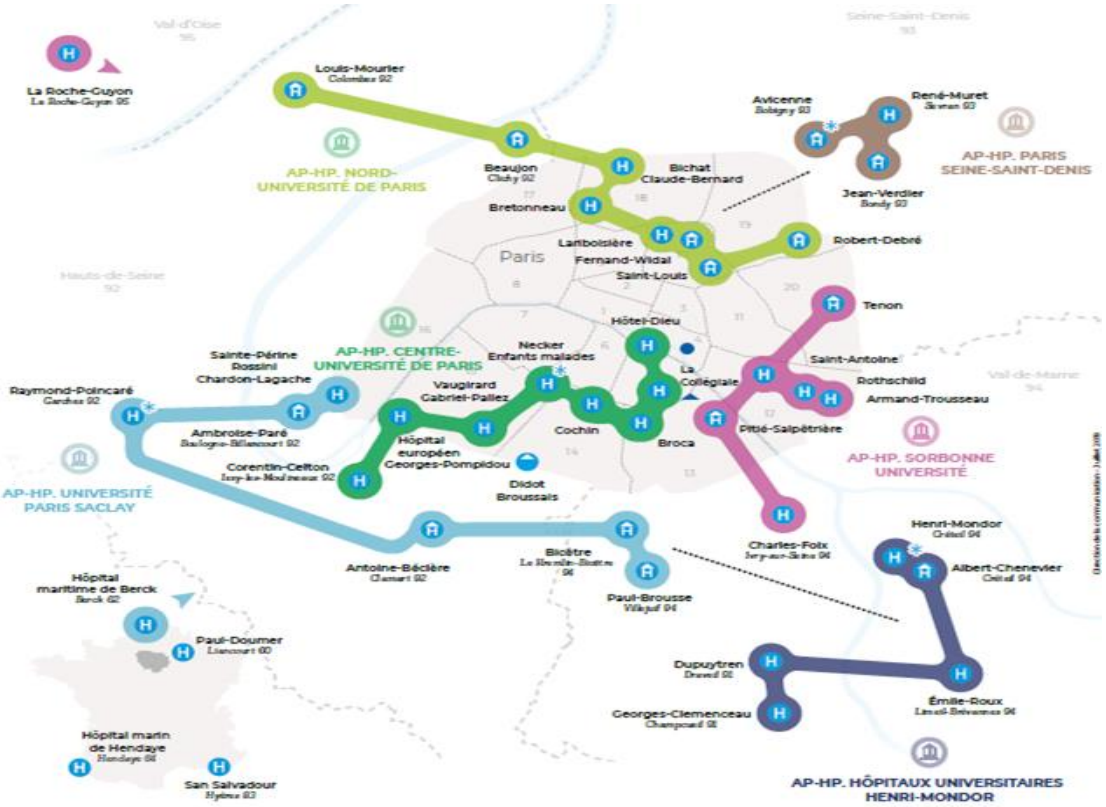


Figure 1 : Hôpitaux AP-HP

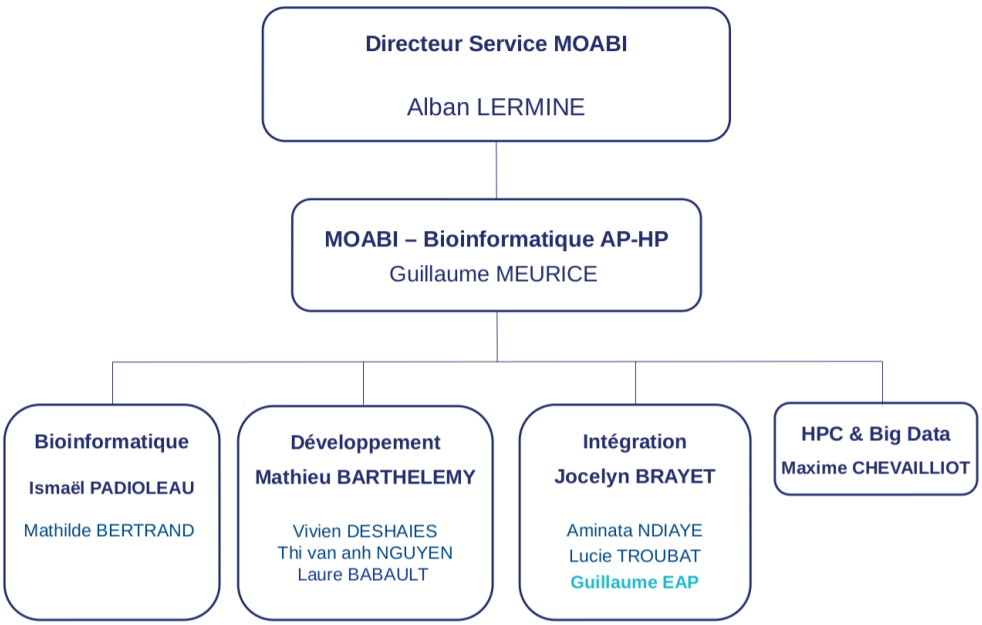


Figure 2 : Organigramme de la plateforme MOABI

1. Introduction

1.1 MOABI : la plateforme bio-informatique de l'AP-HP

Depuis le début des années 2000, la bio-informatique a une importance considérable dans le domaine de la biologie. La gestion et l'analyse de quantités massives de données, en particulier les données de séquençage, sont rendues possibles grâce à l'utilisation d'outils de programmation. Ces outils développés principalement avec Python, R, Perl et C sont intégrés dans des pipelines et utilisés à des fins d'analyse dans le diagnostic et recherche.

C'est dans ce contexte que la plateforme MOABI (Multi-Omics Analytics & BioInformatics) a été fondée à l'AP-HP (Assistance Publique – Hôpitaux de Paris) en 2017. Son objectif est de centraliser et d'analyser les données de séquençage provenant des hôpitaux de l'AP-HP, soit 39 hôpitaux (Figure 1), tout en développant de nouveaux outils et pipelines pour faciliter l'analyse dans le domaine du diagnostic. La plateforme a notamment développé les applications telles que G-route et Leaves, et administre l'interface en ligne, Galaxy, pour faciliter l'utilisation d'outils bio-informatiques par les cliniciens.

Cette plateforme joue également un rôle essentiel dans l'animation de la communauté bio-informatique de l'AP-HP, en offrant un soutien technique et scientifique aux utilisateurs, ainsi que des formations.

La plateforme MOABI reçoit diverses demandes en matière de mise en place solutions pour le diagnostic, telles que la constitution de cohortes de patients, la production et l'analyse de données de séquençage, et le développement de méthodologies de gestion de données.

MOABI est constitué de plusieurs équipes (Figure 2), comprenant des développeurs chargés d'améliorer ou développer les applications, des bio-informaticiens chargés d'accompagner les biologistes dans la validation biologique des résultats issus de pipelines, ainsi que des ingénieurs d'intégration avec lesquels j'ai effectué mon stage. Les ingénieurs d'intégration sont responsables de la construction des pipelines de diagnostic, la recherche et de l'évaluation d'outils et leur installation sur le serveur de calculs.

