Xavier Grand
Master SNS parcours BCD

10 Janvier 2020

Projet SAMY

I. Introduction

Le séquençage « short-read » est l'une des deux principales techniques de séquençage les plus utilisées. Elle à pour but de séquencer un grand nombre de fragments d'acides nucléiques de petite taille (100 à 300 nucléotides). Les séquences de ces fragments, une fois obtenues, sont nommées des « reads ». Les rusultats d'un séquençage sont, de manière standardisée stockés dans un fichier au format FASTQ. Il s'agit d'un fichier de séquences brutes associées à des informations de qualité de séquençage. Dans notre cas, on suppose qu'il s'agisse du séquençage d'un organisme dont il existe déjà des informations de séquence dite de référence. Dans le processus dit classique de traitement des données de séquençage, plusieurs étapes sont réalisées à la suite d'un séquençage.

La première étape consiste généralement à contrôler la qualité du séquençage. Cela à pour but de s'assurer de la fiabilité des séquences obtenues, c'est-à-dire qu'il n'y a pas une trop grande différence de séquences par rapport à la séquence de l'échantillon étudié. Bien-sûr, la séquence de l'échantillon étudié n'est pas forcément connue à l'avance, toutefois, les appareils de séquençage, ou séquenceurs, attribuent une mesure de la qualité des séquences qu'ils produisent/lisent. Il existe divers logiciels permettant d'interroger, interpréter, filtrer les séquences d'un fichier FASTQ sur leur qualité, nous pouvons citer FASTQC à titre d'exemple, que nous avons pu utiliser dans le cadre d'un autre Module du Master.

La deuxième étape consiste à aligner les séquences filtrées sur un génome de référence. A l'inverse d'un séquençage de novo, c'est-à-dire lorsqu'il n'existe pas d'information de séquence pour l'organisme étudié, les séquences sont comparées aux informations connues de l'organisme étudié. Les raisons de cette comparaison seront détaillées dans la partie III de ce rapport. L'alignement des séquences obtenues, ou mapping, peut être réalisé grâce à différents logiciels, nous citerons BWA (http://bio-bwa.sourceforge.net/) et Bowtie (http://bowtie-bio.sourceforge.net/index.shtml) en exemple. Les résultats d'alignement des séquences sont stockés de manière standardisées dans un fichier au format SAM qui sera détaillé dans la partie II de ce rapport.

Le travail réalisé, pour ce projet, est la production d'un programme en langage python qui affiche un résumé des informations du mapping des reads issues du séquençage d'ADN sur une séquence de référence.

II. Format de fichier SAM

1. Définition :

Le format de fichier SAM, pour « Sequence Aligement/mapping », est un format de fichier standardisé issu de l'alignement de reads, ou lecture de séquence nucléotidique, provenant de séquençage NGS, pour « Next Generation Sequencing ».

2. Description :

Il s'agit d'un fichier plat tabulé. Il contient un nombre d'informations minimal telles que le logiciel utilisé pour le mapping ou encore la séquence de référence utilisée (Figure 1).

```
@SQ     SN:Reference    LN:1000000
@PG     ID:bwa  PN:bwa  VN:0.7.9a-r786  CL:../BWA/bwa-0.7.9a/bwa mem ../reference.fasta ../fastqDir/Clone1_1.fastq ../fastqDir/Clone1_2.fastq
Clone1-350000   99      Reference       658711  27      100M    =       658812  201     CAC...TGT       CCC...GFG       NM:i:0  MD:Z:100
Clone1-350000   147     Reference       658812  27      100M    =       658711  -201    CAC...ACC       GGF...AC@       NM:i:0  MD:Z:100
```

Figure 1 : Exemple des premières lignes d'un fichier SAM.

Les premières lignes de ce fichier sont nommées l'entête, ou header. Elles sont systématiquement démarrées par un arobase @. Elles contiennent des informations préfixées telles que (de manière non exhaustive) :

- SQ : indique que la ligne de header traite des informations sur la séquence de référence, elle contient généralement les informations préfixées suivantes :
  - SN : nom de la séquence de référence,
  - LN : taille de la séquence de référence,
- PG : indique que la ligne traite des information relatives au programme d'alignement utilisé pour le mapping des reads, elle peut contenir les informations préfixées suivantes :
  - ID : nom du programme de mapping,
  - PN : nom de la commande d'alignement du programme de mapping,
  - VN : Version du programme de mapping,
  - CL : commande lancée pour réaliser le mapping.

