Automated Verification of Phenotypes using PubMed

Ryan Bridges
Epic Systems
Verona, WI 53593
rybridges90@gmail.com

Jette Henderson
University of Texas at Austin
Austin, TX 78712
jette@ices.utexas.edu

Joyce C Ho
Emory University
Atlanta, GA 30322
joyce.c.ho@emory.edu

Byron Wallace
University of Texas at Austin
Austin, TX 78712
byron.wallace@utexas.edu

Joydeep Ghosh
University of Texas at Austin
Austin, TX 78712
jghosh@utexas.edu

ABSTRACT

In the realm of data driven clinical research, medical concepts, or phenotypes, are used to serve as indicators for patient clusters of interest. Often, studies will use groups of algorithmically generated phenotypes (feature groups) to predict the occurrence of heart disease, diabetes, and others. When these groups are algorithmically generated, the most common method of verification is manual human annotation, which can be time consuming and sometimes inconsistent. In this paper, we propose a supervision-free method of verification that uses co-occurrence in PubMed articles to determine clinical significance.

We are able to show the method separates randomly generated groups of phenotypes from those curated to demonstrate known clinical narratives. Further, we suggest the future potential of the method to explain causation for phenotypes to be grouped and identify subsets of phenotype groups that are better predictors for a particular patient cluster.

1. INTRODUCTION

Computational phenotyping is the practice of mapping the raw information contained in Electronic Health Records (EHRs) into sets of clinically relevant features, or phenotypes. Clinicians can use the EHR-based phenotypes to identify patients with specific characteristics or conditions of interest. Phenotypes also enable cohort identification to target patients for screening tests and interventions, support surveillance of infectious diseases, and aid in the conduct of pragmatic clinical trials and comparative effectiveness research [15]. An example is the type 2 diabetes mellitus phenotype (shown in Figure 1) [7]. The flowchart depicts a series of characteristics that must be present in a patient's EHR for that patient to be identified as a type 2 diabetes case patient.

Constructing phenotypes can be a manual, iterative, and labor-intensive process requiring domain expertise [2, 3, 10]. Recent efforts have focused on machine learning developed methods to automatically generate candidate computational phenotypes in a high-throughput, unsupervised manner [8, 9, 11, 20, 22]. Figure 2 illustrates the conceptual process of a high–throughput, automatic phenotype generation process [10] to aid the discovery of knowledge. However, domain experts are still required to annotate these candidate phenotypes to verify the clinical significance, and several issues can arise during the annotation process beyond time–consumption. First, domain experts may disagree on the clinical relevance of a candidate phenotype. For example, two annotators disagreed on the clinical significance of the phenotype shown in Table 1. Second, the unsupervised methods may generate phenotypes that represent previously unknown, but significant clusters of patients that do not readily map to any previously known domain knowledge. Additionally, given the diverse and different clusters of patients grouped by these methods, annotators may feel the objective or the phenotypes themselves are vague or undefined. Thus, there is a need to develop an automated, data–

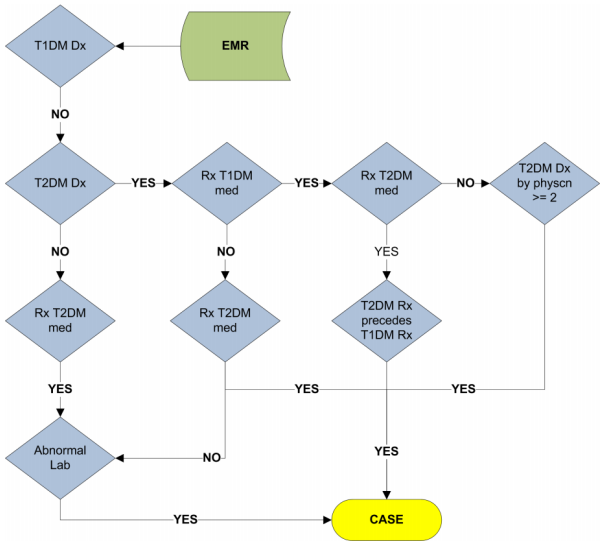

Figure 1: Type 2 diabetes phenotype from the Phenotype Knowledgebase [7], Source: https://phekb.org/phenotype/type-2-diabetes-mellitus

© 2016 ACM. ISBN 978-1-4503-2138-9.
DOI: 10.1145/1235

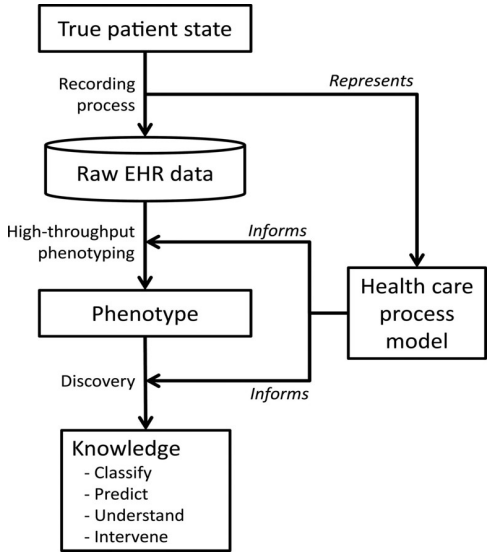

Figure 2: Overview of high-throughput phenotype generation process [7].

Hypertension
Osteoarthrosis and allied disorders
Urinary tract infection
Analgesics

Table 1: Example of a candidate phenotype where annotators disagree on "clinical significance". Blue entries represent the patient's diagnoses, red entries are medications.

driven process to serve as an unbiased means of validation, leveraging all the medical expertise that has been collected to date.