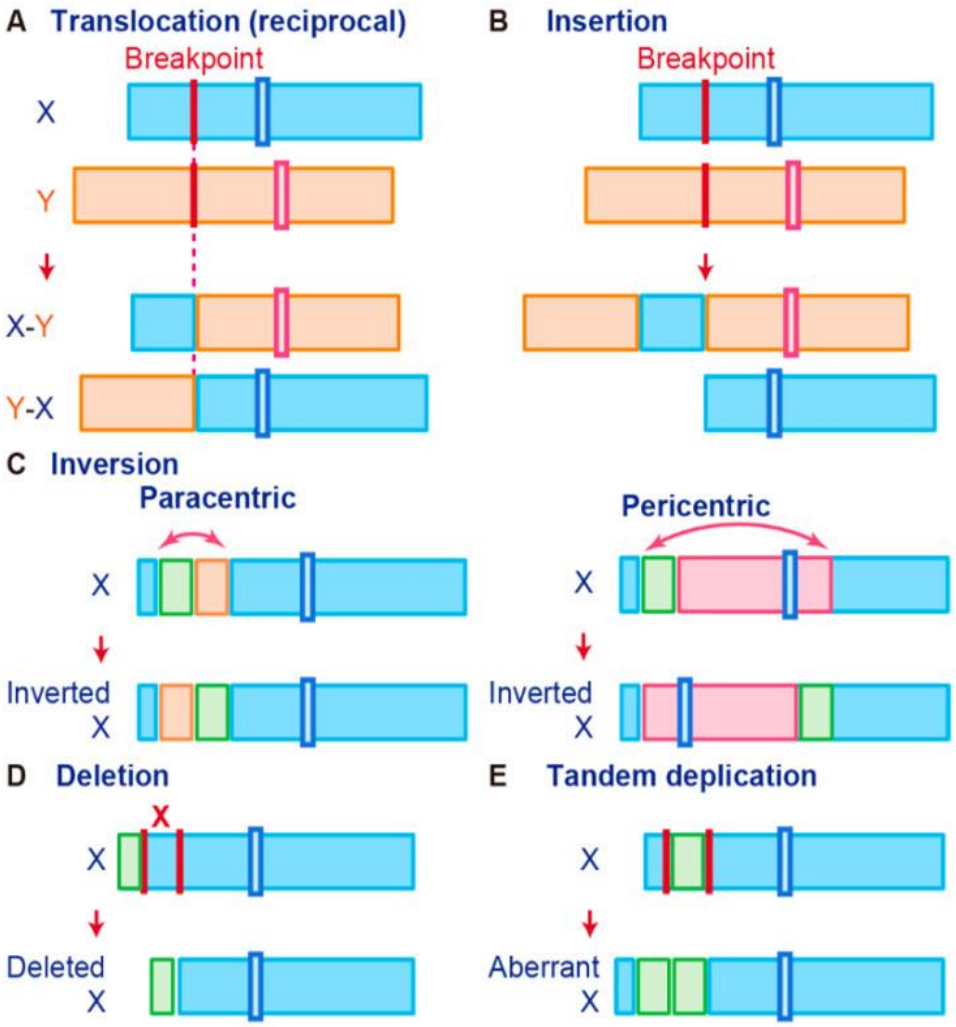


Figure 3 : Anomalies génétiques pouvant entraîner un gène de fusion

A : Translocation (réciproque), échange de segment d'ADN entre deux chromosomes non homologues
B : Insertion, ajout de fragment d'ADN venant d'un autre chromosome
C : Inversion paracentrique, segment chromosomique inversé sans que le centromère soit inclus dans la région inversée | Inversion péricentrique, segment chromosomique inversé avec centromère inclus dans la région inversée
D : Délétion, perte de région d'ADN
E : Duplication en tandem, région d'ADN copiée et insérée à proximité de son emplacement génomique

Le stage que j'ai effectué consiste, dans un premier temps, en la comparaison d'outils de fusions sur ADN ainsi que les validations sur une cohorte de patients dont les fusions de gènes sont connues. Dans un second temps, j'ai dû concevoir un programme en Python qui regroupe les résultats des différents outils dans un seul fichier structuré.

1.2 Les gènes de fusion

Un gène de fusion est un gène hybride résultant de la fusion de deux gènes indépendants. Cette anomalie génétique peut résulter de translocations, de délétions ou d'inversions chromosomiques comme démontré dans la Figure 3. Dans la plupart des cancers, on peut retrouver des réarrangements chromosomiques et/ou des cassures occasionnant des fusions de gènes. L'étude de ce phénomène se situe donc dans le diagnostic médical.

Dépendant d'où se situe le "breakpoint", c'est-à-dire la région de fusion des deux gènes, plusieurs conséquences peuvent être retrouvées sur le phénotype d'un individu. En effet, si le breakpoint se situe sur une région intronique par exemple, il peut y avoir une dérégulation de l'expression de la protéine codée car celui-ci sera sous contrôle de la régulation de l'autre gène. À l'inverse, si le breakpoint se situe sur une séquence codante, une nouvelle protéine sera obtenue avec une nouvelle fonction potentiellement nocive pour l'individu.

L'étude des gènes de fusion se fait le plus souvent au niveau du transcriptome car le séquençage ARN est moins coûteux qu'un séquençage complet du génome. Par conséquent, la grande majorité des outils bio-informatiques ne prend en entrée que des données de séquençage ARN ou est plus optimisée pour les données ARN, mais peut cependant prendre en entrée des données de séquençage ADN.

À MOABI plusieurs pipelines ont été développés pour des données de séquençage ARN ou ADN en entrée. Afin d'avoir plus de choix au niveau de l'utilisation d'outils et d'améliorer la sensibilité de la détection des gènes de fusions dans les pipelines actuels, l'objectif du stage est de trouver et intégrer de nouveaux outils prenant en entrée des données ADN, sachant qu'il y a déjà énormément d'outils pour analyse de gène de fusion avec données de transcriptome, puis de proposer un résultat uniformisé provenant de ces outils d'intérêt.

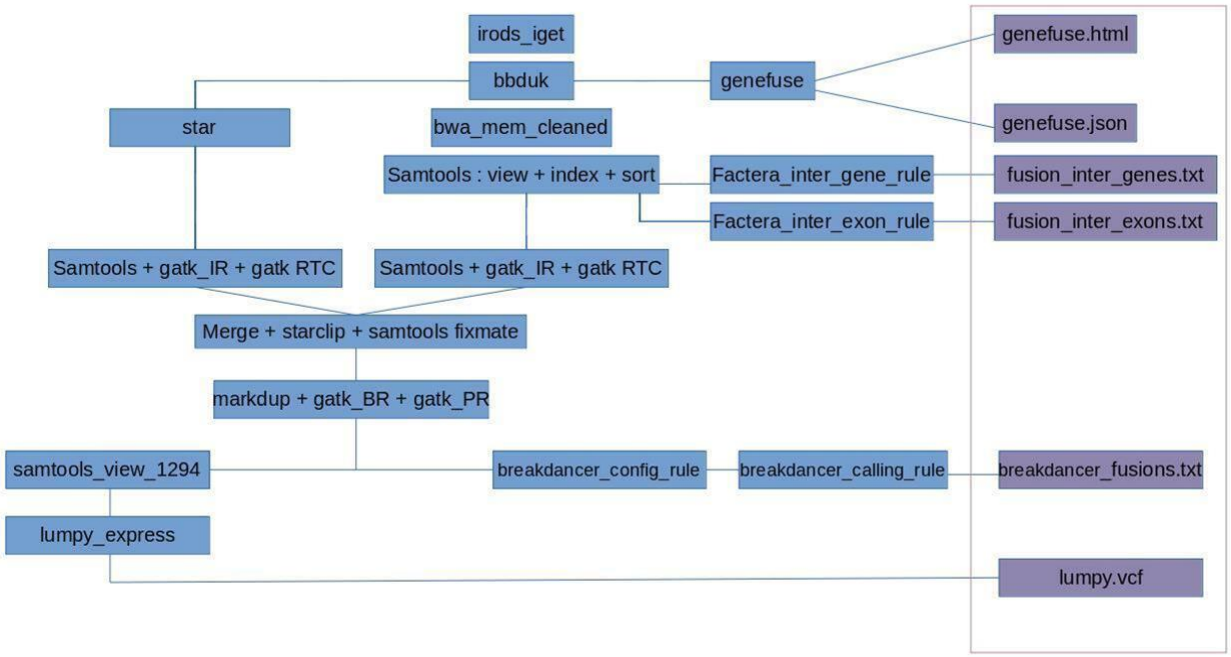


Figure 4 : Pipeline de MOABI pour identification de gène de fusion avec données ADN