Col	Field	Type	Regexp/Range	Brief description
1	QNAME	String	`[!-?A-~]{1,254}`	Query template NAME
2	FLAG	Int	$[0, 2^{16} - 1]$	bitwise FLAG
3	RNAME	String	`\*|[:rname:^*=][:rname:]*`	Reference sequence NAME[10]
4	POS	Int	$[0, 2^{31} - 1]$	1-based leftmost mapping POSition
5	MAPQ	Int	$[0, 2^{8} - 1]$	MAPping Quality
6	CIGAR	String	`\*|([0-9]+[MIDNSHPX=])+`	CIGAR string
7	RNEXT	String	`\*|=|[:rname:^*=][:rname:]*`	Reference name of the mate/next read
8	PNEXT	Int	$[0, 2^{31} - 1]$	Position of the mate/next read
9	TLEN	Int	$[-2^{31} + 1, 2^{31} - 1]$	observed Template LENgth
10	SEQ	String	`\*|[A-Za-z=.]+`	segment SEQuence
11	QUAL	String	`[!-~]+`	ASCII of Phred-scaled base QUALity+33

Table 1 : Les différents champs dans le fichier au format SAM.

Les lignes suivantes, celles qui ne débutent pas par un arobase, sont les lignes du corps du fichier SAM. Elles contiennent l'information d'alignement pour chaque read. Cette information est répartie sur un nombre minimal de 12 champs (Table 1, d'après Li, Heng et al. "The Sequence Alignment/Map format and SAMtools." Bioinformatics (Oxford, England) vol. 25,16 (2009): 2078-9. doi:10.1093/bioinformatics/btp352), mais sans maximum défini. Les 12 (11 obligatoires) champs sont ordonnés de la façon suivante :

- QNAME : Identifiant du read. L'identifiant du read est unique, toutefois, il peut être dupliqué dans le fichier SAM dans le cas d'un séquençage en « paired-end ». C'est-à-dire, lorsque chaque read est séquencé depuis les deux extrémités. Ainsi, dans ce cas, la s équence attendue pour un identifiant de read est unique mais elle peut être double dans un fichier SAM.
- FLAG : correspond à une information binaire relative au mapping (). Il s'agit d'un nombre unique qui résume les informations suivantes :
  - Read en paired-end ou en single-end,
  - Read in a proper pair
  - Read mappé ou non mappé,
  - L'autre read de la paire est mappé ou non mappé,
  - Read mappé sur le brin « Forward » ou « Reverse »,
  - L'autre read de la paire est mappé sur le brin « Forward » ou « Reverse »,
  - Premier read de la paire,
  - Second read de la paire,
  - Le premier alignement du read sur la référence,
  - Le read n'a pas passé le contrôle de qualité, ce cas de figure n'est pas rencontré lorsqu'une première étape de filtration est réalisée dans le processus d'analyse de résultats de séquençage tel que mentionné en introduction.
  - Duplicat technique de PCR.

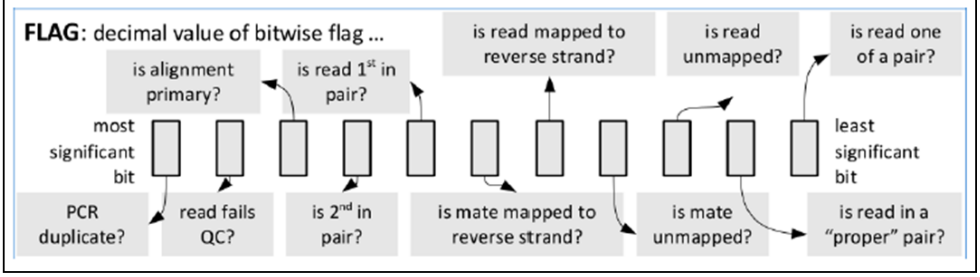

Figure 2 : Décomposition du FLAG bit à bit.
Alban LERMINE, Olivier INIZAN, Ecole bioinformatique Roscoff, 18 novembre 2013

- RNAM : Séquence sur laquelle est mappée le read. Si le read n'est pas mappé, le caractère de remplacement est un astérisque '*'.
- POS : Position la plus 5' du read.

