Supplementary materials for *MetaSRA: normalized sample-specific metadata for the Sequence Read Archive*

November 2016

1 Biologically significant ontology terms

We only map samples to "biologically significant terms." A term is biologically significant if the presence of the term offers information regarding the biochemical processes occurring in the living sample. For example, disease terms are biologically significant. Similarly, terms for ethnic backgrounds such as the Experimental Factor Ontology (EFO) terms for "Caucasian" and "African American" are biologically significant due to the fact that certain genotypes may be more or less common in different ethnic populations.

We manually searched the ontologies for biologically significant terms that are near the roots of the ontologies' directed acyclic graphs. We then assume that all children of a biologically significant term are also biologically significant and retrieve all children of the manually selected nodes.

A file listing all terms deemed to be biologically significant is available from the MetaSRA's website: `http://deweylab.biostat.wisc.edu/metasra`.

2 Preprocessing of the Experimental Factor Ontology

There are certain idiosyncrasies unique to the EFO that caused errors in the ontology mapping process. To address these issues, we modified the EFO during a preprocessing procedure. This procedure involves the following steps:

1. **Add cell line synonyms to the EFO:** We use the Cellosaurus to add synonyms to cell line terms in the EFO. More specifically, each entry in the Cellosaurus includes a set of links to external references of that cell line. Such references for a given cell line may include the EFO's definition of that cell line. We add to the EFO's definitions all of the synonyms that appear in the Cellosaurus's entry for the respective cell line. For example, the EFO's term for the MCF-7 cell line does not include the synonym

"MCF.7"; however, this is included as a synonym for the Cellosaurus's entry. Thus, we add the synonym "MCF.7" to the EFO's entry for MCF-7.

2. **Remove incorrect synonyms from the EFO:** Many of the EFO's cancer-type terms are associated with synonyms that represent a more general concept than the term. Moreover, these synonyms are usually labelled as "exact synonyms" rather than "broad synonyms." For example, the term "hepatocellular carcinoma" has the exact synonym "Liver Cancer." Given this synonym, a metadata entry that includes the substring "Liver Cancer" will map to the term "hepatocellular carcinoma." We assert that this is an incorrect mapping. To avoid these erroneous mappings, we manually removed synonyms from the EFO's cancer-terms that represent a more general concept than the term.

3. **Convert EFO synonyms to lower-case:** Many of the synonyms in the EFO have a first character that is upper-case. For example, the term "breast carcinoma" has the synonym "Carcinoma of breast." For these synonyms, we convert the first character to lower-case. For example, the synonym "Carcinoma of breast" is converted to "carcinoma of breast."

3 Detailed description of the ontology mapping pipeline

The version of the pipeline used to generate the results in this paper consists of the stages listed below. We note that the design of our software is modular and allows for easy implementation of new stages. In the future, we plan to develop and refine the pipeline to further increase the accuracy of the mapped ontology terms.

1. **Filtering key-value pairs:** We created a blacklist of keys and a blacklist of values such that if a key appears in the blacklist of keys or a value appears in the blacklist of values, the key-value pair will be removed from the ontology mapping process.

   We remove all keys that describe a property that likely does not describe the sample. For example, we remove key-value pairs with keys "biomaterial provider", "study name", and "submitter handle." The blacklist of values include values that negate the key. Such values include "normal", "unknown", "none", and "no."

2. **Initializing the Text Reasoning Graph (TRG):** The initial TRG consists of a set of nodes that represent the raw set of key-value pairs describing the sample. First, a node is created for each key-value pair. From each of these "start nodes", we draw two edges to two artifact nodes – one artifact representing the key and the other artifact representing the

value. Figure 1 depicts the initial TRG for the following set of key value pairs:

```
cell line:  Parkin-expressing MRC5 fibroblasts
cell state:  Proliferating
source_name:  Prolif
```