In this paper, we present an automated framework that can provide third-party verifcation to facilitate and improve the annotation process and accuracy. Our method records co–occurrences of phenotypic terms in PubMed[1], a publicly available repository for medical literature that contains over 40GB of medical literature from over 6000 journals. From this data, we use lift, a metric that summarizes if two or more items co–occur more often than average while accounting for the commonality of the item, to determine clinical relevance. We analyze the lift patterns of 50 phenotypes produced by several of the high–throughput methods described above and demonstrate the correlation between lift with the significance as judged by domain experts. We also illustrate the upper and lower bounds of the lift metric by comparing phenotypes generated randomly to those curated to represent known medical concepts. We demonstrate the method is agnostic to the algorithm that generated the phenotypes by showing it can effectively determine the validity of candidate phenotypes produced by two different high-throughput algorithms, as well as curated phenotypes. We note however, that if an algorithm itself uses PubMed to generate phenotypes, this method of verification should not be used.

[1]http://www.ncbi.nlm.nih.gov/PubMed

2. RELATED WORK

Many researchers have used PubMed to explore and discover issues in biology, medicine, and health informatics. Multiple software packages like LitInspector [6] and PubMed.mineR as well as python packages like Pymedtermino[2] and Biopython[3] have been developed to help researchers extract and visualize PubMed. Jensen et al. provide a thorough overview of how PubMed can be harnessed for information extraction and entity recognition [12]. Amongst the two methods they discuss for information extraction, natural language processing and co–occurrence analysis, co–occurrence is more prevalent due to its straightforward implementation and the intuitive interpretation of the results. While co–occurrence analysis does not give information about the type of relationship or any causal information, work done on bias towards publishing positive results allows for the assumption that when two phrases occur together the relationship exists ([4, 5, 18]). Raijpal et al. perform frequency analyses on PubMed abstracts in order to 1) predict emerging trends in genes and the diseases to which they are related and 2) explore the possible link between two diseases, psoriasis and obesity [17]. This type of exploration can be thought of as phenotype discovery.

Co–occurrence analysis of PubMed has also be used to explore relationships between phenotypes and genotypes. Pletscher-Frankild et al. perform co–occurrence analysis on the abstracts along with genetic data and evidence from other studies to discover disease-gene associations [16]. They apply a scheme that takes into account co–occurrence within an abstract and within a sentence, and then convert the co–occurrence scores into Z-scores, which corrects for abstracts of different lengths. Adding another layer of complexity to the co-occurrence, [19] queried MEDLINE for a set of diseases, to construct disease by genes vectors where a vector element is zero or replaced by the number of times the disease and gene co–occur in articles. This approach, called second order co–occurrence, is meant to model the idea that diseases may be related but not necessarily frequently occur together, but they may belong to a cluster of diseases that co–occur frequently overall, which they ascertain through thresholding the cosine similarity of the disease vectors.

There has been limited research into using PubMed as a validation tool. For example, Boland et al. mined EHR records for patients who had disease-specific codes and then compared the association between birth month and the disease to a group of control patients who did not have the disease codes present in their EHRs [1]. They validated their results against papers queried from PubMed that had disease and birth month as topics.

PubMed has also been used as a resource for generating annotations. Neveol et al. used pre-existing tools to generate candidate annotations for PubMed queries and then measured the inter-annotator agreement as well as annotation time between sets of queries with and without the candidate annotations [14]. While they were annotating PubMed in order to understand PubMed users' needs, their work shows that annotating tools can not only speed up annotating time, but increase inter-annotator agreement. However, annotating before annotators can examine the text can have the effect of biasing annotators, so it should be used

[2]http://pythonhosted.org/PyMedTermino/
[3]http://biopython.org/

carefully.

3. METHOD

Annotators are often clinicians volunteering their time, and may or may not have computational backgrounds or annotation experience. Furthermore, medical perspectives can be drastically different amongst annotators as they are impacted by factors such as their medical expertise, patient population, and medical education (medical school and residency). In addition to these reasons, the vague and subjective nature of the annotation task can result in low inter–rater agreement amongst the different clinicians, with one high–throughput phenotyping method reporting an inter–rater agreement of 0.81 [20]. We propose to leverage the 26+ million biomedical literature citations found in PubMed as an objective third–party annotator by developing an automated method to capture co-occurrence of phenotypic items within the clinical narratives described throughout PubMed. Our framework utilizes the inherent publication biases associated with medical literature to observe that if a concept pairing is clinically significant, several papers will make mention of these concepts together in such a large and diverse corpus. This provides a reasonable objective baseline for determining significance that can be used as corroboration for existing annotation or as a tool to assist annotation efforts.

Although the idea is conceptually simple, there are several challenges that our automated framework must address. The representation of each element of the phenotype is important as it can drastically impact the number of articles returned during the PubMed query. Second, the co–occurrence search needs to account for encoding, form/tense, incorrect spellings, and also regularization. Finally, the co–occurrence metric should reflect the number of items contained in the phenotype elements as well as the commonality of the item itself in the PubMed literature. Thus, our automated verification process consists of several steps:

1. Feature (n-gram) generation from phenotypes
2. Counting co-occurrence in PubMed articles
3. Calculating and normalizing feature lifts within a particular group to determine significance

Figures 3 and 4 show the process for a phenotype from the feature generation to the calculation of the clinical significance.