Outil	Fusionfusion	SplitFusion	Factera	Breakdancer	Lumpy-sv
Prend en entrée	ARN ou ADN				
Formats initials des fichiers d'entrée	fastq	fastq ou BAM	BAM	BAM	BAM
Fichier de sortie d'intérêt	fusion_fusion.result.txt	reads.fusion.table.NoFilter.txt	Échantillon_bwa_mem_F256_sorted.factera.fusions.txt	Échantillon_breakdancer_fusions.txt	Échantillon.bam.vcf
Date de la dernière mise à jour	2022	2023	2021	2015	2022
Lien	https://github.com/Genomon-Project/fusionfusion	https://github.com/Zheng-NGS-Lab/SplitFusion	https://factera.stanford.edu/	https://github.com/genome/breakdancer	https://github.com/arq5x/lumpy-sv

Tableau 1 : Outils d'intérêt (fusionfusion, SplitFusion) et outils déjà intégrés dans les pipelines MOABI (Factera, Breakdancer, Lumpy-sv)

La plateforme utilise déjà les outils Factera, Breakdancer, et Lumpy-sv pour les données ADN dans leurs pipelines (Figure 4). Factera offre plusieurs options d'analyse, notamment l'option InterGene qui se focalise sur les fusions dans l'entièreté d'un gène et l'option InterExon qui se concentre sur les cas de fusions exoniques.

2. Matériels et méthodes

24 échantillons constituant la cohorte de patients, dont la fusion BCR-ABL1 est attendue à des coordonnées breakpoint spécifiques, ont été fournis pour validation. Chaque échantillon est constitué de deux fichiers fastq (R1 et R2), leur séquençage étant paired-end, et d'un fichier BAM déjà aligné sur un génome de référence (hg19).

Pour cette validation, les outils d'intérêt ainsi que des outils déjà intégrés dans le pipeline de la plateforme ont été utilisés. Pour l'outil Factera, les options InterGene et InterExon ont toutes les deux été testées.

Un script bash a été développé afin de préparer les fichiers d'entrée pour chacun des outils puis lancer l'analyse de façon automatique sur chaque échantillon. Après cela, afin d'obtenir une vue d'ensemble des résultats, un script Python a été écrit pour parser les informations essentielles des fichiers de sortie.

2.1 Les outils de détections de gènes de fusion

Deux outils d'intérêt prometteurs ont été trouvés puis testés sur 3 échantillons fournis et validés au niveau de leur fonctionnement et résultats : fusionfusion et SplitFusion. Ces deux outils ont donc été utilisés pour la validation des échantillons avec les outils déjà intégrés dans les pipelines. Sur le Tableau 1, les informations sur ces outils peuvent être vues telles que le format du fichier d'entrée de chaque échantillon, le fichier de sortie contenant les informations sur les gènes de fusions de chaque outil, la date de la dernière mise à jour et le lien du site où l'outil peut être récupéré.

fusionfusion --star star.Chimeric.out.sam --out output_dir --reference_genome reference.fa

Options:
[--genome_id {hg19,hg38,mm10}]
[--pooled_control_file POOLED_CONTROL_FILE] [--no_blat]
[--debug] [--abnormal_insert_size ABNORMAL_INSERT_SIZE]
[--min_major_clipping_size MIN_MAJOR_CLIPPING_SIZE]
[--min_read_pair_num MIN_READ_PAIR_NUM]
[--min_valid_read_pair_ratio MIN_VALID_READ_PAIR_RATIO]
[--min_cover_size MIN_COVER_SIZE]
[--anchor_size_thres ANCHOR_SIZE_THRES]
[--min_chimeric_size MIN_CHIMERIC_SIZE]
[--min_allowed_contig_match_diff MIN_ALLOWED_CONTIG_MATCH_DIFF]
[--check_contig_size_other_breakpoint CHECK_CONTIG_SIZE_OTHER_BREAKPOINT]
[--filter_same_gene]
[--star_sj_tab star.SJ.out.tab]
[--star_aligned_bam star.Aligned.sortedByCoord.out.bam]

Figure 5 : Ligne de commande basique et options pour l'outil fusionfusion

python3 /path/to/SplitFusion/exec/SplitFusion.py --refGenome reference.fa --annovar Snpeff --output output_dir --sample_id sample --fastq_file1 fastq1 --fastq_file2 fastq2 --thread cpu

Options:
[--bam_file BMS_FILE] [--fastq_file1 FASTQ_FILE1] [--fastq_file2 FASTQ_FILE2]
[--panel_dir PANEL_DIR] [--panel PANEL] [--steps STEPS]
[--AnnotationMethod ANNOTATIONMETHOD] [--thread THREAD]
[--minMQ MINMQ] [--minMQ1 MINMQ1]
[--minMapLength MINMAPLENGTH]
[--minMapLength2 MINMAPLENGTH2]
[--maxQueryGap MAXQUERYGAP] [--maxOverlap MAXOVERLAP]
[--minExclusive MINEXCLUSIVE]
[--FusionMinStartSite FUSIONMINSTARTSITE]
[--minPartnerEnds_BothExonJunction MINPARTNERENDS_BOTHEXONJUNCTION]
[--minPartnerEnds_OneExonJunction MINPARTNERENDS_ONEEXONJUNCTION]

Figure 6 : Ligne de commande basique et options pour l'outil SplitFusion

2.1.1 Fusionfusion

Fusionfusion détecte les gènes de fusion en prenant en entrée un génome de référence, un répertoire de sortie et un fichier star.Chimeric.out.sam produit par l'outil STAR. Ce dernier contient des informations d'alignement pour les lectures chimériques. Les lectures chimériques sont des lectures qui s'alignent sur des régions génomiques différentes et qui suggèrent une potentielle éventualité de fusion ou réarrangement à cet endroit. STAR est plus spécialisé pour les données de transcriptome mais reste utilisable pour les données ADN.

À partir de ce fichier, fusionfusion simplifie l'analyse en fournissant directement les gènes de fusion détectés, ainsi que leur emplacement chromosomique et les noms des gènes impliqués dans la fusion.

Fusionfusion propose d'autres options dans la ligne de commande (Figure 5), seuls les éléments d'entrée obligatoires ont été donnés ici pour faire fonctionner l'outil de façon basique.

