Cytometry-Ontology Framework

Adrin Jalali, Mélanie Courtot,
Raphael Gottardo, Richard Scheuermann, Ryan Brinkman

Abstract

In this document we propose a mechanism to link the output from any flow cytometry based gating or clustering (*i.e.*, manual or automated analysis) to the Cell Ontology (CL) in order to provide semantic names to the identified cell populations. This requires that the analysis phase ends by labelling the identified cell populations with immunophenotypic characteristics.

1 Problem Definition

Finding one or more cell populations of interest is a core part of cytometry data analysis (*e.g.*, to find a cell population correlating to a certain disease). Identifying a cell population is not well defined and it can vary from being defined as a cluster by an algorithm, to a term like "Natural Killer T-Cells". Sometimes a group of cells is referred to as an immunophenotype (*e.g.*, $CD5^+$ often identifies T-cells, but someone might instead use $CD7^+$ to identify T-cells). This situation makes it difficult to integrate the output of algorithms to external knowledge sources. The same problem occurs when the goal is to link output of different algorithms, or output of the same algorithm on different experiment when experiment markers are not the same. Hence, we propose a framework that enables researchers to report their results by ontology terms. Therefore people will be able to find previous research on the same cell population, or related to the same cell population (*i.e.*, a subset of that cell population), even if different markers are used. The core module of this framework is an *ontology labeller* that attempts to provide an semantic identifier to a cell population based on its marker expression profile

(*i.e.*, the immunophenotype), The analysis approach used to identify cell populations can be automatic or manual; the only thing that matters is the format of the output that is given to the ontology labeller.

2 Analysis

The analysis is the prior phase to the ontology labeller in our framework. This phase can be manual (*e.g.*, manual gating), or automatic (*e.g.*, flowType, FLOCK, flowClust). Considering that the ontology labeller input is an immunophenotype characterized by markers, the output of the analysis phase must be in this format Figure 1. In addition to presence and absence, the strategy used to convert fluorescence intensity distributions into immunophenotypic qualities will also have to deal with "low, intermediate, and high" expression attributes.

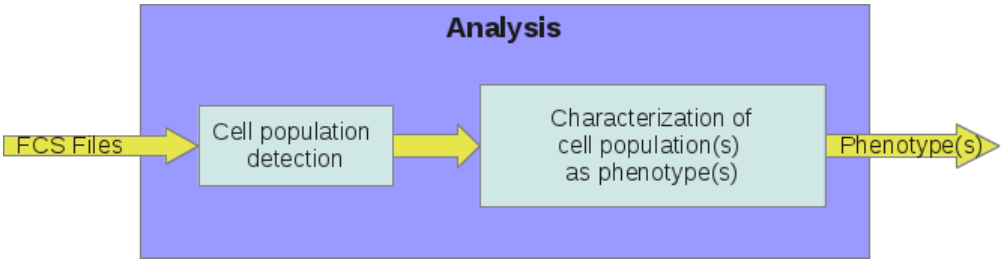

Figure 1: Analysis phase

The cell population characterization as immunophenotypes might be embedded into the cell population detection phase (*e.g.*, FLOCK, flowType), or might be a separate module (*e.g.*, there is a need to design a separate module to be able to use flowClust as cell population detector in this framework). A duty of this module is to have its output in the immunophenotype format that is required to be able to give it to the ontology labeller. As depicted in Figure 1, we refer to both (1) cell population detection and (2) characterization of cell population(s) by immunophenotype(s), together, as the analysis phase.

3 Ontology Labeler

The ontology structure is represented as a graph, in which, nodes represent immunophenotypes and we only choose edges that represent the "is a" relationship. This structure does not have any loops and therefore is a directed acyclic graph (DAG). The ontology labeller's responsibility is to find a small portion of this DAG (sub-DAG) as the corresponding ontology of a given immunophenotype. An example of a part of the ontology structure is illustrated as in Figure 2.