3.1 Feature Extraction

Since medical terms can have multiple synonyms and representations across different articles, our framework first generates a suitable list of synonyms and related concepts for each element in the phenotype (each phenotypic item). Table 1 displays an example of a candidate phenotype, which was generated by an automatic method, that could be presented to the annotators. In this example, the term "hypertension" can also be referred to in various articles as "high blood pressure", "HBP", or even "cardiovascular disease." Thus, the appropriate terms must be used so that hypertension as a concept may be discovered in a PubMed article reasonably often when it is being mentioned (recall), but not produce many false positives (precision). To produce a representation that has high recall and high precision, we first generate a large set of possible representative n-grams, and then filter all the candidate n-grams down to the most relevant n-grams (the filtration process is discussed in the next section).

A naïve approach is to use the item as it appears rather than to perform the candidate generation and filtration process. However, this can yield low recall as the text is often too specific or not phrased naturally. For example, note the phenotypic item "unspecified chest pain" in the phenotype in Figure 3. The adjective "unspecified" reduces the chance that the more appropriate bi-gram "chest pain" will be discovered. While omission of "unspecified" is a clear step to improve recall without sacrificing precision ("unspecified" is in fact omitted in pre-processing), not every phenotypic item has such a clear solution. Likewise, using a collection of individual words (unigrams) to represent phenotypes yields high recall, but low precision, as it is difficult to filter out enough of the words that lack specificity, but are important in some cases (e.g. "disease" or "results"). "Unspecified", "chest" and "pain" will obviously find all occurrences of "chest pain," but will also capture mentions having little to do with "chest pain" resulting in low precision.

In order to achieve high recall, a phenotypic item must be recognized when a conceptually equivalent or similar term is present and capture situations when an entirely dissimilar string is used to represent the same item (e.g. "heart attack" and "myocardial infarction"). In order to recognize such "aliases," we utilized several medical ontologies to collect a set of closely related concepts and synonyms to a given phenotypic item. One of the most complete and commonly used ontologies is the "Systemized Nomenclature of Medicine - Clinical Terms" (SNOMED-CT) ontology [21]. The first order connections on the SNOMED-CT ontology graph for a concept provided a reasonable number of aliases that we could then filter. We supplemented SNOMED-CT with two other common ontologies, ICD–10 and the NCBI MeSH terms. During implementation, we extracted the SNOMED-CT and ICD–10 ontologies with a python library called Pymedtermino.[4] Biopython,[5] a tool which also provides an interface to the Entrez tools, provided access to the MeSH terms. We queried the Pymedtermino and Entrez APIs to collect aliases from these three ontologies, and then placed the related terms from each ontology into a (potentially large) list of candidate concepts that may represent the phenotypic item. After assigning every phenotype a pooled set of related concepts, we removed stopwords, and then extracted the set of all unigrams, bi-grams and tri-grams from the related concepts.

Figure 3 shows the feature generation process for the phenotypic items "heart failure" and "antianginal agents". For "heart failure", our framework generates the related concepts "congenital heart disease", "left ventricular structure", "myocardium", and "heart valve disorder" as indicated by the middle column of the figure. In the case of the term "antianginal agents", the algorithm generates "thyroid structure", "hydrochloride", "mophologic abnormality" and "penbutolol product" as potential n–grams. In the scenario when a phenotypic item contains many (greater than 250) related n-grams, a subset of 250 were randomly selected. The choice of 250 n-grams allowed sufficient coverage of the related concepts, while allowing the computation time of the filtration