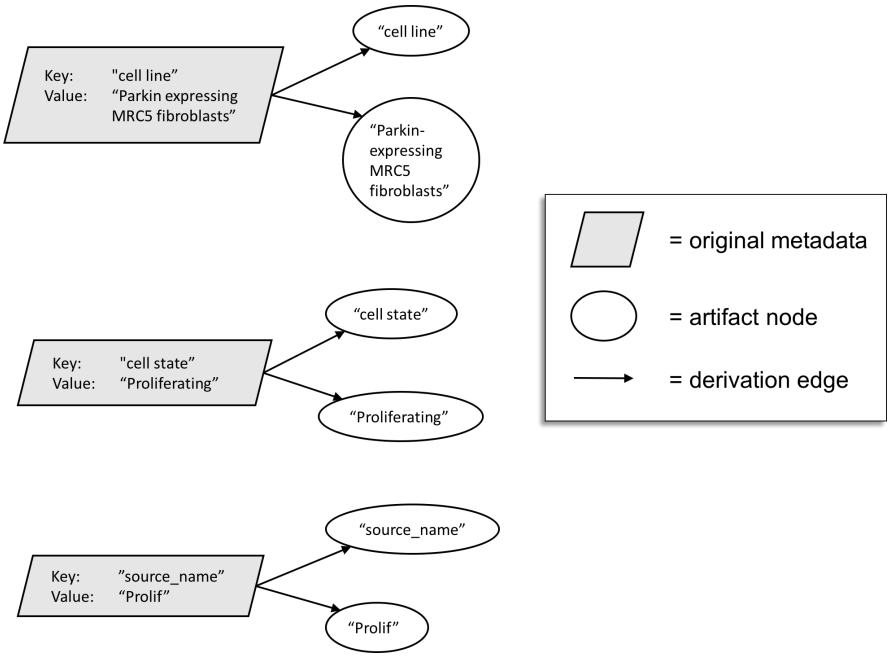

Figure 1: The initial TRG created from a set of key-value pairs describing a sample.

3. **Generating n-grams:** From each artifact node, we generate all n-grams for $n = 1, \ldots, 8$. We use the Python Natural Language Toolkit (nltk) to tokenize the text before constructing n-grams. For each n-gram generated from an artifact, we draw an edge from the original artifact to the derived artifact. Figure 2A illustrates an artifact node from the graph of Figure 1 with derived artifacts representing n-grams.

4. **Lowercase:** From each artifact node that represent artifacts with uppercase characters, we draw an edge to a new artifact node that has all lowercase characters. Figure 2B demonstrates this process.

5. **Delimiters:** The NLTK's tokenizer does not split on the characters "+", "-", "/", and "_". We therefore, split all artifact strings by these delimiters as shown in Figure 2C.