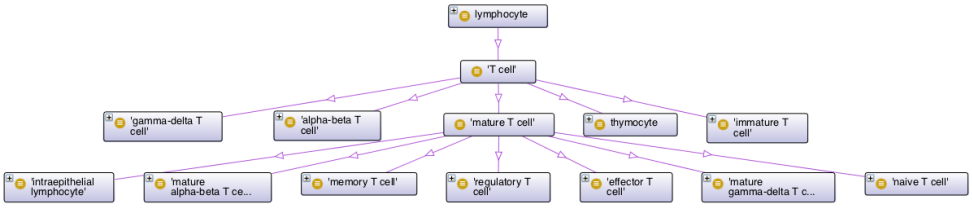

Figure 2: Analysis phase

This core module thus gets a cell population of interest as an immunophenotype characterized by markers (*e.g.*, $CD4^{+}CD45RO^{+}$) and returns a sub-DAG of the ontology structure with the root *Memory T Cells*. In general, the output of the ontology labeller is a set of sub-DAGs from the ontology DAG. The returned result can be (1) an empty set; (2) a single node; (3) a single connected sub-DAG; (4) or a set of sub-DAGs. Giving more specific immunophenotypes to the module results in a pruned result and in the best case, a small DAG or a single node of the ontology structure representing the cell population of interest.

4 Programmatic linking to the CL

The goal of programmatically linking the CL (specifically hematopoeitic cell types [1]) is to be able to retrieve the label of cell populations that have been identified using an immunophenotype label. We present here the list of resources and tools required in order to associate each set of immunophenotype with the label of the corresponding

ontology term.

4.1 CL

4.1.1 Remote access

The CL can be accessed remotely via the OBO Neurocommons [2] SPARQL endpoint [3] or the He group SPARQL endpoint [4]. Both resources update their bundles nightly based on the export performed by the OBO Foundry pipeline [5]. At the time of this writing, there is an issue between the Neurocommons instance and the OBO Foundry export, preventing upload of an updated version of the CL. The current version of the CL available does not include the external terms that have been imported via MIREOT [6] nor does its term have the new URIs complying with the approved ID policy [7]. Those limitations currently prevent us from relying on this installation. In short, the choice of remote source needs further investigation in light of for example potential update issues.

4.1.2 Local access

One option to not be dependent on remote resources is to download the CL.owl file directly from the CL subversion repository [8]. The platform-independent ARQ toolkit [9] can be used to query local Resource Description Framework (RDF) [10] files with SPARQL [11].

4.1.3 Content of the CL

In our preliminary tests, we faced some issues with the content of the CL. Those can be divided in three types:

1. Modelization issues: for example, some properties were asserted in the wrong hierarchy. For example, has_high_plasma_membrane_amount is not a subproperty of has_plasma_membrane_part, preventing our query to retrieve all expected results. This type of error above can be easily addressed by providing feedback to

the CL developers group, via their Sourceforge tracker[1].

2. Missing information: the CL currently imports information about biological complexes from the Gene Ontology [12] and the Protein Ontology [13], with information for their subunits in the latter. As a consequence, even though the CL contains information about the CD3 epsilon subunit as well as the T-cell receptor complex, there is no relation linking both entities. It is unclear where such assertions between a complex and it subunits should be made.

3. Scope of the CL: the CL's current aim is to identify only those markers that are necessary and sufficient to define a cell type. As a consequence, only a handful of markers are currently available, and complex cases such as a combination of 6 or 7 markers would probably not be identifiable. Due to this feature of CL, information about any additional markers would be irrelevant to the definition of a single cell type and therefore could be ignored. However, David Dougall in Richard's group did compile a list of marker expression characteristics for hematopoietic cell types that are not required for their definition and made those available in the Cell Type database component of ImmPort. This information could be made available to start address this issue, and a complete knowledge base of all markers seems like a useful parallel goal.

Options to address the issues above include close collaboration with CL developers, identification of those sets of immunophenotypes that are most common, inclusion of this list of immunophenotypes with corresponding labels as well as missing relation/information from the CL in a distinct file extending the CL. Advantages of the latter are that:

1. It would provide us more flexibility in development
2. Nothing prevents CL to include part or all of it in next versions
3. CL could even offer to release with or without FCM extension for their users
4. It should be relatively small and hopefully quite easy to build.