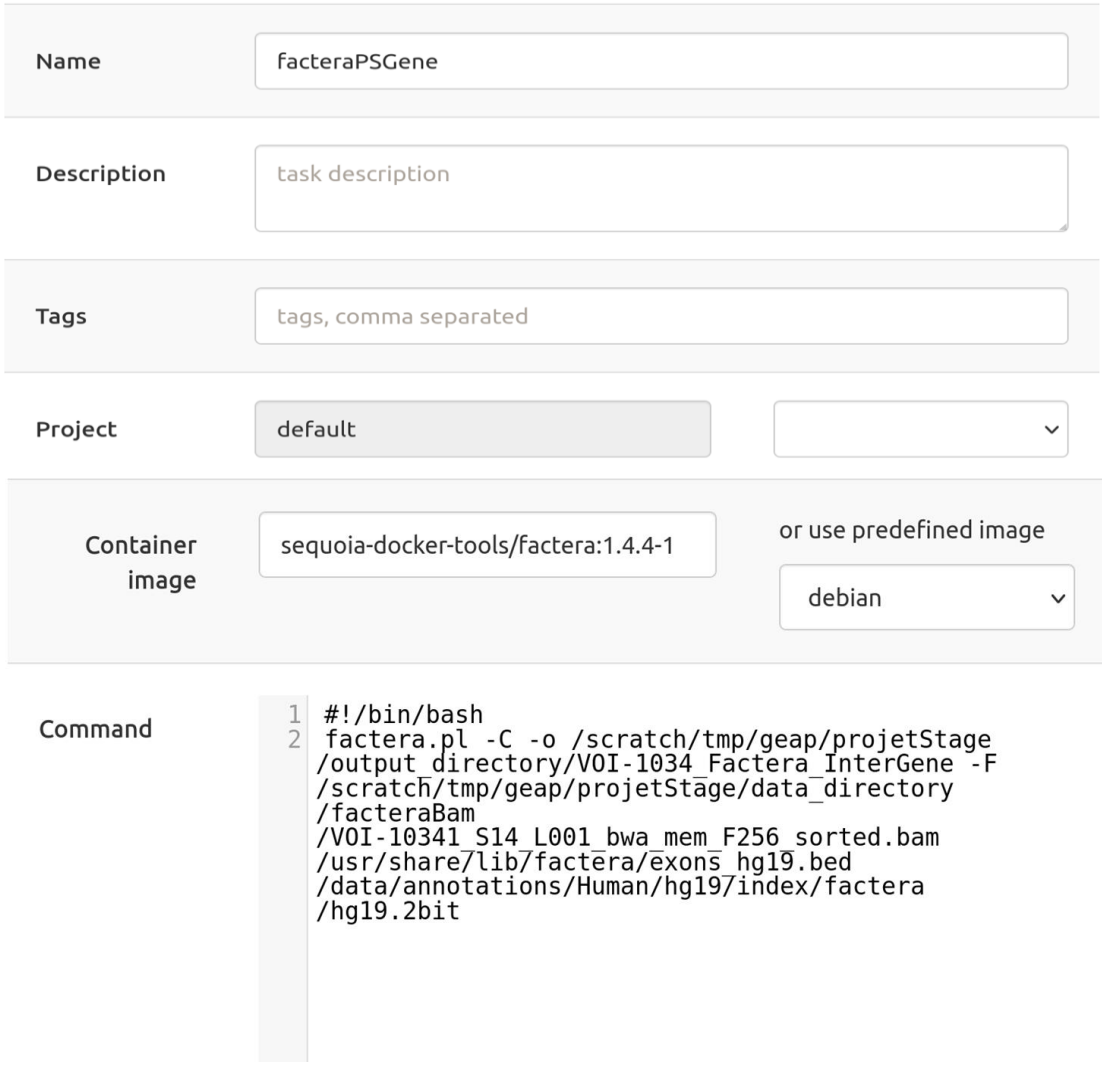
2.1.2 SplitFusion

SplitFusion nécessite plusieurs dépendances pour être installé, dont les outils bio-informatiques Samtools, BWA, Bedtools, Annovar, ainsi que les langages de programmations Perl, Python, et R. Il est aussi plus spécifique aux données de transcriptomes mais prend tout de même en entrée les données ADN et donne des résultats exploitables. Comme fusionfusion, SplitFusion propose d'autres options dans la ligne de commande (Figure 6).

En entrée, cet outil prend obligatoirement un génome de référence, le nom d'échantillon, les fichiers fastq ou le fichier BAM de l'échantillon, le nombre de CPU, et le chemin de sortie. Pour les tests que nous avons effectués, nous sommes directement partis des fichiers BAM.

En sortie, on obtient de nombreux fichiers mais un seul contient les résultats de gènes de fusions. Dans ce fichier, on obtient les informations telles que l'identifiant de l'échantillon, les gènes de fusion trouvés, le cadre de lecture, la fonction potentielle ou rôle biologique des gènes si la fusion est déjà connue.

A



B

Figure 7 : Interface Go-docker
A : Création d'un job
B : Fin d'un job

2.2 Lancement et automatisation des outils

2.2.1 GitLab, GO-Docker et les serveurs MOABI/AP-HP

Les outils sont installés dans le GitLab de la plateforme MOABI, qui est une interface basée sur Git et qui possède une multitude de fonctionnalités pour la gestion de projets de développement logiciel. Elle propose un wiki intégré, un système de suivi des bugs, l'intégration continue et la livraison continue. Cette interface est utilisée par MOABI pour stocker les outils et leurs dépendances dans des conteneurs.

Ces conteneurs sont très efficaces pour l'utilisation d'outils d'un pipeline, car ils garantissent une exécution optimale et indépendante de l'environnement hôte. En fixant les versions des outils et des bibliothèques nécessaires, on évite les conflits entre dépendances qui pourraient causer le dysfonctionnement des outils. Chaque version d'outil est associée à une image de conteneur spécifique, permettant un accès facile et contrôlé aux outils nécessaires.

Les pipelines sont également stockés dans GitLab. Avant de déployer un outil ou un pipeline sur le cluster, un test fonctionnel du conteneur est effectué. Si le test échoue, l'outil ne pourra pas être utilisé sur le cluster (cela ne veut pas dire que l'outil donnera les résultats attendus mais qu'il est simplement fonctionnel).

Le lancement d'outils ou de pipelines sur le cluster s'effectue à l'aide de Go-Docker, une interface qui accède à des images de conteneurs d'outils et exécute une tâche sur un cluster. Pour lancer un processus, vous devez fournir le nom du job, les tags associés pour une recherche plus facile ultérieurement, un lien vers l'image GitLab correspondante à l'outil, ainsi que le nombre de CPU et de RAM alloués à l'exécution par le cluster (Figure 7A). Lorsqu'un job est lancé, il est affiché dans une section dédiée avec l'état "en cours" et un numéro d'identification. Une fois terminé, le job est déplacé vers la section des jobs terminés (Figure 7B).

Il y a accès à deux journaux : l'un pour les erreurs, qui affiche les messages d'erreur en cas de dysfonctionnement du job, et l'autre pour les sorties standards, qui affiche les messages qu'il y a eus du début jusqu'à la fin.

Une autre option pour lancer des outils via Go-Docker est d'utiliser un serveur de la plateforme. Dessus, les jobs peuvent être lancés en utilisant l'utilitaire "godjob" et en fournissant les