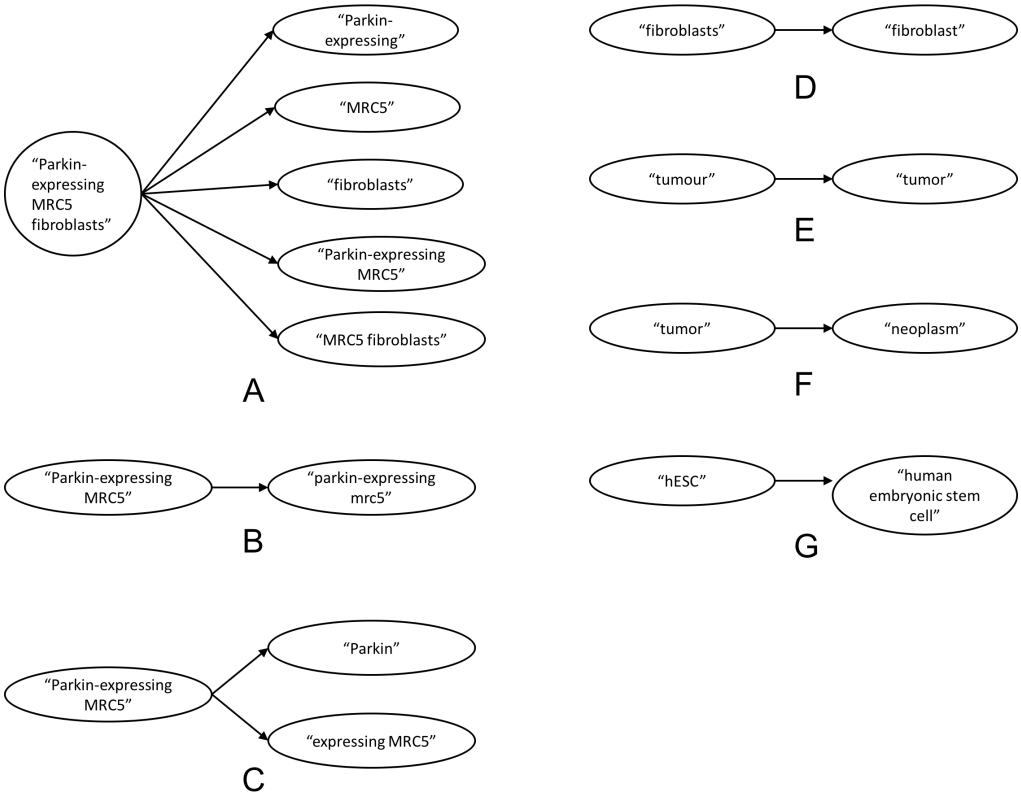

Figure 2: (A) An artifact node with derived n-grams. (B) An artifact with derived lowercase artifact. (C) Delimiting artifacts on special characters. (D) Deriving inflectional variants. (E) Deriving spelling variants. (F) Deriving custom synonyms. (G) Expanding acronyms.

6. **Inflectional variants:** We derive the inflectional variants of all artifacts by consulting the SPECIALIST Lexicon. This is demonstrated in Figure 2D.

7. **Spelling variants:** We derive the spelling variants of all artifacts by consulting the SPECIALIST Lexicon. This is demonstrated in Figure 2E.

8. **Manually annotated synonyms:** There are certain words that are very common in the metadata, but that are not included in the ontologies. For example, the word "tumor" is extremely common in the metadata, but is not present in the Disease Ontology. We filled such gaps by creating a small, custom thesaurus. In our thesaurus, "tumor" is given the synonym "neoplasm." The word "neoplasm" is a term in the Disease Ontology that is semantically equivalent to "tumor." This process is demonstrated in Figure 2F.

9. **Custom acronym expansion:** There are certain acronyms that are common in the metadata, but are not included in the ontologies. For example, the acronym "hESC" is very common in the metadata, but is not present in the Cell Ontology. We filled such gaps by expanding common acronyms. For example, we expand "hESC" to "human embryonic stem cell." This process is demonstrated in Figure 2G.

10. **Exact string matching:** We perform a preliminary mapping step in which we map artifacts to ontology terms by searching for exact matches between the artifact strings and term names and synonyms in the ontologies. This preliminary mapping stage is performed quickly using a trie data structure.

11. **Context-specific synonyms:** We create a list of "context-specific synonyms" and derive synonyms for artifacts when that artifact was derived from a value with a specific key. For example, a common key-value pair is `sex: F`. Here, the string "F" is an abbreviation for "female"; however, this is only known because the key maps to the EFO term for "sex." "F" in another context may not be an abbreviation for "female." This process is illustrated in Figure 3.

12. **Fuzzy string matching** We perform fuzzy string matching between the artifacts and the ontology terms. To more efficiently perform fuzzy string matching we store all ontology term names and synonyms in a Burkhard-Keller metric tree [1] with the bag-distance metric defined in [2]. When performing fuzzy string matching between a query string s and the strings in the metric tree, we retrieve all strings in the metric tree that are within a distance of 2 from s using bag-distance. Since bag-distance is a lower-bound on edit-distance, this process filters out all strings in the ontologies whose lower bound is greater than the threshold of 2 that we impose on fuzzy string matching. We then explicitly compute edit distance between s and the retrieved strings. We call a match if the following conditions

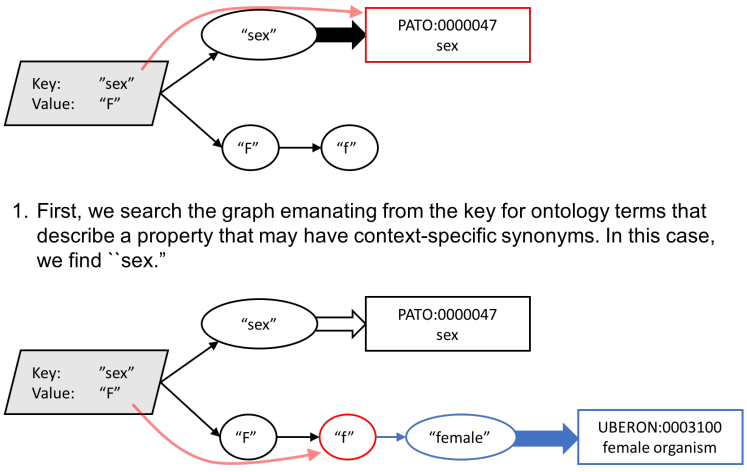

Figure 3: An example describing context specific synonyms.

hold: the length of the artifact is greater than 2, the edit distance is less than 2, and the edit distance is less than 0.1 the length of the longer string.