[4]http://pythonhosted.org/PyMedTermino/
[5]http://biopython.org/

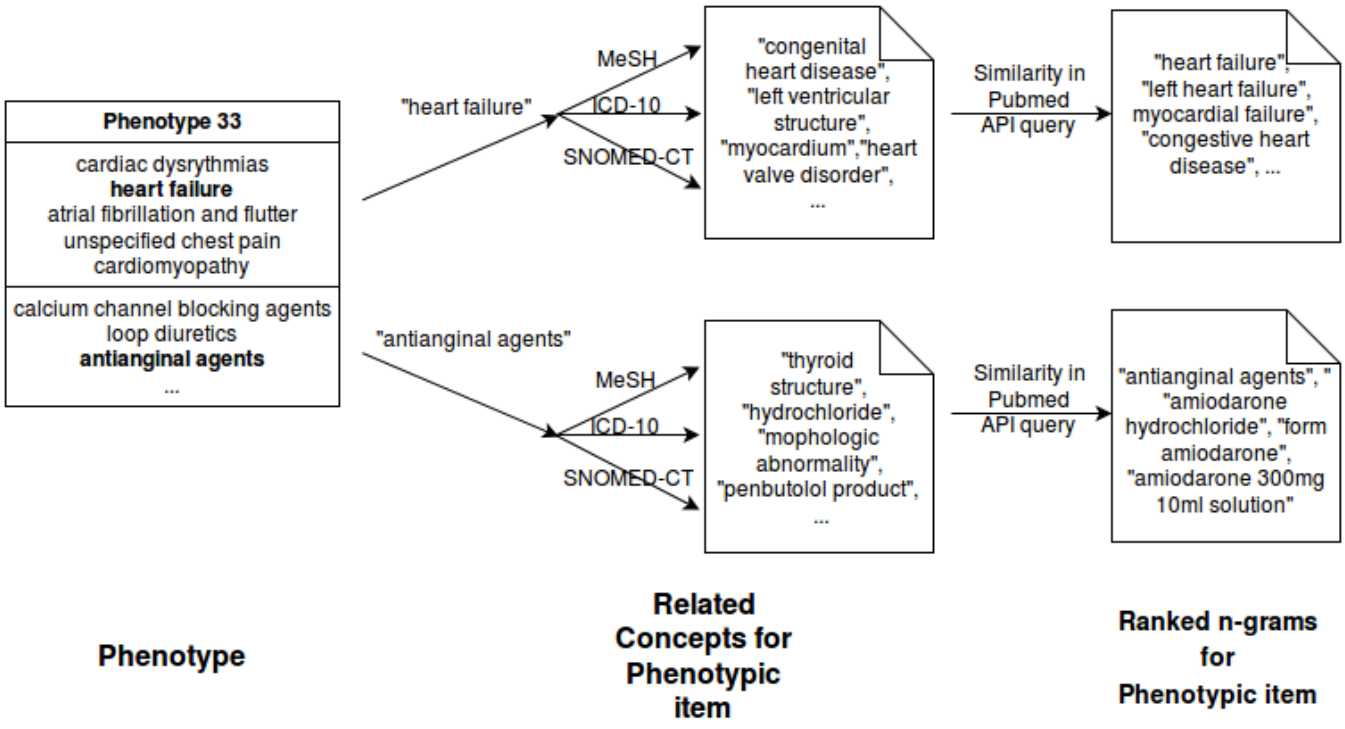

Figure 3: Feature extraction for phenotypic items

step to remain feasible.

3.2 Selection and Filtration of candidate n-grams

After extracting all possible n–grams (unigrams, bigrams and/or trigrams) relating to the item, our framework determines the n–grams that are most related to the phenotypic item, which we refer to as the selection and filtration process. This process orders the n–grams generated from the previous process (e.g., the first and middle column of 3) by "relevance" so that our algorithm avoids unnecessary queries of the PubMed system and can therefore run more efficiently. In addition, it also provides a tunable knob to find the optimal number of representative n–grams to optimize both precision and recall of the phenotypic item concept in the articles.

We calculate the "relevance" as the percentage overlap between the sets of PubMed query results from the original phenotypic item and each of its representative n-grams. This is done by recording the set of papers returned by each query and then calculating the size of the intersection between the set of papers returned for the original item and each of the subsequent n–gram queries. We tried Word2vec [13] as a semantic similarity measure, but the empirical results generated more false positives than our PubMed querying method. Thus, the semantic similarity of the phenotypic item phrase and its n–grams are roughly measured by the PubMed search index, rather than a more complicated semantic measure.

Each phenotypic item is assigned a ranked list, based on the relevance score, of representative n–grams. Table 2 shows four examples of the original phenotypic item and the ranked list with the eight highest ranked n-grams and their associated relevance scores. Based on our experimental results, we found selecting fifteen or even ten of the top ranked n–grams produced many false positives. Using six of the top ranked n-grams gave a tolerable number of false positives. In addition to restricting each representation to six n-grams, we pared down the list of aliases even more by ordering the set of all n–grams by their sentence frequency in PubMed, as well as their interaction frequency with other phenotypes, and removing the most frequent 5% from the sentence frequency and interaction frequency lists. Table 3 lists the removed features and the items that are typically uninformative or generic. More work on consensus filtration, however, is merited.

3.3 Co-occurrence search in PubMed

NCBI has a publicly available download of PubMed. Using a randomly selected subset of 25% of the articles available in PubMed, we searched for occurrences of the representative/relevant n-grams (generated in the last section) for all items within each phenotype. For all articles in the subset, any sentence containing one or more of the n-grams from any phenotypic item was noted, and the set of n-grams appearing in the sentence was added to a master list of all co-occurrences. Each sentence was minimally processed, only regularizing capitalization and encoding (utf-8), taking out words included in NLTK's English stopword list, using a conservative regular expression to remove references (e.g. Smith, et al.), and removing special characters like quotes and parenthesis. The form/tense and spelling of words were left as written to be consistent with the n-grams derived from the ontology related phrases.

Any occurrence of an n-gram that was a part of the set of 3 to 5 n-grams representing a phenotype counted as an

Original representation	Ranked list of n–grams
'Angiotensin-converting enzyme inhibitors'	('angiotensin-converting enzyme, inhibitor', 0.858) ('reaction ace inhibitor', 0.214) ('due ace', 0.207) ('hyperkalaemia due angiotensin-converting', 0.138) ('angiotensin-converting-enzy me inhibitor allergy', 0.082) ('inhibitor induced hyperkalemia', 0.071) ('antihypertensive drug disorder', 0.065) ('antihypertensive agent disorder', 0.065)
'Disorders of lacrimal system'	('lacrimal apparatus diseases', 0.636) ('disorders lacrimal system', 0.603) ('disorders lacrimal', 0.551) ('lacrimal system', 0.51) ('lacrimal apparatus', 0.478) ('lacrimal', 0.437) ('lacrimal structure', 0.315) ('lacrimal drainage', 0.229)
'anxiolytics, sedatives, and hypnotics'	('anxiolytics sedatives hypnotics', 1.0) ('anxiolytics sedatives', 0.996) ('hypnotic anxiolytic', 0.518) ('hypnotic drug abuse', 0.304) ('dependent sedative', 0.272) ('sedative hypnotic drug', 0.264) ('sedative', 0.261) ('sedatives hypnotics', 0.261)
'Other and unspecified complications of the puerperium, not elsewhere classified'	('unspecified complications puerperium', 1.0) ('other unspecified complications',0.229) ('unspecified complications',0.217) ('complications puerperium',0.079) ('unspecified', 0.047) ('other unspecified', 0.043) ('puerperium', 0.043) ('complications puerperium elsewhere', 0.028)

Table 2: Four original phenotypic items and their associated top eight most highly ranked representative n–grams. The score represents the semantic similarity measure derived from the PubMed search index.