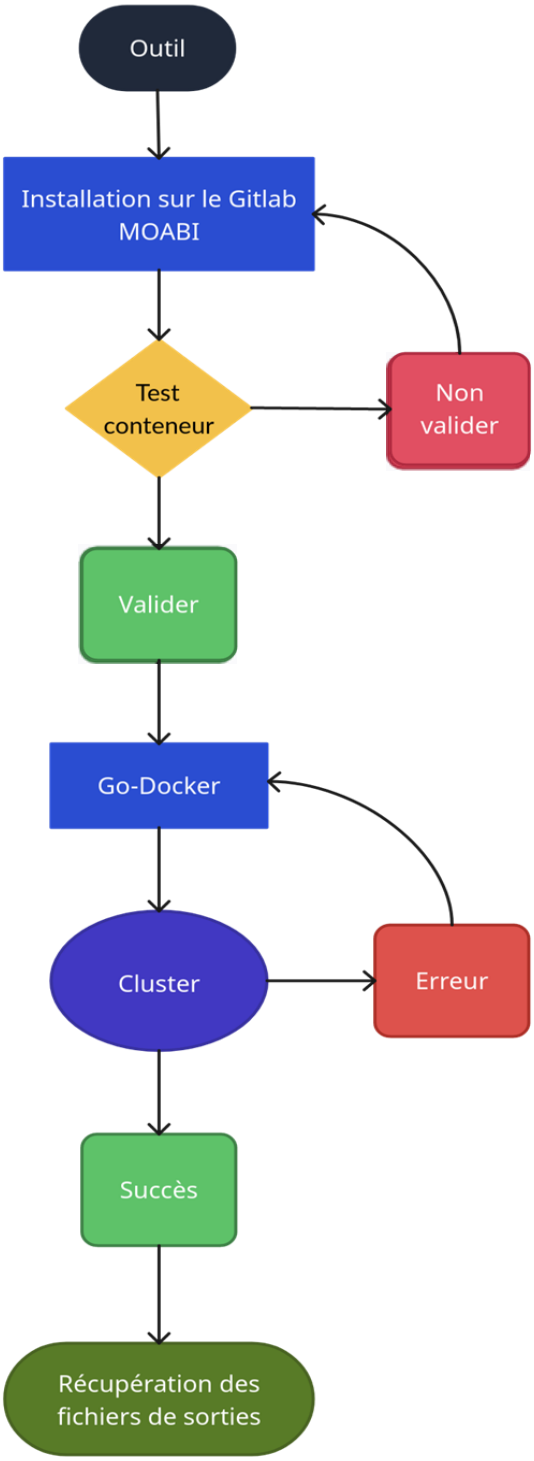


Figure 8 : Schéma du protocole de lancement d'outils

mêmes éléments d'entrée que ceux utilisés sur l'interface web de Go-Docker. Cela offre une alternative pratique pour lancer les jobs dans le cas où il y aurait des problèmes d'accès direct à Go-Docker via un navigateur web.

Les serveurs de la plateforme offrent également d'autres fonctionnalités, telles que le transfert de données et la manipulation de fichiers. Chaque membre dispose d'un espace personnel dédié où il peut stocker des documents et des fichiers.

Pour résumer, l'intégration et les tests fonctionnels de lancement d'outils et de pipelines sont gérés sur GitLab, tandis que Go-Docker est utilisé pour exécuter efficacement ces outils et pipelines en utilisant le cluster (Figure 8). Le lancement des outils via la ligne de commande sur un serveur offre une alternative à l'interface web de Go-Docker.

2.2.2 Le cluster MOABI/AP-HP

Les outils utilisés sur la plateforme MOABI sont particulièrement adaptés à la manipulation de données volumineuses, nécessitant une grande quantité de mémoire RAM et l'utilisation de plusieurs CPU. Pour surmonter les limitations d'un ordinateur individuel, la plateforme dispose d'un cluster de calculs.

Ce cluster est composé de 1800 CPU et 7To de RAM (1To = 1000Go) répartis en 43 serveurs. Ces serveurs sont interconnectés par des câbles réseau haut débit et équipés de commutateurs puissants. Cette infrastructure robuste permet d'exécuter de multiples pipelines de manière consécutive ou simultanée.

La supervision de l'activité de la plateforme peut être suivie en direct. Les réservations et l'utilisation des CPUs et de la RAM, ainsi que l'état d'avancement des pipelines, sont affichées sur un écran situé dans les locaux, visibles par tous. Pour la plateforme, cela permet de monitorer en direct l'exécution de tous les pipelines et de réagir au plus vite en cas d'erreur.

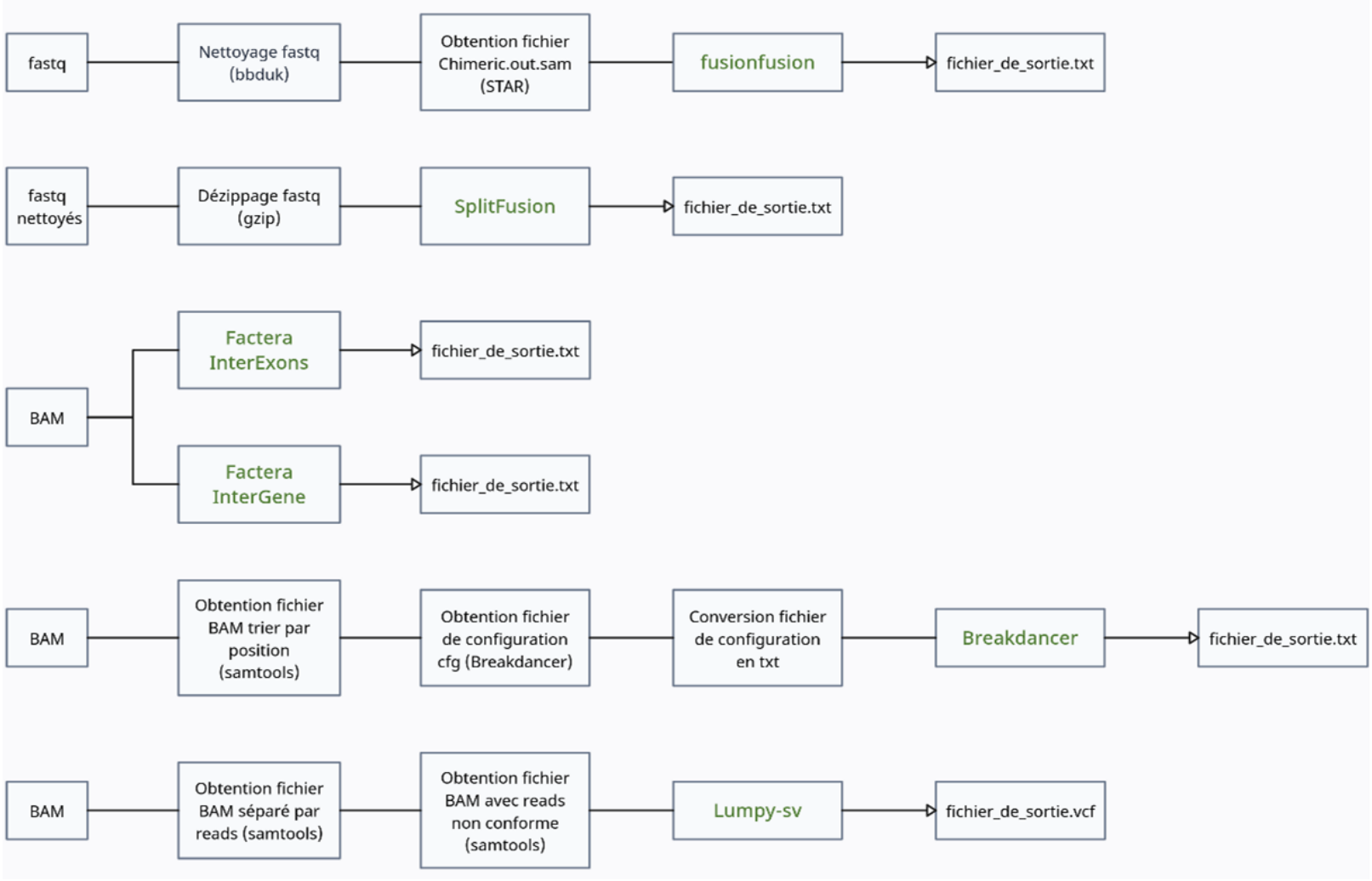


Figure 9 : Schéma des étapes du script d'automatisation

2.2.3 Script d'automatisation

Dans le but d'optimiser la procédure et gagner du temps, un programme d'automatisation a été développé pour exécuter les outils sur les 24 échantillons. Ce script, écrit en bash (code en Annexe 1), prend deux arguments : le répertoire "data_directory" qui contient les données d'entrée et le répertoire "output_directory" qui stocke les données de sortie. Il se sert de la commande godjob pour lancer un outil. La Figure 9 présente la séquence d'exécution des outils et leur préparation.

Le répertoire "data_directory" contient les données d'entrée initiales, c'est-à-dire les fichiers fastq qui ont été séparés en deux répertoires distincts selon les lectures (reads R1 et R2), ainsi que les fichiers BAM rassemblés dans un seul répertoire. "data_directory" inclut également les données préparées pour les outils. Par exemple, pour l'outil fusionfusion, le répertoire de sortie provenant de STAR, qui contient le fichier star.Chimeric.out.sam pour un échantillon donné, est placé dans ce répertoire de fichiers d'entrée.