13. **Matching to custom terms:** There are several noun-phrases that are common in the metadata and that are superstrings of ontology terms, but that do not imply that the sample maps to the contained ontology term. For example, the phrase "blood type" describes a phenotype of the organism, but does not indicate that the sample was derived from blood. Similarly, "tissue bank" describes the organization that provided sample, but does not necessarily imply that the sample is a tissue sample. To differentiate the larger noun-phrase from the ontology term it contains, we maintain a custom list of misleading noun-phrases and remove ontology term mappings if those mappings were derived a substring of a misleading noun-phrase. For example, given the string "blood type", the artifact "blood" will be blocked from mapping to the ontology term for blood because it is a substring of the noun-phrase "blood type." Currently, we have 27 noun-phrases in our index and we will continue to build this index as we find more misleading noun-phrases that contain ontology terms.

14. **Remove extraneous cell-line matches:** Many cell lines have short names that oftentimes resemble acronyms or gene names. For example, "SRF" is a gene as well as a cell line in the Cellosaurus. Similarly, "MDS" is often used as an acronym for Myelodysplastic Syndromes and also happens to be the name of a cell line in the Cellosaurus.

    We remove extraneous mappings to cell line terms by searching the graph

emanating from the key for a lexical match to ontology terms such as those for "cell line" and "cell type". If such a match is *not* found, we search the graph emanating from the value for artifacts that have a lexical match to a cell line ontology term and remove all such ontology term nodes. This process is illustrated in Figure 4.

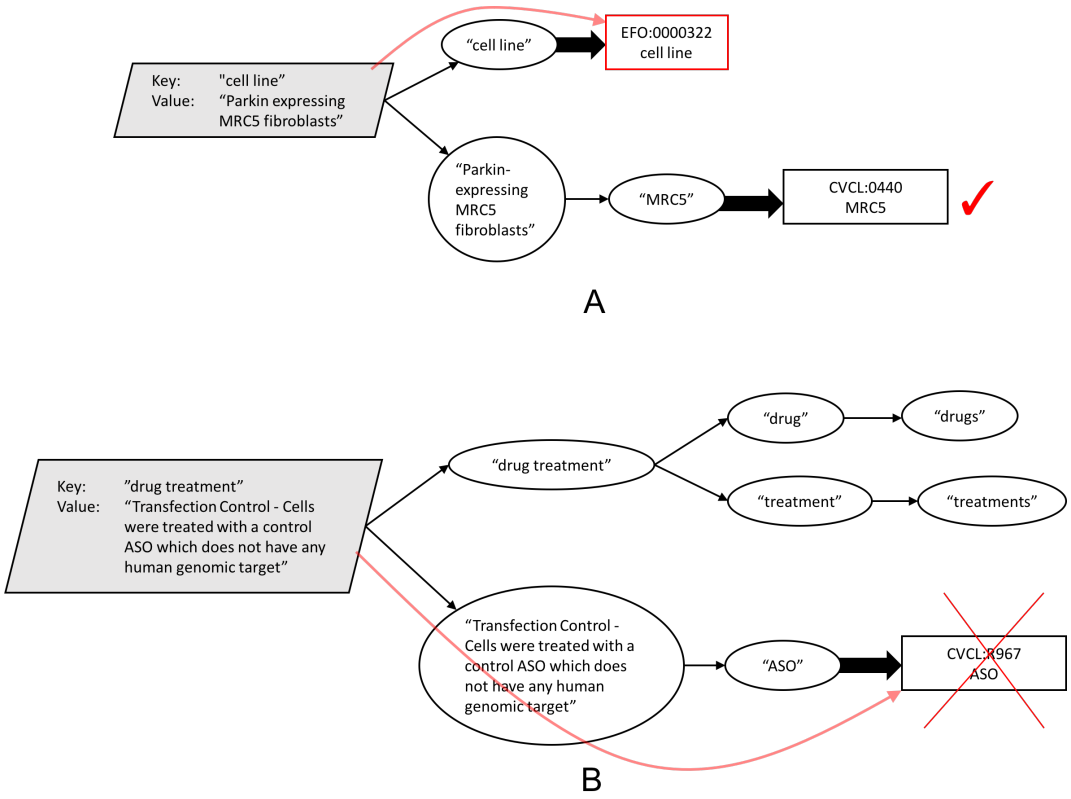

Figure 4: (A) The term "cell line" was found in the graph emanating from the key. Thus, we keep the cell line term for "MRC5" in the graph emanating from the value. (B) No ontology term for "cell line" or "cell type" was found in the graph emanating from the key. We therefore remove the cell line term for "ASO" in the graph emanating from the value.