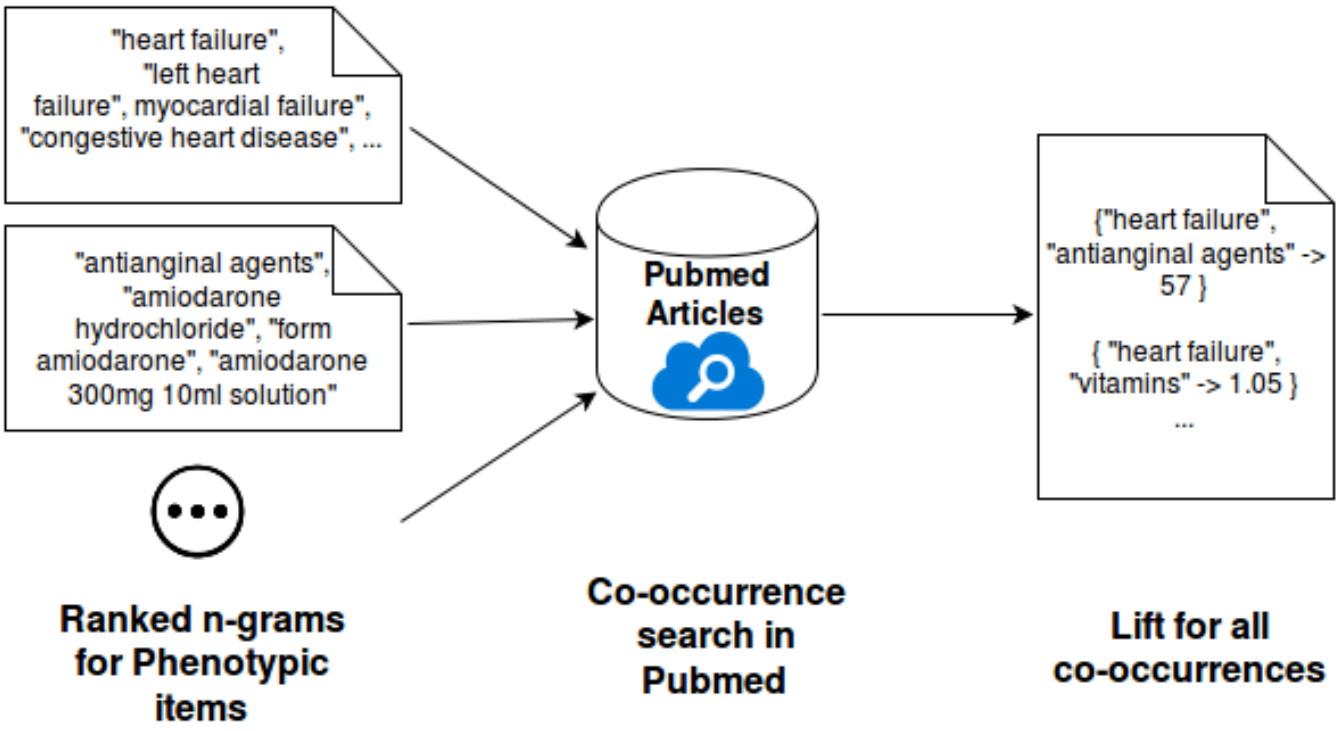

Figure 4: Significance calculation for phenotypic items

'body', 'upper', 'ie', 'skin', 'stress', 'conditions', 'type', 'agents', 'multiple', 'value', 'diseases', 'tumor', 'disorders', 'infections', 'tissue', 'disease' , 'period', 'active', 'part', 'site' , 'form'

Table 3: Examples of uninformative n-grams that were removed.

appearance for its phenotypic item. This simplifying assumption eliminated the need to weigh n-grams by their "relevance". The assumption also ignores if a sentence contains more than one n-gram for an item. Using this measure of co-occurrence, the lift of every co-occurrence of phenotypic items was calculated. Recall that given A, B, and C in a sentence:

$$\text{lift}(A, B, C) = \frac{P(A \cap B \cap C)}{P(A) * P(B) * P(C)}$$

Probabilities are calculated as the number of sentences where the item occurs divided by the total number of sentences. While studying all possible combinations of phenotypic items may be interesting for identifying significant subsets of phenotype groups or connections between phenotypes, we examined only the average lift of phenotypic items within a given phenotype. This allows for a simple classification of a phenotype as "clinically significant" or "not clinically significant"– this classification is discussed below.

For every co-occurrence, all possible subsets of co-occurring phenotypes were also counted. For example, when A, B, C, and D co-occurred in a sentence, a co-occurrence for (A,B,C), (B,C,D), (A,B), (A,C), and so forth, were counted. In this way, the lifts for any combination in the power set of all phenotypes that co-occurred were counted. This allows for convenient lookup of any co-occurrence of interest. We also introduce a term, "phenotype cardinality", for convenience, to denote the number of items in a co-occurrence. For example, (A,B,C) would have phenotype cardinality 3. While this complete set of co–occurrences is theoretically very large, not every combination of phenotypes is realized in practice.

The lift metric normalizes co–occurrence probability by the probability of all phenotypes co-occurring assuming they are probabilistically independent. This means that any lift greater than 1 indicates statistical "significance". However, since these co-occurrences are subject to grammatical rules, etc., lifts for co-occurrences are nearly always greater than 1. As we are interested in filtering out all but clinical significance (ignoring significance introduced by grammar and language convention), we randomly generated phenotypes from a set of phenotypic items, and measured the lift significance of these "phenotypes" to establish the level of lift significance introduced by the possible grammatical and language artifacts. We found that, across all phenotypes (including the randomly generated ones), lift was strongly positively dependent on the number of items included in the co-occurrence. In fact, lift appears to be almost perfectly exponential as a function of the number of items included in the co-occurrence, which is illustrated in Figure 5.