- MAPQ : Qualité du mapping, soit −10log10(p), où p est la probabilité, estimée par l'outil de mapping, que le read soit mal mappé.
- CIGAR : résumé d'informations relatives à l'alignement du read tel que les insertions/délétions par exemple (Figure 2, de Alban LERMINE, Olivier INIZAN, Ecole bioinformatique Roscoff, 18 novembre 2013).
- RNEXT : Caractère qui indique la séquence sur laquelle est mappée l'autre read de la paire, en cas de paired-end. Le caractère = est utilisé si le chromosome est le même, et le caractère * si le read est single-end.
- PNEXT : En cas de paired-end, position la plus 5' de l'autre read de la paire; on utilise 0 en single-end ou si l'autre l'autre read de la paire ne mappe pas, etc.
- TLEN : En cas de paired-end, taille de la lecture, soit la différence entre la position la plus 5' du read 5' et la position la plus 3' du read 3', c'est-à-dire l'autre read de la paire.
- SEQ : La séquence du read.
- QUAL : La qualité du fragment telle que renseignée dans le fichier FASTQ issu du séquenceur ; le caractère * est utilisé si l'information est indisponible.
- 12ème champs : Informations supplémentaires relative à l'alignement, ce champs contient des informations préfixées telles que le « MD:Z: » qui résumé l'information de la variation de séquence par rapport à la séquence de référence, les substitutions par exemple. Les informations de ce champ sont facultatives, l'ordre n'est pas standardisé. Seuls les préfixes permettent de définir l'information stockée.

Tous les reads mappés sont représenté par rapport au brin forward de la référence. Les reads mappés sur le brin forward sont représentés par leur séquence complémentaire.

III. Exemple d'application (motivation biologique)

Dans cette partie, nous traiterons de l'intérêt, bien qu'évident, du séquençage de nouvelle génération en short-read, à travers trois exemples. Un premier exemple est l'assemblage de génome complet d'un individu, dans le cas où il existe un génome de référence pour cette espèce. Un second exemple relatif à la détection de variation génétique, nous nous intéresserons aux SNP, pour « Single Nucleotid Polymorphism », et son utilisation en amélioration des plantes. Et enfin un dernier exemple traitera de l'expression de gènes en génome complet par séquençage. Il s'agit, pour ce dernier, d'un séquençage d'ADNc, dit complémentaire, qui est en réalité la rétrotranscription des ARN messagers ; ainsi, il s'agit bien de séquençage ADN bien que l'origine des acide nucléiques soit de l'ARN.

1. Assemblage de génome

Le séquençage ADN, en short-read, peut être utilisé afin de séquencer un individu d'une espèce dont un génome de référence est disponible. Il est également possible de séquencer en short-read un individu d'une espèce dont il n'existe pas de génome de référence, or ce cas de figure est à la marge du travail effectué lors de ce projet, aussi nous n'en discuterons pas.

Si les approches short-read ne permettent pas toujours de réaliser un assemblage complet d'un génome en raison de la faible longueur des reads, elles n'en sont pas pour autant un outil inutile, bien au contraire. En effet, elles permettent d'obtenir une profondeur de séquençage importante, rapidement, à coût modéré. Aussi, plus la profondeur de séquençage est importante, plus les erreurs techniques de séquences sont « diluées » dans des séquences sans erreur. Ainsi, le mapping de résultats de séquençage ADN short-read est un outil de choix pour l'amélioration de séquence de génome complet.

Par conséquent, s'il est possible d'identifier des erreurs de séquençage, il est aussi possible d'identifier du polymorphisme entre les séquences de différents individus. Ou plus simplement, le polymorphisme d'un individu par rapport à un génome de référence. Il existe de nombreuses applications à l'identification de polymorphismes génétiques, nous en discuterons à propos d'un exemple dans le paragraphe suivant.

2. Détection de SNP et amélioration des plantes

Le secteur de l'amélioration des plantes, emblématique de la recherche effectuée par les semenciers, utilise les nouvelles technologies de séquençage ADN afin d'identifier du polymorphisme dans les différentes variétés d'espèces à intérêt agronomique. En effet, le polymorphisme génétique, entre autres en amélioration des plantes, est utilisé afin de cartographier le génome des individus/variétés/cultivars/écotypes d'une espèce.

Une approche très en vogue dans cette industrie aujourd'hui est la sélection génomique. Elle a pour but d'associer, sur un panel de diversité de variétés relativement réduit (une à quelques centaines), un ou plusieurs caractères phénotypiques à un ensembles d'allèles représentés par des marqueurs génétiques. Grâce à cette association, il est possible de réaliser une prédiction statistique de la part d'un allèle à un marqueur dans un caractère. Ainsi, le sélectionneur pourra à loisir associer dans son processus de sélection les allèles aux marqueurs qui lui permettront de choisir les caractères de la plante qu'il désire obtenir.