[1] `http://sourceforge.net/tracker/?func=browse&group_id=76834&atid=925065`

Our understanding is PRO represents the species-specific classes of protein complexes, while GO represents the species-independent classes of protein complexes. [14]. If everything is in PRO there should be no issue, however if the complex is in GO and the subunit in PRO then one of them needs to add the axiom (*i.e.*, importing the missing GO or PRO term and add the logical restriction).

With respect to CL being incomplete, a mechanism for automatically updating the CL based on the community uploaded data would help address this. There is a need to be able to forward requests to CL, but also a way for CL to respond efficiently to requests.

4.2 Use SPARQL from R

Several libraries for use of SPARQL within R are available and briefly described below.

4.2.1 OntoCat R package

OntoCat [15] is a high-level Application Programming Interface (API) for interacting with ontology resources. An R package is provided; it does however only provide access via predefined methods and does not allow for direct SPARQL querying of the file, which seems to prevent its usage for our purpose at this stage.

4.2.2 Package r-sparql

The R-SPARQL package [16] provides minimal method to query RDF via SPARQL in R. It however doesn't provide reasoning capabilities, which are required to draw correct inferences from the CL.

4.2.3 Package rrdf

The RRDF package [17] is another R package supporting the RDF. It seems better maintained that the package mentioned above (last commit end 2011 vs. mid-2010). While it doesn't provide reasoning capabilities at this stage, personal contact with

the developer, Egon Willighagen[2], indicates that there should be no issue to add this feature if required.[3]

4.3 Known limitations

The matching between the immunophenotypes and the CL will require establishing correspondence between the relation and the presence or absence of the marker. Additionally to build the SPARQL query a correspondence table between the names and URIs of the relations should be maintained. Finally, the marker themselves will be matched as string to the entities in the ontology. As a consequence, the matches will be approximate, based on regular expressions. For example, $CD4^-$ would be translated into *lacks_plasma_membrane_part CD4*.

5 Pipelines

Here we present different pipelines for various analysis methods, depending on the analysis phase output type. The simplest output of the analysis phase is one immunophenotype for each cell population. For instance, in the gate plotted in Figure 3, $Kappa^-Lambda^+$ is the output. Providing that immunophenotype to the ontology labeller, the returned result would be null set, a small portion of the ontology (provided as a DAG), or a large portion of the ontology provided as a DAG. Depending on the output, the user then iterates to possibly narrow down or generalize the returned result as illustrated in Figure 4. Using more or less markers in the analysis or changing the parameters of the analysis step are possible choices in doing the iteration.

As an aside example of some of the potential issues in automated labelling, the correct assignment of the $Kappa^-$ population immunophenotype label may be challenging in practice as this requires domain knowledge that B cell are expected to express either

[2] `http://egonw.github.com/`

[3] While at this stage a reasoner need to be run over the CL to infer correct results, this could be done in a pre-processing step. Also, it is anticipated that the CL developers provide an inferred version of their resource in the future, such as is commonly done for Foundry ontologies. In this case, we could directly rely on this rather than requiring addition of a reasoner in the R packages.

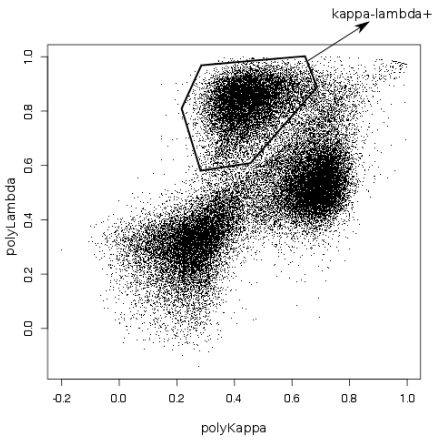

Figure 3: Sample manual gating

kappa only or lambda only and never both, and recognition that immunoglobulin subunits on the cell surface are notorious for background staining. In fact, the distribution of kappa marker fluorescence in distinctly higher than the kappa distribution of the double-negative population in the lower right.