Due to this dependence on the number of items, the lift of each co-occurrence is normalized according to the number of items it contains. Co–occurrences are first placed into groups by the number of phenotypic items represented (regardless of the clinical phenotype group membership). For convenience, we refer to the number of phenotypic items

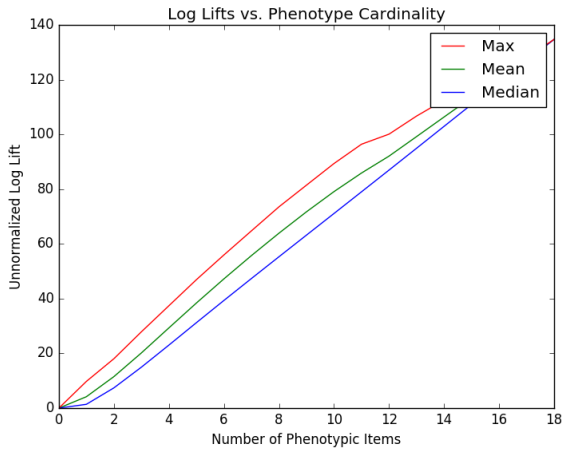

Figure 5: Log lifts versus Phenotype Cardinality

in a co-occurrence as the "phenotype cardinality". Table 4 shows an example of a phenotype with the phenotype cardinality of six and a lift of 17. Then, the mean, median and standard deviation of the lifts for each of the phenotype cardinality groups was calculated. We note that the mean and standard deviation of each of these cardinality groups were skewed high, as the max lifts were significantly further from the median than the below-median lifts (Figure 5).

For this reason, each lift was centered around the median of its phenotype cardinality group and then divided by the standard deviation. Since the standard deviation is artificially increased by the largest lifts, the normalized above median lifts are still much greater than 0 than the below median lifts are below. This fact makes it so that a lift threshold is likely to be closer on average to the lift of not significant groups than to significant groups.

4. EMPIRICAL STUDY

4.1 Dataset Description

Our study uses the annotated results of candidate phenotypes generated by different unsupervised, high–throughput phenotype generation processes. The first automatic method, Rubik [20], generated phenotypes from a de–identified EHR dataset from Vanderbilt University Medical Center with 7,744 patients over a five year observation period. For more details about the pre–processing of the data and phenotype generation, please refer to their paper [20]. The authors graciously shared the file with 50 computational phenotypes as well as the annotations of the three domain experts. For each phenotype, each expert assigned one of the following three choices: 1) yes - the phenotype is clinically meaningful, 2) possible - the phenotype is possibly meaningful, and 3) not – the phenotype is not clinically meaningful. The second set of candidate phenotypes were generated by Marble [9] using the EHR data of a random subset of 10,000 patients from the *Centers for Medicare and Medicaid Services (CMS) Linkable 2008-2010 Medicare Data Entrepreneurs' Synthetic Public Use File (DE-SynPUF)*, a publicly available dataset with claim records that span 3 years.[6] The

[6]For more information see https://www.cms.gov/Research-Statistics-Data-and-Systems/Downloadable-Public-Use-Files/SynPUFs/DE_Syn_PUF.html

'Chest pain, unspecified', 'Unspecified chest pain', 'miscellaneous GI agents','otic preparations', 'sulfonamides', 'vitamins and minerals' (lift is 17)

Table 4: A co-occurrence with 'phenotype cardinality' of 6 and lift of 17

50 candidate phenotypes that Marble generated were then annotated by two domain experts in a manner identical to above. We combined the 30 Rubik-generated candidate phenotypes with the 50 Marble-generated candidate phenotypes and used the resulting set of 80 candidate phenotypes in the co-occurrence experiment. Of these 80 phenotypes, the annotators found that approximately 14% are clinically meaningful, 78% are possibly significant and 8% are not clinically meaningful.

4.2 Results

We made the assumption that co-occurrences with higher phenotype cardinality should be favored (i.e., the more of the phenotype is represented, the more can be said about the significance of the phenotype as a whole). To serve this preference, we counted co-occurrences with the largest phenotype cardinality first, and then ignored any co-occurrence that was a subset of any larger co-occurrence. This choice allows co-occurrences of any size to contribute to the average of a phenotype group but favors interactions including more phenotypes of the group, assuming this is a better representation of the significance of the group as a whole.

Each combination of phenotypic items from a particular clinical group (for which a co-occurrence in PubMed was found) was counted if it was not a subset of a larger combination. This was achieved by simply ordering the combinations in descending cardinality order, and greedily inserting the combination into a set if it was not a subset of an already counted combination. This way, combinations with larger phenotype cardinality are preferred, but their subsets are not double counted. Then, for each phenotype, the average (normalized) lift in the counted set is calculated.

First, we look at the normalized lifts generated by the high-throughput methods. Figure 7 shows the normalized lift average of the phenotypes generated by Marble and Rubik ([8, 9, 20]) using 6 n-grams to represent every phenotypic item. Figure 8 shows the precision, recall and F1 score for classifying phenotypes to their "significant" or "not significant" annotator labels using the optimal normalized lift threshold. 6 n-grams was chosen to provide classification with the best balance between precision and recall, achieving an F1 score of 0.87 (2 N-grams scored 0.88, but had lower precision). Note the general trend of precision decreasing as more N-grams are used to represent phenotypic items.

Figure 9 shows the normalized lift average of the curated phenotypes, the items of which, again, are represented by 6 N-grams. Similarly, by choosing the optimal threshold, we are able to achieve 100% true negative classification, and 80% true positive classification, or an F1 score of 0.89.