Les marqueurs les plus « fins » aujourd'hui, c'est-à-dire les marqueurs permettant d'avoir la plus fine définition pour une carte génétique, sont les SNPs. De ce fait, le séquençage short-read dans ce cas de figure est le plus adapté. En effet, les espèces à fort intérêt agronomique telles que le maïs ou le riz sont des espèces séquencées depuis de nombreuses années (< 2010), elles ont donc un génome de référence de bonne qualité. Il est donc possible, par le séquençage short-read, de

séquencer un panel de diversité. Ainsi, pour chaque lignée du panel, l'alignement des reads permettra d'identifier les SNP par rapport à la séquence de référence. Chaque lignée du panel est phénotypées pour les caractères d'intérêt, il est alors possible d'associer génotype (ensemble des allèles aux marqueurs SNP) et phénotype.

Le coût réduit du séquençage et la capacité à réaliser un alignement de qualité des reads sont un outil d'une grande valeur pour l'industrie semencière.

3. Expression de gène, transcriptomique

Une application du séquençage short-read, et du mapping de ces reads, est la mesure de l'expression des gènes. En effet, de la même façon que lorsque l'on souhaite mesurer l'expression de gènes par Northern-Blot, RT-qPCR ou encore par puce à ADN, il est nécessaire de réaliser une étape de rétrotranscription des ARN messagers extrait du tissu vivant dans lequel on souhaite mesurer l'expression des gènes. Ainsi, des ANDc, dit complémentaires, sont obtenus. Il est alors possible de séquencer ces ADNc.

De plus, le séquençage short-read est quantitatif. Ainsi, il est possible de mapper les reads issus du séquençage des ADNc et de quantifier le nombre de reads qui mappe sur une même position d'un génome de référence. Lorsque cette position correspond à un gène exprimé, il est possible d'extrapoler l'accumulation des ARN messagers, c'est-à-dire de l'expression du gène transcrit.

IV. Projet/script

L'objectif de ce projet est de développer un script python qui permet de lire et de « parser » (répertorier, organiser, indexer, rendre accessible) un fichier de résultat de séquençage short-read au format SAM. Puis d'en interpréter les informations afin de quantifier les reads qui sont correctement mappés, ceux qui ne sont pas mappés, ceux qui sont partiellement mappés, et les paires de reads (lorsque la situation le permet, séquençage « paired-ends ») dont l'un est correctement mappé et l'autre n'est pas mappé, ou bien, les paires de reads dont l'un est mappé et l'autre est partiellement mappé.

Le script python a été développé sur la base de plusieurs fonctions stockées dans différents scripts python dans un répertoire « lib », pour librairie ; et d'un script « main » nommé Samy.py. Le projet a donc été réalisé sous la forme d'un programme nommé Samy. Un fichier Read-me.md est associé à Samy, il est disponible au format MarkDown.

Il est possible de décomposer le programme en plusieurs étapes (Figure 3) :

- La lecture du fichier SAM ainsi que la prise en charge de paramètres.
- Le parsing du fichier SAM.
- L'interprétation des données.

- L'affichage des statistiques relatives au mapping et la filtration des reads en un ou plusieurs nouveaux fichiers SAM.