15. **Map to linked-superterms:** The domain covered by the EFO overlaps with many of the other ontologies because it includes cell types, anatomical entities, diseases, and cell lines. In many cases, the EFO is inconsistent with other ontologies in how it draws edges between terms. For example, the term "lung adenocarcinoma" and "adenocarcinoma" are present in both the Disease Ontology and the EFO; however "adenocarcinoma" is a parent of "lung adenocarcinoma" only in the Disease Ontology and not in

the EFO. These inconsistencies pose a problem when we filter for the maximal phrase-length in the metadata. For example, when a sample maps to "lung adenocarcinoma" and "adenocarcinoma", we remove "adenocarcinoma" because it is a substring of "lung adenocarcinoma". This is valid for the Disease Ontology because the term for "adenocarcinoma" is implied by "lung adenocarcinoma" by its position in the ontology. However, this results in a false negative for the EFO version of this term.

To counteract this problem, we link the terms in the EFO to terms in the other ontologies. Two terms are linked when they share the same term-name or exact-synonym. Then, when an artifact maps to a term, we traverse the term's ancestors and map to any terms that are linked to those ancestors. In the case of "lung adenocarcinoma", we would traverse the ancestors of this term in the Disease Ontology and map to the EFO's "adenocarcinoma" because it is linked to the Disease Ontology version of this term. Figure 5 illustrates this process.

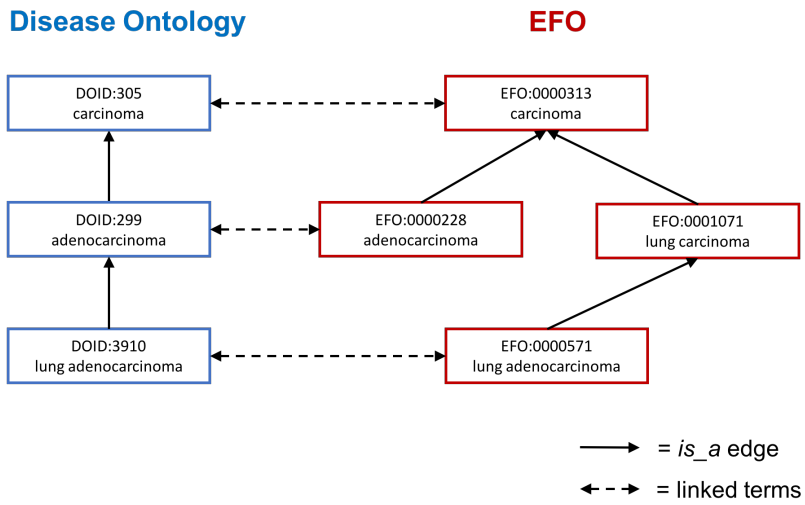

Figure 5: An example of linked terms between the Disease Ontology and the EFO. If a sample maps to "lung adenocarcinoma" in the Disease Ontology, we follow all ancestors and also map to linked terms of those ancestors. In this case, the EFO's "adenocarcinoma" term will also be mapped.

16. **Cell line disease implications:** The EFO is missing edges between disease cell line terms and the corresponding disease terms. For example, the term "cancer cell line" does not have an edge to "cancer." To fill this gap, if a sample maps to a cell line category term, we also map to the corresponding disease terms.

17. **Block superterm mapping:** It is a common occurrence for disease

ontology terms to include anatomical entities in their name. For example, "breast cancer" includes "breast" as a substring. It would be incorrect to map "breast" to the sample because this word localizes the cancer, but does not localize the origin of the sample. We note that it is possible the sample was indeed derived from breast tissue; however, it is also possible the sample originated from other tissue such as a malignant site. We maintain a conservative approach and avoid mapping to "breast." We implement this process by designing each artifact node to keep track of the original character indices in the metadata from which it was derived. After mapping all artifacts to the ontologies, we remove all ontology terms that were lexically matched with an artifact node that is subsumed by another artifact node that matches with an ontology term.