Le répertoire "output_directory" contient plusieurs sous-répertoires, chacun contenant les fichiers de sortie des outils. Chaque sous-répertoire peut contenir un ou plusieurs fichiers, mais seul un fichier par outil contient les informations d'intérêt.

2.2.4 Script de parsing

Afin d'obtenir et analyser les résultats d'intérêt de tous les échantillons, un script rédigé en Python a été mis en place. Il permet de rassembler les éléments essentiels venant de l'ensemble des outils de fusions utilisés pour notre série de validation dans un tableau Excel (code en Annexe 2).

Parmi ces informations, nous avons : les coordonnées du breakpoint du gène1 et gène 2, les noms des gènes impliqués dans la fusion, et le nombre de reads supporteurs. Un système de score a été mis en place afin d'indiquer pour chaque fusion le nombre d'outils l'ayant trouvée.

Pour son exécution, plusieurs paramètres sous forme d'arguments doivent être fournis au programme : le nom de l'échantillon, le fichier de sortie d'intérêt de chaque outil, la différence maximum de variation autorisée entre deux paires de coordonnées pour qu'elles soient considérées comme équivalentes et un répertoire de sortie (Figure 10).

ParsingScript.py -n sample_name -f fusionfusion_output_file -s splitfusion_output_file -g facteraIntergene_output_file -e facteraInterExon_output_file -b breakdancer_output_file -l lumpy_output_file -d max_acceptable_diff -o output_directory

Figure 10 : Ligne de commande du script de parsing de résultats

Rang	**Outil**	**Score**
1	Lumpy-sv	23/24
2	SplitFusion	22/24
3	Fusionfusion/Factera (InterGene/InterExon)	20/24
4	Breakdancer	0/24

Tableau 2 : Classement des outils par ordre décroissant du score

Le fonctionnement du script est le suivant : pour chaque outil, un dictionnaire est créé dans le but d'y enregistrer chaque fusion détectée. Les paires de coordonnées uniques à chaque fusion servent de clé dans ce dictionnaire. Le script parcourt ensuite les fichiers de sortie un par un pour chaque échantillon et vérifie si d'autres paires de coordonnées sont équivalentes à une clé existante dans le dictionnaire. Une équivalence est validée si la différence entre chaque breakpoint de deux paires de coordonnées est inférieure ou égale à la différence maximum de variation autorisée donnée en argument. Si c'est le cas, les détails de la fusion sont ajoutés sous cette clé, en indiquant également l'outil correspondant.

Si les coordonnées ne correspondent à aucune clé existante, elles sont ajoutées comme une nouvelle clé dans le dictionnaire, accompagnées de toutes les informations de fusion qui s'y rapportent. Un score est ensuite attribué pour le nombre de fois que la fusion a été trouvée. L'ensemble des résultats fournis par ce programme est dressé dans un fichier Excel.

Ce sera ensuite la responsabilité des biologistes, ayant à leur disposition les noms et les coordonnées des gènes de fusion d'intérêt, d'évaluer si les fusions attendues ont été détectées.

3. Résultats

3.1 Score des outils

Le Tableau 2 résume le score obtenu par chaque outil par ordre décroissant. Le score attribué correspond au nombre de fois que l'outil a trouvé la fusion attendue pour chacun des 24 échantillons.

Fusionfusion a réussi à valider 20 échantillons sur 24, tandis que SplitFusion en a validé 22 sur 24. Les outils Factera InterGene et Factera InterExon ont également réussi à valider 20 échantillons sur 24, les même que fusionfusion. Pour sa part, Lumpy-sv a réussi à en valider 23 sur 24. En revanche, Breakdancer n'a pu valider aucun des échantillons.

En résumé, l'outil Lumpy-sv se distingue en ayant réussi à valider le plus grand nombre d'échantillons, suivi de près par SplitFusion. Fusionfusion et Factera InterGene/InterExon ont obtenu le même score en validant les mêmes échantillons. Enfin, Breakdancer n'a abouti à la validation d'aucun échantillon.

	A	B	C	D	E	F	G
1	**Fusion found**	**tool name**	**fusion name**	**breakpoint1**	**breakpoint2**	**supporting reads(* = paired-end reads data)**	**Score**
2	chr9 133597064 chr22 23632541	factera intergene	ABL1-BCR	chr9 133597064	chr22 23632541	23	10
3		lumpy-sv	NA	chr9 133597064	chr22 23632541	34*	
4		lumpy-sv	NA	chr22 23632541	chr9 133597064	34*	
5		lumpy-sv	NA	chr9 133597063	chr22 23632545	7*	
6		lumpy-sv	NA	chr22 23632545	chr9 133597063	7*	
7		SplitFusion	DRICH1 upstream---FAM163B intronic	chr22 23632541	chr9 133597064	120	
8		SplitFusion	DRICH1 upstream---FAM163B intronic	chr22 23632545	chr9 133597063	52	
9		SplitFusion	FAM163B intronic---DRICH1 upstream	chr22 23632543	chr9 133597065	1	
10		factera interexons	NA	chr9 133597064	chr22 23632545	23	
11		fusionfusion	BCR-ABL1	chr22 23632541	chr9 133597064	64*	
12	chr12 114309629 chr9 134289797	factera intergene	RBM19-PRRC2B	chr12 114309629	chr9 134289797	3	2
13		factera interexons	NA	chr9 134289797	chr12 114309629	2	
14	chr8 46844276 chr8 43793661	factera intergene	KIAA0146-POTEA	chr8 46844276	chr8 43793661	2	1
15	chr8 46848705 chr8 43835302	factera intergene	KIAA0146-POTEA	chr8 46848705	chr8 43835302	1	1

Figure 11 : Vue partielle du tableau Excel produit par le script de parsing

chr10	94295003	-	chr19	7806233	-	---	IDE	---	CD209	---	10
chr22	23633827	-	chr9	133622802	+	---	BCR	---	ABL1	---	17
chr22	23633827	+	chr9	133622805	-	---	BCR	---	ABL1	---	16
chr4	9405093	+	chr5	170837719	+	---	---	---	NPM1	---	30

Figure 12 : Contenu du fichier de résultat fusionfusion

Nom des colonnes (non inclus dans le fichier):
Breakpoint_1 Coordonnées_1 Orientation_breakpoint_1 Breakpoint_2 Coordonnées_2 Orientation_breakpoint_2 Nucléotides_insérés_dans_les_breakpoints Gène_1 Jonction_exon-intron_chevauchant_le_breakpoin_1 Gène_2 Jonction_exon-intron_chevauchant_le_breakpoint_2 Nombre_de_paired-end_reads_supporteurs

3.2 Script de parsing

La Figure 11 présente un exemple de tableau généré par le script de parsing pour un échantillon. Les fusions sont classées en ordre décroissant en fonction de leur score, qui indique combien de fois une fusion a été détectée. La fusion attendue, BCR-ABL1, est celle ayant obtenu le score le plus élevé, soit 10. Lumpy-sv l'a identifiée 4 fois, SplitFusion 3 fois, et les autres outils (fusionfusion, Factera InterGene/InterExon) l'ont identifié qu'une seule fois chacun. En deuxième position, la fusion RBM19-PRRC2B a été repérée 1 fois par Factera InterGene et 1 fois par Factera InterExon, totalisant un score de 2. Les fusions suivantes, en dessous de celles-ci, sont celles détectées une seule fois par un unique outil.

On remarque des divergences au niveau des coordonnées et des reads supporteurs pour une fusion identifiée à plusieurs reprises par un seul outil. Bien que la raison de ces divergences reste inconnue, ces fusions ont néanmoins été incluses pour l'analyse par les biologistes.