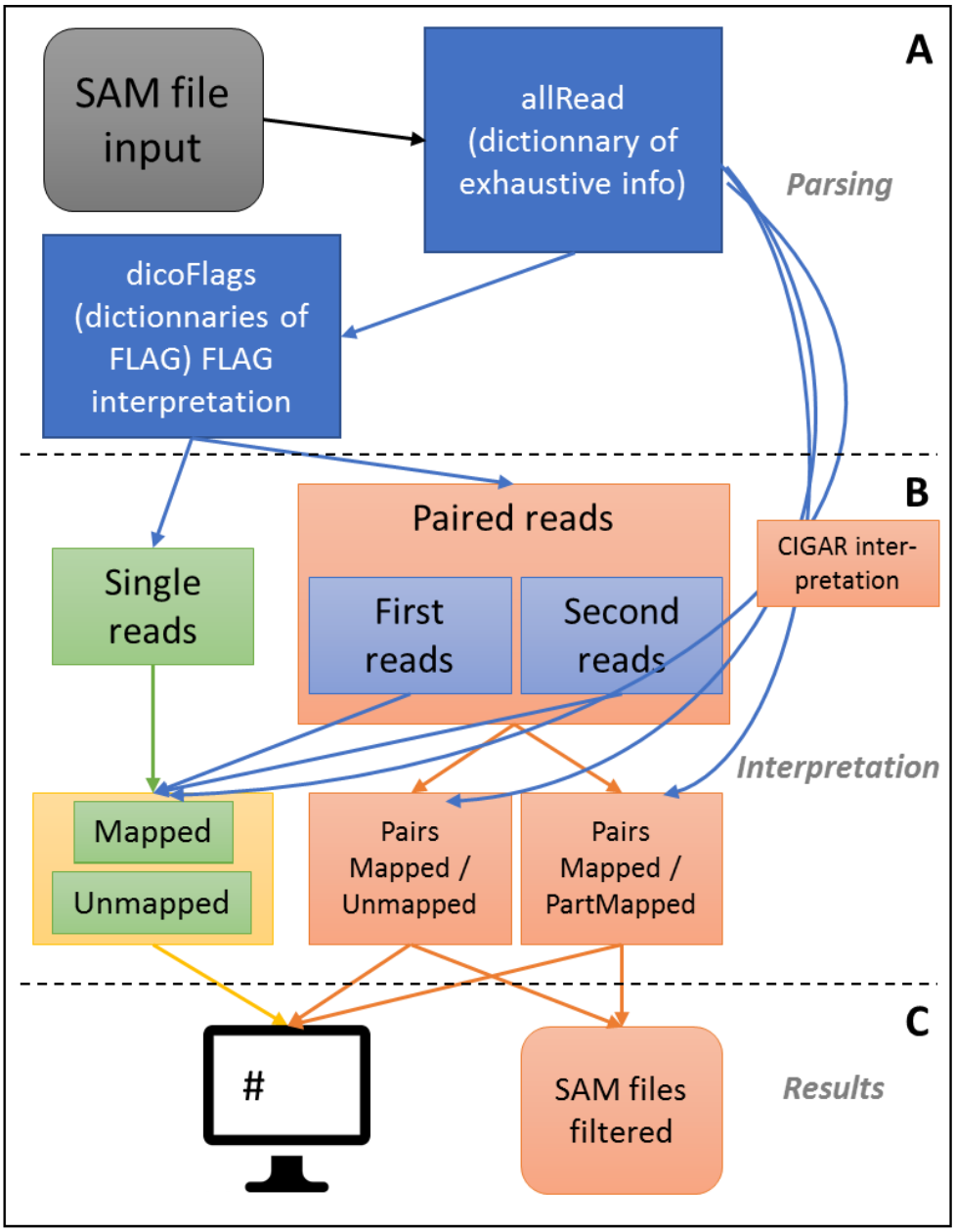

Figure 3 : Schema fonctionnel de Samy :
A : Le fichier d'entrée SAM est parsé dans un dictionnaire nommé allRead par la function saveAllData. La function dicoFlag crée 3 dictionnaires de reads en function de leur catégorie ("Single", "First in pair", "Second in pair").
B : L'interprétation des informations de FLAG et CIGAR permet la creation de listes de reads en function de leur statut d'alignement.
C : Les résultats sont affichés à l'écran et stockés à la demande dans de nouveaux fichiers SAM.

1. Lecture des fichiers SAM

Le fichier d'entré est indiqué en paramètre, pour cela, la librairie *argparse* est utilisée. Cette librairie prend en charge les différents paramètres du programme. Elle permet également de générer l'aide et les *usage*. Pour plus de détails, se reporter à l'aide (Samy. Py --help/-h) ou au fichier Read-me.md.

Le fichier est ouvert et traité ligne à ligne. Les lignes correspondant aux entêtes, c'est-à-dire qui commencent par un arobase, sont traitées pour récupérer les informations relatives à la séquence de référence et au programme utilisé pour le mapping. Puis, les lignes du corps sont traitées par la fonction *saveAllData* importée depuis la librairie *lib* qui parse les informations pour chaque read de manière exhaustive.