18. **Custom consequent mappings:** We maintain a small list of 6 common terms that imply other terms. For example, if a cell maps to a the EFO term for "cell line", we consequently map the sample to the Cell Ontology's term for "cultured cell."

19. **Real-value property extraction:** We maintain a list of ontology terms that define real-value properties. Currently, we use 7 terms including "age", "passage number", and "timepoint." Future work will entail expanding this list to more terms. To extract a real value property from a key-value pair, we search the graph emanating from the key for a match to a property ontology term. If such a property is found, we search the graph emanating from the value for an artifact that represents a numerical string as well as a unit ontology term node (for example "46" and "year"). From this process, we extract the triple (property, value, unit). For example, given the key-value pair `age: 46 years old`, we extract ("age", 46, "year").

20. **Filtering mapped ontology terms by semantic similarity:** The ontologies are structured so that each synonym of an ontology term is given a synonym-type. These types include "exact", "broad", and "narrow." These synonym-types describe the relationship between the synonym string and the term name. An "exact" synonym indicates that the string is semantically closer to the ontology term name than a "broad" synonym. If an artifact matches with multiple ontology terms, we examine the targets within these terms that to which the artifact matched. We rank these targets according to the semantic similarity with the ontology term name and take the match with the highest similarity. For example, an "exact" synonym is semantically closer to the ontology term name than a "narrow" synonym. Thus, given an artifact that matches to an "exact" synonym of one term and a "narrow" synonym of another, we discard the "narrow" synonym match and keep the "exact" synonym match. If an artifact matches to multiple targets in the ontologies, the ontology term with the semantically nearest matched target is likely to be the best match with the artifact.

For example, given an artifact "skin", we find several terms in the Uberon ontology that have a synonym "skin": "zone of skin" (exact synonym), "skin epidermis" (broad synonym), "skin of body" (related synonym), and "integument" (related synonym). Of these terms, "skin" is semantically most similar to the term "zone of skin" because it is an exact synonym of this term. We therefore keep this mapping and discard the rest.

21. **Consequent cell line mappings:** Our pipeline draws edges between cell line ontology term nodes and the ontology terms that describe the cell line. For example, if the TRG contains the node for the cell line "HeLa", we draw an edge to the ontology terms for "adenocarcinoma" and "female" because this cell line was derived from a woman with cervical adenocarcinoma. We consider such mappings to be consequent mappings because they are retrieved using an external knowledge base.

    This knowledge base was created from data we scraped from the ATCC website at `https://www.atcc.org`. To construct mappings between cell lines and ontology terms, we ran a variant of our pipeline on the scraped cell line data. We scraped cell line metadata for all cell lines that are present in the Cellosaurus.

22. **Consequent developmental stage mappings:** If the sample maps to a real-value property with property "age" and unit "year", we check whether the value is greater than 18. If so, we consequently map the sample to the EFO and Uberon terms for "adult."

4 Evaluation of sample-type prediction on the training set

We note that the test data set was small compared to the training data set. To provide an estimate of the performance of the classifier on a larger data set, we ran the algorithm using leave-one-out cross validation on the training set. The algorithm achieved 0.855 accuracy on this data set. Figure 6A shows the confusion matrix and Figure 6B shows the calibration of the model.

5 Prediction procedure for predicting sample-type

Although we train a one-vs-rest classifier using logistic regression binary classifiers, we ultimately use a custom decision procedure for making a sample-type prediction. This procedure entails limiting the possible predicted sample-types based on ontology terms that were mapped by our computational pipeline. The algorithm chooses among the remaining possible sample-types by selecting the sample-type with highest confidence according to the one-vs-rest classifier.