Dans l'ensemble, 509 fusions ont été découvertes (voir Annexe 3 pour le fichier Excel). Plus de la moitié d'entre elles, soit 263, proviennent de SplitFusion. Ensuite, Breakdancer a identifié 228 fusions. Factera InterExon en a trouvé 5, tandis que Factera InterGene et Lumpy-sv en ont trouvé 4 chacun. Fusionfusion n'en a repéré qu'une seule. Cette tendance se répète pour la plupart des échantillons, où SplitFusion et Breakdancer détectent le plus grand nombre de fusions.

4. Discussion

4.1 Comparaison des outils

4.1.1 Fusionfusion

Fusionfusion se distingue par sa simplicité et son efficacité en fournissant les coordonnées génomiques de manière claire ainsi que le nom attendu de la fusion (Figure 12), comparé à Factera InterExon, Breakdancer et Lumpy-sv qui ne proposent aucun nom de gènes et fournissent des informations non pertinentes pour les cas de gènes de fusion (comme d'autres aberrations génétiques). À l'exception de Lumpy-sv et SplitFusion, les autres outils ont obtenu des résultats similaires à fusionfusion.

SampleID GeneExon5_GeneExon3 frame num_partner_ends num_unique_reads exon.junction breakpoint transcript_5 transcript_3 function_5 function_3 gene_5 cdna_5 gene_3 cdna_3 intragene known

reads ENPP7P13-MIR9901_intergenic---NAT1_intronic N.A. 1 1 One chr16_33964528__chr8_18170507 - NM_001291962 intergenic intronic ENPP7P13-MIR9901 - NAT1 - 0 0

reads FAM8A1_exon1---VCY-NLGN4Y_intergenic N.A. 1 1 One chr6_17601123__chrY_14442869 NM_016255 - exonic intergenic FAM8A1 714 VCY-NLGN4Y - 0 0

reads DRICH1_upstream---SARDH_intronic N.A. 151 756 0 chr22_23632423__chr9_133666368 NM_016449 NM_007101 upstream intronic DRICH1 - SARDH - 0 0

reads TARID_ncRNA_intronic---WDR62_intronic N.A. 24 71 0 chr19_36066506__chr6_133594148 NR_109982 NM_173636 ncRNA_intronic intronic TARID - WDR62 - 0 0

Figure 13 : Vue partielle du fichier de résultat SplitFusion

A

Est_Type Region1 Region2 Break1 Break2 Break_support1 Break_support2 Break_offset Orientation Order1 Order2 Break_depth Proper_pair_support Unmapped_support Improper_pair_support Paired_end_depth Total_depth Fusion_seq Non-templated_seq

TRA ABL1 BCR chr9:133622803 chr22:23633828 30 94 1 1+ 2+ NC CN 74 63 0 11 15 CCCTGTCTCAAAAAACCAAAACAAAATCAAGTCTCCCTCAGTCTTCCTCT <> CTCTTCCTTGCCCCGTGCACTCAACCTTGCATCCCCAAACCAAACCTATT -

B

Est_Type Region1 Region2 Break1 Break2 Break_support1 Break_support2 Break_offset Orientation Order1 Order2 Break_depth Proper_pair_support Unmapped_support Improper_pair_support Paired_end_depth Total_depth Fusion_seq Non-templated_seq

TRA chr9:133589706 chr22:23634727 chr9:133622803 chr22:23633828 30 79 1 1+ 2+ NC CN 76 65 0 11 16 CCCTGTCTCAAAAAACCAAAACAAAATCAAGTCTCCCTCAGTCTTCCTCT <> CTCTTCCTTGCCCCGTGCACTCAACCTTGCATCCCCAAACCAAACCTATT -

Figure 14 : Vue partielle d'un fichier de résultat Factera InterGene (A)
et Factera InterExon (B)

4.1.2 SplitFusion

SplitFusion permet également d'obtenir la fusion attendue, tout en fournissant les coordonnées des chimères avec les noms des gènes correspondant aux exons 5 et exon 3 du gène de fusion. Il détecte un plus grand nombre de fusions que les autres outils et fournit des informations supplémentaires, comme la fonction des exons 5 et exon 3, ainsi que la reconnaissance de fusions déjà connues.

Cependant, il convient de noter que la présentation des résultats peut parfois être difficile à comprendre, comme vu dans la Figure 13. L'alignement des informations avec les résultats n'est pas optimisé pour une compréhension aisée, ce qui peut nécessiter des efforts supplémentaires pour interpréter les données correctement. De plus, les noms des gènes correspondant aux exons 5 et exon 3 du gène de fusion ne sont pas ceux attendus. La validation d'un échantillon se fait donc en regardant directement si les coordonnées attendues sont présentes.

En ce qui concerne la recherche de fusions, SplitFusion a réussi à identifier deux fusions attendues que les autres outils, à l'exception de Lumpy-sv, n'ont pas pu détecter.

4.1.3 Factera

L'outil Factera qui offre deux options, InterGene et InterExon, donne des résultats satisfaisants pour les deux options. En effet, 20 échantillons sur 24 ont été validés, les mêmes que fusionfusion. Cela confirme que pour ces 20 échantillons la fusion BCR-ABL1 est dans une région exonique. Au niveau de la présentation des fusions, InterGene et InterExon fournissent les mêmes informations mais avec une différence. InterGene donne en région le nom du gène (exemple : région 1 = BCR, région 2 = ABL1) tandis que InterExon donne pour ces régions des coordonnées dans lesquels les coordonnées breakpoint se situent (Figure 14).

InterGene permet donc de retrouver plus facilement le gène attendu et InterExon permet de savoir si la fusion se situe dans une région exonique.

4.1.4 Breakdancer

L'outil Breakdancer n'a produit aucun résultat valide, avec des divergences de 100pb à 1000pb par rapport à ce qui est attendu pour une coordonnée de breakpoint. Dans le meilleur des cas, il retrouve une coordonnée de breakpoint acceptable ainsi que les chromosomes attendus.

#Chr1 Pos1 Orientation1 Chr2 Pos2 Orientation2 Type Size Score num_Reads num_Reads_lib GBO-8774_Pos.bam
chr1 36931519 6221+501- chr1 36933964 6221+501- ITX -112 99 493 /scratch/tmp/geap/projetStage/data_directory/BamByPos/GBO-8774_Pos.bam|493 NA

chr9 133622690 20+1- chr22 23633243 4087+277- CTX -273 99 10 /scratch/tmp/geap/projetStage/data_directory/BamByPos/GBO-8774_Pos.bam|10

Figure 15: Vue partielle d'un fichier de résultat
Breakdancer avec un cas d'anomalie génétique et un cas de fusion

#CHROM POS ID REF ALT QUAL FILTER INFO FORMAT ADE-9743_S7_L001
chr11 534521 2 N <DEL> . . SVTYPE=DEL;STRANDS=+-:4;PE=4

chr9 133666516 4_1 N N[chr22:23632272[. . SVTYPE=BND;STRANDS=+-:142;PE=142

Figure 16 : Vue partielle d'un fichier de résultat
Lumpy-sv avec un cas d'anomalie génétique et un cas de fusion

Par conséquent, il est possible de supposer qu'il est nécessaire d'utiliser des options supplémentaires, au-delà de celles par défaut, pour que Breakdancer puisse fournir des résultats valides et affirmer sa recherche de fusion. Cependant, il est important de souligner que cette nécessité de configuration supplémentaire remet en question l'efficacité d'utilisation de Breakdancer par rapport aux outils d'intérêt fusionfusion et SplitFusion.

Sa présentation des résultats, quant à elle, est similaire à Lumpy-sv et ne présente pas seulement des fusions mais aussi d'autres types de variants structuraux (Figure 15).