2. Parsing de l'information

- *Parsing exhaustif :*

Dans un premier temps, la fonction *saveAllData* crée un dictionnaire nommé *allRead* dont les clefs sont les identifiants des reads « QNAME ». Ce dictionnaire contient des dictionnaires dont les clefs sont les identifiants des 11 champs du fichier SAM, ainsi que d'autres clefs créées lors de ce parsing, il s'agit de la clef « BINFLAG », et de la clef « MD:Z: ». A chaque clef est associée une liste des valeurs pour chaque champ pour un QNAME donné.

Dans la version actuelle du programme, pour les besoins du projet et par gain en temps de calcul, mais surtout de ressource mémoire, seuls les champs FLAG, CIGAR, et les nouveaux champs BINFLAG et MD:Z: sont créés. Les informations relatives à la clef BINFLAG sont des chaines de 11 caractères qui représentent une transformation des valeurs du champ FLAG en binaire. La fonction *bin* de la librairie *lib* permet cette transformation. Les informations relatives au champ MD:Z: sont l'extraction de la chaine de caractère préfixée par « MD:Z: » (voir II.2.).

```
# Structure de allread:
# allread [
        'Clone1-350000' {
        'FLAG': ['99', '147'],
        'BINFLAG': ['000011000011', '000100100011'],
        'CIGAR': ['100M', '100M'],
        'MD:Z:': ['100', '100']
        }
]
```

Lors du parsing, des « irrégularités » que l'on peut rencontrer lorsqu'un fichier SAM est corrompu sont prises en compte. En effet, si la première ligne du fichier chargé ne commence pas par un arobase, c'est-à-dire qu'il ne s'agit pas d'une ligne d'entête, le programme affiche un avertissement et met le programme en pause en attente de l'intervention de l'utilisateur :

ATTENTION, pas d'entête à ce fichier. Veuillez vérifier que le fichier n'est pas corrompu.

…

Tappez ENTER pour continuer…

De plus, si les lignes qui ne commencent pas par un arobase, c'est-à-dire les lignes du CORPS du fichier SAM ne contiennent pas un minimum de 11 champs, un avertissement est affiché et le programme est mis en pause en attente de l'intervention de l'utilisateur :

ATTENTION, le nombre de champs n'est pas conforme. Veuillez vérifier que le fichier n'est pas corrompu.

…

Tappez ENTER pour continuer…

- *Parsing des dictionnaires de FLAG*

Afin de prendre en compte la possibilité de travailler sur du séquençage Single-end et Paired-ends, une information stockée dans le champs complexe FLAG, il a été choisi de créer des dictionnaires qui servent à l'interprétation des FLAG. Trois dictionnaires sont créés, un dictionnaire qui contient tous les read en single-end, et deux autres dictionnaires « symétriques », *First* et *Second*, qui contiennent les reads « first in pair » et « second in pair ». Chaque dictionnaire est rempli en fonction du FLAG de chaque read.

```
# Structure des dictionnaires de FLAG :
# dicoFirst [
        'Clone1-350000' {
        'paired': 1, 'proper': 1, 'unmap': 0, 'mateUnMap': 0, 'strand': 0,
        'mateStrand': 1, 'first': 1, 'second': 0, 'alignment1': 0, 'QC': 0,
        'duplicate': 0
        }
]
# dicoSecond [
        'Clone1-350000' {
        'paired': 1, 'proper': 1, 'unmap': 0, 'mateUnMap': 0, 'strand': 1,
         'mateStrand': 0, 'first': 0, 'second': 1, 'alignment1': 0, 'QC': 0,
        'duplicate': 0
        }
]
```

C'est grâce aux informations de FLAG stockés dans les dictionnaires de FLAG et les informations de CIGAR que les comptages, ou « statistiques » de mapping sont calculées.

3. Comptages des reads en fonction de leur alignement sur la référence :

- *Comptage sur la base des FLAG :*

En parcourant les dictionnaires de FLAG, des informations sont directement accessibles telles que :

- Le nombre total de reads = somme de la taille des trois dictionnaires.
- Nombre total de paires de reads = somme des occurrences 1 de la clef « paired » dans le dictionnaire dicoFirst (ou dicoSecond).
- Nombre de read en Single end = somme des occurrences 0 de la clef « paired » dans les dictionnaires dicoFlag.
- Nombre de reads mappés = somme des occurrences 0 de la clef « umap » dans les dictionnaires dicoFlag.
- Nombre de reads non-mappés = somme des occurrences 1 de la clef « umap » dans les dictionnaires dicoFlag.

Ces valeurs sont calculées et stockées pour affichage.