We note that while lift thresholding classifies phenotypes with relative success in both high-throughput and curated

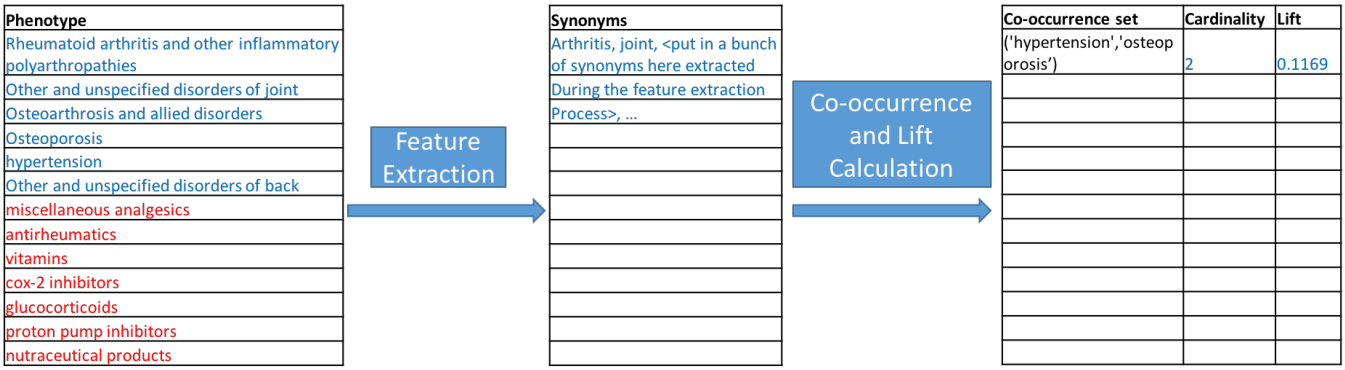

Figure 6: Example of lift calculation process

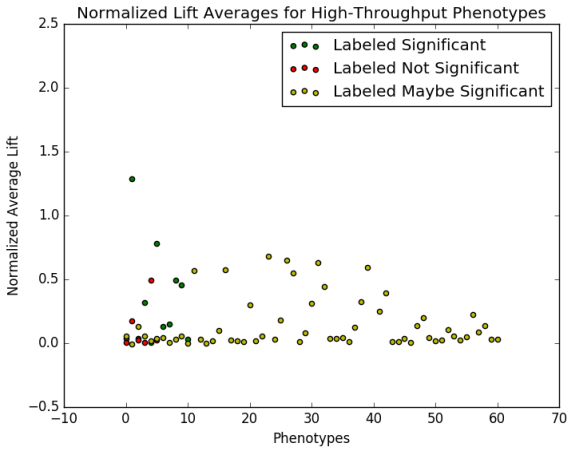

Figure 7: Normalized Average Lift of Marble/Rubik Phenotypes

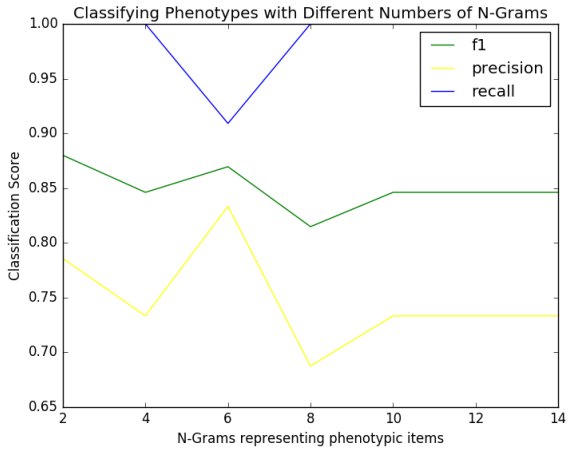

Figure 8: Classification Scores versus Maxium Number of N-grams Representing Phenotypic items

phenotypes, the method does not provide a universal threshold that works well for both. We can infer that phenotypes from different sources need to be grouped so that appropriate thresholds can be generated for each source of phenotypes. We also note that with the exception of a few outliers, the majority of phenotypes are very close to the optimal threshold. This suggests that further work is needed to improve the predictive value of lift thresholding.

5. CONCLUSION

We have presented an automated method for verifying the significance of phenotype groupings using co-occurrence of diagnoses/medications within the phenotype in a corpus of medical literature.

By representing phenotypes as a small set of relevant n-grams and calculating the lift of phenotype co-occurrences in PubMed, we were able to classify a small set of curated phenotypes with an F1 score of 0.89, and a set of phenotypes generated from EHR tensor data with an F1 score of 0.87. While this ground truth set is small, the method shows promise to provide an objective and automated method of verification for arbitrary phenotype groups.

Further, since the item co-occurrences are found in natural language, a set of sentences describing the phenotypic item co-occurrence can be reported and synthesized into a human readable explanation for the significance of the phenotype. Previous work [14] has shown that annotators produce better annotations in less time when starting from pre-annotated results from automatic tools. This implies that in addition to corroborating human annotation, automatic labeling can be used to facilitate the annotation process.

Work to further verify and improve this method is merited, as a reasonably high level of classification accuracy was achieved without complex feature selection, or using co-occurrences from the remaining 75% of available PubMed articles. With these additions, the method could further help improve phenotype annotation quality.

6. REFERENCES

[1] M. R. Boland, Z. Shahn, D. Madigan, G. Hripcsak, and N. P. Tatonetti. Birth month affects lifetime disease risk: a phenome-wide method. *Journal of the American Medical Informatics Association*, page ocv046, 2015.