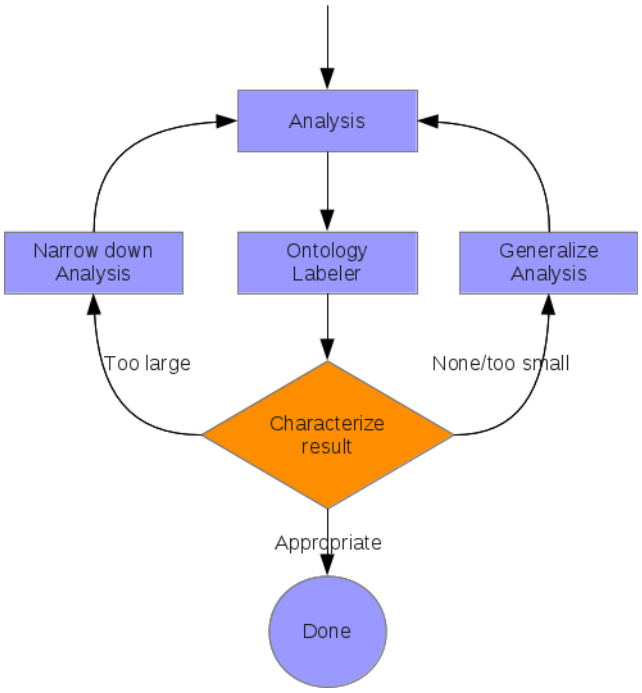

Figure 4: Single immunophenotype pipeline

Another type of the analysis output is the case of having multiple immunophe-

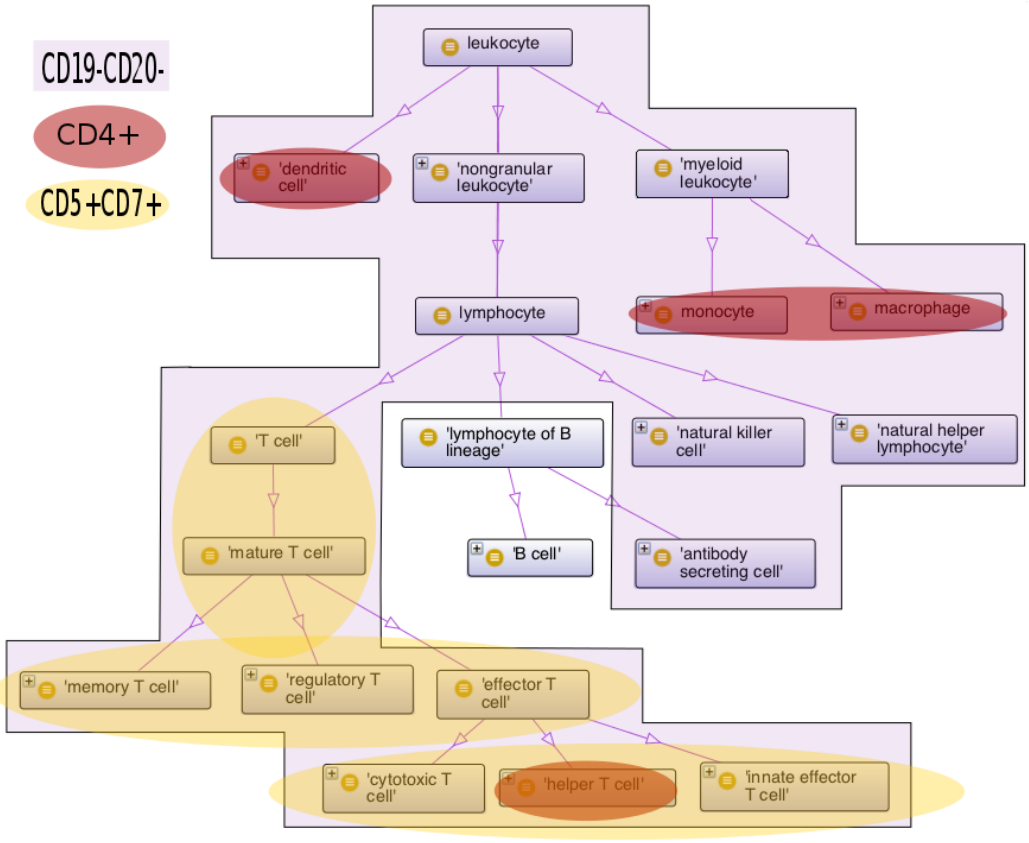

Figure 5: Immunophenotype regions in a part of the ontology structure

notypes for each cell population (*e.g.*, $CD5^+$ and $CD7^+$). In this case, the question will be to find the part of the ontology structure that represents the cell population of interest with a high degree of confidence (*i.e.*, intuitively, the part of the ontology structure that is most referred by the analysis output set of immunophenotypes). For that, the associated part of the ontology for each immunophenotype is found using ontology labeller, then for each time that a part of the ontology structure is returned, its confidence level is increased. The latter phase is the duty of *confidence level calculator* module that is placed after ontology labeller. Finally the part of the structure having the most confidence level will be reported as output (Figure 6). For example, assume that the output of the analysis phase includes these three immunophenotypes: $CD4^+, CD19^-CD20^-, CD5^+CD7^+$. $CD4^+$ refers to T helper cells,

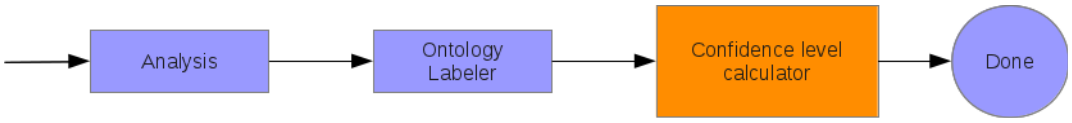

Figure 6: Multiple immunophenotypes pipeline

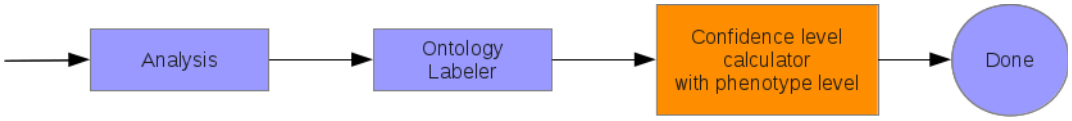

Figure 7: Parent/Child immunophenotypes pipeline

monocytes, macrophages, and dendritic cells; $CD19^{-}CD20^{-}$ refers to non B-cells; and $CD5^{+}CD7^{+}$ refers to T-cells. Therefore adding up these three immunophenotypes will result in T helper cells having more confidence value which is returned by the confidence level calculator. This is illustrated in Figure 5.