- *Comptage sur la base du CIGAR :*

4. Représentation/présentation des données

De manière très simple, via le terminal, le programme affiche les informations relatives à la séquence de référence ainsi que des « statistiques » relatives aux résultats de mapping (Figure 4). Une fonctionnalité d'option passées en paramètres permet de choisir les informations à afficher ou non.

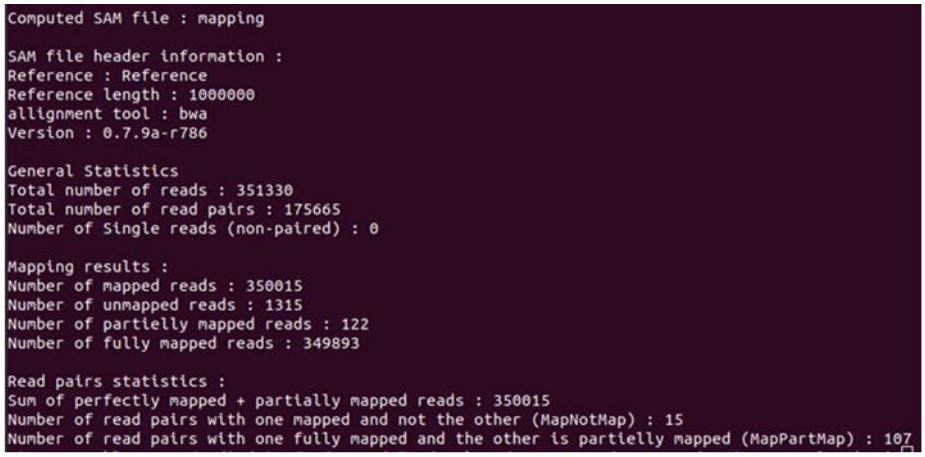

Figure 4 : Exemple d'affichage des résultats de mapping par Samy.

De plus, une fonctionnalité optionnelle est disponible via des paramètres qui permet de filtrer le fichier SAM d'entrée en de nouveaux fichiers SAM en fonction des résultats de mapping.

V. Discussion sur les aspects positifs et négatifs du script

1. Ce que fait le script

Le programme affiche des informations en fonction des paramètres/options renseignés dans la ligne de commande :

- L'utilisateur peut choisir d'afficher ou non les informations relatives à la séquence de référence telles que sont ID et sa longueur, et les informations relatives au programme de mapping utilisé telles que le nom du programme et sa version.
- L'utilisateur peut choisir d'afficher des « statistiques » de mapping sommaires, ou plus complète.

Le programme filtre le fichier SAM en de nouveaux fichiers SAM en fonction des résultats de mapping. Dans sa version actuelle, il permet de sauver :

- Les reads qui font partis des paires MapNotMap.
- Les reads qui font partis des paires MapPartMap.

2. Ce qu'il ne fait pas/Pistes d'améliorations

- La prise en charge de la nomenclature des chemins vers les fichiers d'entrées est laissée sous la responsabilité à l'utilisateur.
- La librairie lib n'est pas a proprement parlé une librairie python. Il ne s'agit pour le moment que d'un container (un dossier) qui contient les différents scripts pour chaque fonction. Faire une librairie permettrait d'importer toutes les fonctions avec un seul appel.
- Le parsing de l'information du fichier SAM n'est pas complet dans la version actuelle. Seuls les champs utiles aux besoins du projet sont remplis. Toutefois, il ne manquerait pas grand-chose pour tout remplir en créant un canevas de dictionnaire vide et en le remplissant d'un coup avec toutes les informations de la ligne du fichier SAM.
- La fonction *bin* est inutile, en informatique, TOUTE valeur est stockée en binaire, il faut travailler sur le typage des valeurs plus que sur des chaines de caractères.
- La solution des dictionnaires de FLAG n'est pas du tout optimale, il vaudrait mieux joindre toute l'information dans un seul dictionnaire comme le dictionnaire *allRead*. La manipulation des arguments en paramètre, grandement facilitée par la librairie *argparse*, devrait permettre d'adapter en fonction des besoins de l'utilisateur les informations à parser.
- La filtration du fichier SAM pourrait prendre en compte toutes les combinaisons de reads possibles. Une filtration uniquement sur les reads mappés, ou sur les reads non mappés, ou partiellement mappés, ou encore sur les paires de reads dont les deux reads sont parfaitement mappés.
- Les résultats de comptage pourraient être sauvés dans un fichier.