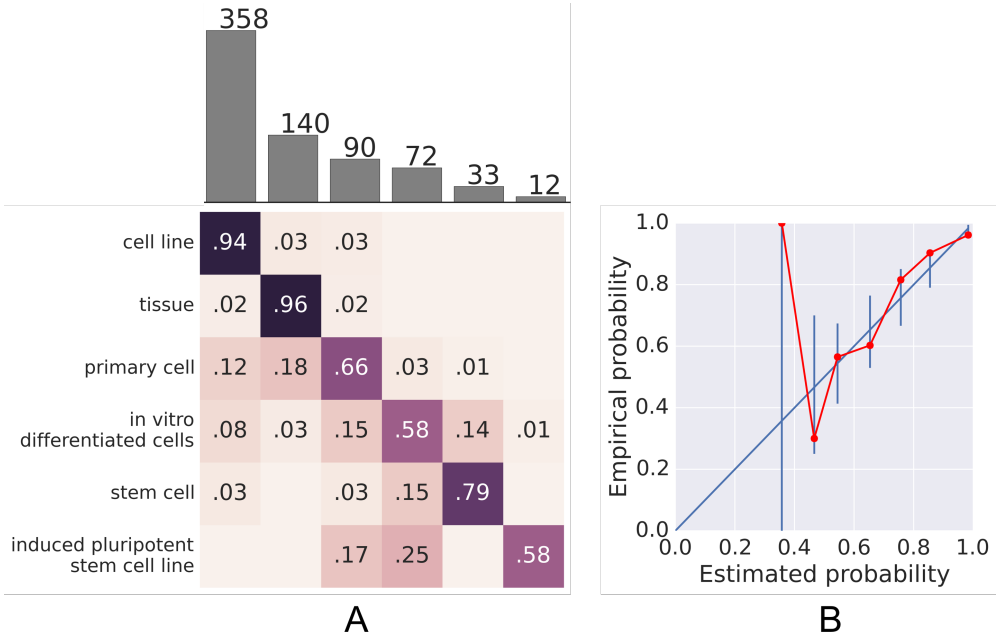

Figure 6: (A) The confusion matrix of the algorithm on the training set evaluated using leave-one-out cross validation. The bar graph above the matrix displays the distribution of classes within the training set. (B) Plotting the calibration of the classifier.

To provide an example, if the ontology term "stem cell" was mapped to the sample, we set $p(y = j|x) = 0$ for $j \in \{$`tissue, cell line, primary cells`$\}$. We then compute the probabilities $p_i := \frac{p(y=i|x)}{\sum_h p(y=h|x)}$ for each i. Our final prediction is then $\hat{y} = \text{argmax}_i p_i$. In summary, this process asserts that if "stem cell" mapped to the sample, then the sample must be either a stem cell sample, in vitro differentiated cell sample, or induced pluripotent stem cell sample. We let the classifier decide which is the most likely label among these possible labels.

More specifically, we follow the following steps for making a prediction:

1. If the sample maps to "xenograft" (EFO:0003942), then we predict `tissue` with confidence 1.0.

2. If the sample was passaged (i.e. maps to a real-value property tuple with property "passage number" and unit "count"), then we assert the sample cannot be `tissue`. If the number of passages is greater than 0, then we assert the sample cannot be `primary cell`.

3. If the sample maps to a cell line, then we check the Cellosaurus for the cell line category. We map the Cellosaurus cell-line category to a set of possible sample types as follows:

   - Induced_pluripotent_stem_cell: `in vitro differentiated cells`, `induced pluripotent stem cell line`
   - Cancer_cell_line: `cell line`
   - Transformed_cell_line: `cell line`
   - Finite_cell_line: `cell line`
   - Spontaneously_cell_line: `cell line`
   - Embryonic_stem_cell: `stem cells`, `in vitro differentiated cells`
   - Telomerase_cell_line: `cell line`
   - Conditionally_cell_line: `cell line`
   - Hybridoma: `cell line`

4. If the sample maps to the term "stem cell", then we remove the possibility that the sample type is `cell line`, `tissue`, or `primary cells`.

5. If the sample maps to a specific cell-type term (i.e. any term that is a child of "somatic cell"), then we remove the possibility that the sample-type is `tissue`. This follows from our observation that when the metadata describes a specific cell-type, the sample consists of homogenous cells that have been isolated and filtered. The sample no longer consists of cells positioned in their original three dimensional structure and is thus not a tissue sample.

References

[1] W.A. Burkhard and R.M. Keller. Some approaches to best-match file searching. *Communications of the ACM*, 4(230-236), 1973.

[2] I. Bartolini, P. Ciaccia, and M. Patella. String matching with metric trees using an approximate distance. *SPIRE 2002 Proceedings of the 9th International Symposium on String Processing and Information Retrieval*, 2002.