A more complicated type of analysis output is that for each cell population, a number of immunophenotypes are returned such that there is a parent/child relationship between them making a DAG structure. This parent/child relationship between immunophenotypes means that a parent immunophenotype is a more general one than its child. Using that structure we again calculate confidence level for parts of the ontology structure that are returned, like previous scenario, plus adding *level* of the immunophenotype (*i.e.*, immunophenotypes with less distance to the most general immunophenotype in the structure are in a higher level) as a parameter to the system. We make higher level immunophenotypes, which are also less specific, have less impact on the calculated confidence level of the ontology structure. This module is indicated as *confidence level calculator with immunophenotype level* in Figure 7. Then like we have done before, the returned result would be the part of the ontology having the highest confidence value.

References

[1] Alexander D. Diehl, Alison Deckhut Augustine, Judith A. Blake, Lindsay G. Cowell, Elizabeth S. Gold, Timothy A. GondreLewis, Anna Maria Masci, Terrence F. Meehan, Penelope A. Morel, Anastasia Nijnik, Bjoern Peters, Bali Pulendran, Richard H. Scheuermann, Q. Alison Yao, Martin S. Zand, and Christopher J. Mungall. Hematopoietic cell types: Prototype for a revised cell ontology. *Journal of Biomedical Informatics*, 44(1):75 – 79, 2011.