[2] R. J. Carroll, A. E. Eyler, and J. C. Denny. Naive electronic health record phenotype identification for

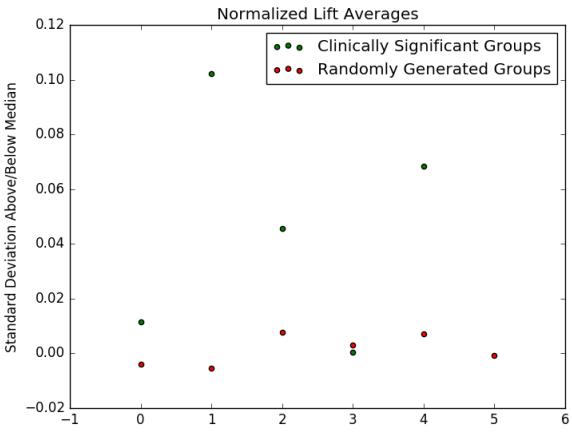

Figure 9: Normalized Average Lift of Curated Phenotypes

rheumatoid arthritis. In *AMIA Annu Symp Proc*, volume 2011, pages 189–96, 2011.

[3] Y. Chen, R. J. Carroll, E. R. M. Hinz, A. Shah, A. E. Eyler, J. C. Denny, and H. Xu. Applying active learning to high-throughput phenotyping algorithms for electronic health records data. *Journal of the American Medical Informatics Association*, 20(e2):e253–e259, 2013.

[4] K. Dickersin. The existence of publication bias and risk factors for its occurrence. *Jama*, 263(10):1385–1389, 1990.

[5] P. J. Easterbrook, R. Gopalan, J. Berlin, and D. R. Matthews. Publication bias in clinical research. *The Lancet*, 337(8746):867–872, 1991.

[6] M. Frisch, B. Klocke, M. Haltmeier, and K. Frech. Litinspector: literature and signal transduction pathway mining in pubmed abstracts. *Nucleic acids research*, 37(suppl 2):W135–W140, 2009.

[7] M. G. Hayes, L. Rasmussen-Torvik, J. A. Pacheco, W. K. Thompson, L. L. Armstrong, J. C. Denny, P. L. Peissig, A. W. Miller, W.-Q. Wei, S. J. Bielinski, C. G. Chute, C. L. Leibson, D. R. Crosslin, C. S. Carlson, K. M. Newton, W. A. Wolf, R. L. Chisholm, and W. L. Lowe. Use of diverse electronic medical record systems to identify genetic risk for type 2 diabetes within a genome-wide association study. *Journal of the American Medical Informatics Association*, 19(2):212–218, Mar. 2012.

[8] J. C. Ho, J. Ghosh, S. R. Steinhubl, W. F. Stewart, J. C. Denny, B. A. Malin, and J. Sun. Limestone: High-throughput candidate phenotype generation via tensor factorization. *Journal of Biomedical Informatics*, 52:199–211, Dec. 2014.

[9] J. C. Ho, J. Ghosh, and J. Sun. Marble: High-throughput phenotyping from electronic health records via sparse nonnegative tensor factorization. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 115–124, 2014.

[10] G. Hripcsak and D. J. Albers. Next-generation phenotyping of electronic health records. *Journal of the American Medical Informatics Association*, 20(1):117–121, 2013.

[11] C. Hu, P. Rai, C. Chen, M. Harding, and L. Carin. Scalable bayesian non-negative tensor factorization for massive count data. In *Machine Learning and Knowledge Discovery in Databases*, pages 53–70. Springer, 2015.

[12] L. J. Jensen, J. Saric, and P. Bork. Literature mining for the biologist: from information retrieval to biological discovery. *Nature reviews genetics*, 7(2):119–129, 2006.

[13] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient Estimation of Word Representations in Vector Space. *arXiv.org*, Jan. 2013.

[14] A. Névéol, R. I. Doğan, and Z. Lu. Semi-automatic semantic annotation of pubmed queries: a study on quality, efficiency, satisfaction. *Journal of Biomedical Informatics*, 44(2):310–318, 2011.

[15] NIH Health Care Systems Research Collaboratory. Rethinking Clinical Trials: A Living Textbook of Pragmatic Clinical Trials. July 2014.

[16] S. Pletscher-Frankild, A. Pallejà, K. Tsafou, J. X. Binder, and L. J. Jensen. Diseases: Text mining and data integration of disease–gene associations. *Methods*, 74:83–89, 2015.

[17] D. K. Rajpal, X. A. Qu, J. M. Freudenberg, and V. D. Kumar. Mining emerging biomedical literature for understanding disease associations in drug discovery. *Biomedical Literature Mining*, pages 171–206, 2014.

[18] J. M. Stern and R. J. Simes. Publication bias: evidence of delayed publication in a cohort study of clinical research projects. *Bmj*, 315(7109):640–645, 1997.

[19] M. Von Korff, B. Deffarges, and T. Sander. Data mining in medline for disease-disease associations via second order co-occurrence. In *Computational Intelligence, 2015 IEEE Symposium Series on*, pages 314–321. IEEE, 2015.

[20] Y. Wang, R. Chen, J. Ghosh, J. C. Denny, A. Kho, Y. Chen, B. A. Malin, and J. Sun. Rubik: Knowledge guided tensor factorization and completion for health data analytics. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1265–1274. ACM, 2015.

[21] H. Wasserman and J. Wang. An applied evaluation of snomed ct as a clinical vocabulary for the computerized diagnosis and problem list. In *AMIA Annual Symposium Proceedings*, volume 2003, page 699. American Medical Informatics Association, 2003.

[22] S. Yu, K. P. Liao, S. Y. Shaw, V. S. Gainer, S. E. Churchill, P. Szolovits, S. N. Murphy, I. S. Kohane, and T. Cai. Toward high-throughput phenotyping: unbiased automated feature extraction and selection from knowledge sources. *Journal of the American Medical Informatics Association*, 22(5):993–1000, Apr. 2015.