4.1.5 Lumpy-sv

En fin de compte, Lumpy-sv s'est avéré être l'outil qui a confirmé la validité du plus grand nombre d'échantillons, soit 23 sur 24. Dans certains cas, Lumpy-sv a été le seul outil capable de confirmer la validité des échantillons, et l'unique échantillon qu'il n'a pas validé n'a pas été confirmé par les autres outils non plus. Selon les informations transmises par les biologistes, cet échantillon présente une VAF% (Variant Allele Frequency/Fréquences Alléliques : proportion de reads indiquant la présence de la fusion) de 0,22, la plus basse parmi tous les échantillons. Cette constatation révèle une qualité de séquençage non optimale pour cet échantillon et ne remet donc pas en cause l'efficacité de Lumpy-sv.

En ce qui concerne la présentation des résultats, Lumpy-sv se concentre davantage sur les variants structuraux (insertion, délétion, inversion, etc.) que sur les fusions spécifiquement. Il offre donc des informations complémentaires qui peuvent s'avérer pertinentes pour les biologistes (voir Figure 16). Les noms des fusions ne sont pas directement inclus ; pour les obtenir, l'utilisation d'outils d'annotation de fichiers VCF tels que Snpeff est nécessaire car le fichier de sortie de Lumpy-sv est au format VCF.

4.2 Tests options/données d'entrée dans les lignes de commande

Il est aussi essentiel de réaliser des tests exhaustifs avec les différentes options de chaque outil ainsi que sur les fichiers qu'ils requièrent en entrée. Par exemple, la création de fichiers BAM à partir de star.Chimeric.out.sam pourrait être envisagée pour évaluer si cette approche génère des résultats plus précis. Cette démarche est d'autant plus pertinente pour l'outil Breakdancer, étant donné qu'il n'a généré aucun résultat positif. Si aucune amélioration significative n'est observée, il pourrait être envisagé de le substituer avec l'un des autres outils en question.

4.3 Optimisation du script de parsing

En ce qui concerne le script de parsing des résultats, il serait intéressant d'intégrer des informations relatives au gène dans le tableau de résultats. Par exemple, déterminer si le gène est déjà répertorié dans les bases de données publiques et, le cas échéant, quel impact il pourrait avoir sur le phénotype de l'individu. Une telle inclusion faciliterait grandement la recherche pour les biologistes. Il serait également avisé d'ajouter une fonctionnalité pour générer des graphiques qui illustrent les données fournies par chaque outil. Cette démarche contribuerait grandement à élargir les perspectives d'analyse des résultats.

5. Conclusion

Les outils fusionfusion et SplitFusion ont démontré un potentiel solide. En effet, ces outils ont obtenu des scores satisfaisants et sont en concurrence directe avec les outils intégrés, tels que Lumpy-sv et Factera. En ce qui concerne Breakdancer, son intégration doit être remise en question en raison de ses résultats insatisfaisants.

La prochaine étape sera de voir si l'intégration de fusionfusion et SplitFusion apportera une réelle valeur ajoutée à la plateforme. Pour cela, des tests du pipeline de gènes de fusion avec données ADN doivent être effectués en ajoutant ces deux outils à la liste des outils existants. L'objectif est de vérifier si les résultats restent cohérents avec ou sans leur intégration et si leur ajout est véritablement nécessaire. À titre d'exemple, les outils fusionfusion et Factera InterGene/InterExon ont obtenu le même score.

En termes de facilité d’utilisation, fusionfusion nécessite deux étapes préparatoires, tandis que Factera requiert trois étapes. Par ailleurs, leur présentation des résultats est similaire. Des comparaisons doivent donc être effectuées au niveau des résultats, tels que l'identification de l'outil ayant détecté le plus grand nombre de fusions. Par conséquent, il serait avisé de tester ces outils sur des échantillons nécessitant la recherche de plusieurs fusions, afin de réaliser une comparaison approfondie.

6. Bibliographies

6.1 Figures

Figure 1 : Plateforme bio-informatique MOABI (AP-HP)

Figure 2 : Plateforme bio-informatique MOABI (AP-HP)

Figure 3 : Kenzui Taniue, Nobuyoshi Akimitsu (2021). Fusion Genes and RNAs in Cancer Development, doi: 10.3390/ncrna7010010.

Figure 4 : Plateforme bio-informatique MOABI (AP-HP)

Figure 5 : https://github.com/Genomon-Project/fusionfusion

Figure 6 : https://github.com/Zheng-NGS-Lab/SplitFusion

Figure 7 : http://www.genouest.org/godocker/

Figure 12 : https://github.com/Genomon-Project/fusionfusion

Figure 13 : https://github.com/Zheng-NGS-Lab/SplitFusion

Figure 14 : https://factera.stanford.edu/

Figure 15 : https://github.com/genome/breakdancer

Figure 16 : https://github.com/arq5x/lumpy-sv

6.2 Outils

Fusionfusion : https://github.com/Genomon-Project/fusionfusion

- Star : https://github.com/alexdobin/STAR

SplitFusion : https://github.com/Zheng-NGS-Lab/SplitFusion

- Samtools : http://www.htslib.org/
- BWA : https://bio-bwa.sourceforge.net/
- Bedtools : https://bedtools.readthedocs.io/en/latest/
- Annovar : https://annovar.openbioinformatics.org/en/latest/
- Snpeff : https://pcingola.github.io/SnpEff/

Factera : https://factera.stanford.edu/

Breakdancer : https://github.com/genome/breakdancer

Lumpy-sv : https://github.com/arq5x/lumpy-sv

6.3 Interfaces

Galaxy : https://usegalaxy.org/

GitLab : https://docs.gitlab.com/

GitHub : https://docs.github.com/fr

Go-Docker : http://www.genouest.org/godocker/

6.4 Articles

Taniue, K., & Akimitsu, N. (2021). Fusion Genes and RNAs in Cancer Development. *Non-Coding RNA*, *7*(1), 10. doi : 10.3390/ncrna7010010

Wang, Q., Xia, J., Jia, P., Pao, W., & Zhao, Z. (2012). Application of next generation sequencing to human gene fusion detection: computational tools, features and perspectives. *Briefings in Bioinformatics*, *14*(5), 506-519. doi : 10.1093/bib/bbs044

Gricourt, G., Tran Quang, V., Cayuela, J., Boudali, E., Tarfi, S., Barathon, Q., Daveau, Romain., Joy, C., Wagner-Ballon, O., Bories, D., Pautas, C., Maury, S., Rea, D., Roy, L., & Sloma, I. (2022). Fusion Gene Detection and Quantification by Asymmetric Capture Sequencing (aCAP-Seq). *The Journal of Molecular Diagnostics*, *24*(11), 1113-1127. doi : 10.1016/j.jmoldx.2022.07.004

Zhang, B., Song, Z., Bao, C. Y., Xu, C., Wang, W., Chu, H. Y., Lu, C., Wang, H., Bao, S., Gong, Z., Keung, HY., Chow, M., Zhang, Y., Cheuk, W., Yang, M., Cho, W., Chen, J., Zheng, Z. (2020). Ultrasensitive gene fusion detection reveals fusion variant associated tumor heterogeneity. doi : 10.21203/rs.3.rs-39138/v1

7.Annexes

Annexe 1 : Lien GitHub du script d'automatisation

https://github.com/geap1999/FusionGenesProjectGuillaume-MOABI/blob/main/Batch5tools.sh

Annexe 2 : Lien GitHub du script de parsing de résultats

https://github.com/geap1999/FusionGenesProjectGuillaume-MOABI/blob/main/ParsingScript.py

Annexe 3 : Lien GitHub d'un fichier Excel obtenu par le script de parsing

https://github.com/geap1999/FusionGenesProjectGuillaume-MOABI/blob/main/Output_example_ARA-8782_fusion_genes_data.xlsx