[2] Alan Ruttenberg, Jonathan A. Rees, Matthias Samwald, and M. Scott Marshall. Life sciences on the semantic web: the neurocommons and beyond. *Briefings in Bioinformatics*, 10(2):193–204, 2009.

[3] Neurocommons sparql endpoint - `http://sparql.neurocommons.org/`, Retrieved 2012.

[4] He group sparql endpoint - `http://sparql.hegroup.org/sparql`, Retrieved 2012.

[5] OORT - OBO Ontology Release Tool - `http://code.google.com/p/owltools/wiki/OortIntro`, Retrieved 2012.

[6] M. Courtot, F. Gibson, A. L. Lister, J. Malone, D. Schober, R. R. Brinkman, and A. Ruttenberg. Mireot: The minimum information to reference an external ontology term. *Applied Ontology*, 6(1):23–33, 2011.

[7] A. Ruttenberg, M. Courtot, and C. Mungall. OBO Foundry Identifier policy - `http://obofoundry.org/id-policy.shtml`, Retrieved 2012.

[8] Cell ontology OWL file - `http://purl.obolibrary.org/obo/cl.owl`, Retrieved 2012.

[9] ARQ - a SPARQL Processor for Jena - `http://incubator.apache.org/jena/documentation/query/index.html`, Retrieved 2012.

[10] Resource Description Framework (RDF) / W3C Semantic Web Activity - http://www.w3.org/RDF/.

[11] SPARQL Query Language for RDF - http://www.w3.org/TR/rdf-sparql-query/.

[12] MA Harris, J Clark, A Ireland, J Lomax, M Ashburner, R Foulger, K Eilbeck, S Lewis, B Marshall, C Mungall, J Richter, GM Rubin, JA Blake, C Bult, M Dolan, H Drabkin, JT Eppig, DP Hill, L Ni, M Ringwald, R Balakrishnan, JM Cherry, KR Christie, MC Costanzo, SS Dwight, S Engel, DG Fisk, JE Hirschman, EL Hong, RS Nash, A Sethuraman, CL Theesfeld, D Botstein, K Dolinski, B Feierbach, T Berardini, S Mundodi, SY Rhee, R Apweiler, D Barrell, E Camon, E Dimmer, V Lee, R Chisholm, P Gaudet, W Kibbe, R Kishore, EM Schwarz, P Sternberg, M Gwinn, L Hannick, J Wortman, M Berriman, V Wood, N de la Cruz, P Tonellato, P Jaiswal, T Seigfried, R White, and Gene Ontology Consortium. The gene ontology (go) database and informatics resource. *Nucleic acids research*, 32(Database issue), 01 2004.

[13] Darren A. Natale, Cecilia N. Arighi, Winona C. Barker, Judith A. Blake, Carol J. Bult, Michael Caudy, Harold J. Drabkin, Peter D'Eustachio, Alexei V. Evsikov, Hongzhan Huang, Jules Nchoutmboube, Natalia V. Roberts, Barry Smith, Jian Zhang, and Cathy H. Wu. The Protein Ontology: a structured representation of protein forms and complexes. *Nucleic acids research*, 39(Database issue), January 2011.

[14] Carol J Bult, Harold J Drabkin, Alexei Evsikov, Darren Natale, Cecilia Arighi, Natalia Roberts, Alan Ruttenberg, Peter D'Eustachio, Barry Smith, Judith A Blake, and Cathy Wu. The representation of protein complexes in the protein ontology (PRO). *BMC Bioinformatics*, 12(1):371, 2011.

[15] Tomasz Adamusiak, Tony Burdett, Natalja Kurbatova, K Joeri van der Velde, Niran Abeygunawardena, Despoina Antonakaki, Misha Kapushesky, Helen Parkin-

son, and Morris A Swertz. Ontocat - simple ontology search and integration in java, r and rest/javascript. *BMC Bioinformatics*, 12(1):218, 2011.

[16] R-SPARQL - `http://code.google.com/p/r-sparql/`.

[17] RRDF - `http://cran.r-project.org/web/packages/rrdf/`.