Information Retrieval for Genetic Mutations and Diseases

by

Erica Guo

Submitted to the Department of Computer Science
in partial fulfillment of the requirements for the degree of

Bachelor of Science in Computer Science

at

BROWN UNIVERSITY

May 2020

Author .....
Department of Computer Science
May 22, 2020

Certified by .....
Carsten Eickhoff
Advisor
Thesis Supervisor

Certified by .....
Jeff Huang
Reader
Thesis Supervisor

Accepted by .....
Thomas W. Doeppner Jr.
Director of Undergraduate Studies

Information Retrieval for Genetic Mutations and Diseases

by

Erica Guo

Submitted to the Department of Computer Science
on May 22, 2020, in partial fulfillment of the
requirements for the degree of
Bachelor of Science in Computer Science

Abstract

There is a vast amount of information online regarding genetic mutations and diseases that can be found within public genomic databases. My peers at the Warren Alpert Medical School of Brown University suspected that information within these databases could be utilized to uncover diseases eligible for genetic editing. They believed that allowing users to query for the following characteristics - type of inheritance, type of penetrance, and type of mortality - can expedite the process of finding diseases that are suitable candidates for genetic editing. However, even relatively comprehensive and regularly maintained databases such as Online Mendelian Inheritance in Man (OMIM) or GeneReviews do not easily allow people to query diseases using these characteristics. Therefore, I developed a resource containing information from GeneReviews that allowed users to identify genetic disorders using these characteristics.

Thesis Supervisor: Carsten Eickhoff
Title: Advisor

Thesis Supervisor: Jeff Huang
Title: Reader

Acknowledgments

First, I would like to thank my primary thesis supervisor, Professor Carsten Eickhoff, and my reader, Professor Jeff Huang, for providing valuable feedback and for making this project possible and tangible. I would like to thank Professor Philip Gruppuso for providing informative background on the properties of genetic disorders. I would also like to thank Professor Eli Adashi for providing recommendations on the future scope of the project. Finally, I would like to thank my family and friends for their feedback and encouragement.

Contents

1 Introduction **6**

1.1 Criteria for Gene Editing Candidates 6

1.1.1 High Mortality 6

1.1.2 High Penetrance 7

1.1.3 Autosomal Dominant Inheritance 8

1.2 Discovering New Gene Editing Candidates 9

2 Related Work **10**

2.1 Overview of Public Genomic Databases 10

2.1.1 Genetic Testing Registry 10

2.1.2 ClinVar 12

2.1.3 OMIM 14

2.1.4 GeneReviews 16

3 Method **17**

3.1 Designing A Solution 17

3.1.1 Inputs and Outputs 17

3.1.2 The Application 19

3.2 Information Extraction 21

3.2.1 Where should the relevant information come from? 21

3.2.2 Extracting Relevant Information 21

3.2.3 Structuring Information for Retrieval 21

3.3 Information Retrieval 22

3.3.1 Constructing Queries from User Input 22
3.3.2 Scoring Documents 23

4 Future Steps **23**
4.1 Performance Metrics 24
4.1.1 Precision 24
4.1.2 Recall 24
4.2 Trade-offs Between Performance Metrics 25
4.3 Improving Performance Metrics 25
4.3.1 Adding sources in information extraction 25
4.3.2 Changing the terms and phrases within information extraction 26
4.3.3 Changing the terms and phrases within a query 26
4.3.4 Boosting certain terms and phrases within a query 26
4.3.5 Boosting a field 26

5 Conclusion **27**

Chapter 1: Introduction

1.1 Criteria for Gene Editing Candidates

It has been proposed that gene editing be used to treat or prevent diseases like cancer and asthma [6]. To treat or prevent diseases, gene editing "makes corrections to patient genomes" to "prevent, stop, or reverse disease" [6]. According to my peers at the Warren Alpert Medical School, the diseases that are suitable for gene editing have the following characteristics: High mortality, high penetrance, and/or autosomal dominant inheritance. I will explain the rationale behind this assertion in the following sections.

1.1.1 High Mortality

Generally, diseases that severely impact on a patient are of more immediate concern and therefore are more acceptable candidates for gene editing. Gene editing is considered acceptable if it can reduce "the risk of serious disease or disability in a prospective child"; however, the use of gene editing on "less serious conditions" generates "significant public discomfort" [16]. The consequences of a disease are considered severe enough to permit gene editing if they substantially lower the quality of life or make it likely for the patient to die.

An example of disease with consequences that significantly lowers the quality of life would be LCA10, or Leber's congenital amaurosis 10, which causes hereditary blindness [8]. LCA10 was a target for experimental trials involving a new gene editing technique called Clustered Regularly Interspaced Short Palindromic Repeats

(CRISPR–Cas9) [8]. Although blindness is not life-threatening, it can inconvenience an individual to the extent that genetic editing is considered appropriate treatment.

Among severe consequences, the ones that are considered the most concerning are ones that are life-threatening. As a result, disorders that have life-threatening consequences require more immediate attention and therefore are more likely to be popular targets for gene editing. Life-threatening consequences include cardiac arrest, renal failure, and liver failure. For example, familial dilated cardiomyopathy is an inherited disease that prevents the heart from pumping blood efficiently, often resulting in heart failure [2]. Because of heart failure is fatal, there is a critical need for interventions like gene therapy as a form of treatment [14]. As a result, familial dilated cardiomyopathy is a popular target for gene editing techniques such as CRISPR-Cas9 and Transcription Activator-like Effector Nucleases (TALENS) [9].

To summarize, disorders with a higher likelihood of fatal consequences have a high critical need for gene editing. Therefore, they would be better candidates for gene editing.

1.1.2 High Penetrance

If a genetic disorder has high penetrance, a high percentage of patients with the genetic mutation specific to that disorder will display phenotypal characteristics [5]. Likewise, if it has low penetrance, a low percentage of patients will display that phenotype.

Gene editing is more appropriate for disorders with easily observable effects. Therefore, disorders with high penetrance are better candidates for gene editing. If the genes of a patient are edited with the intention of treating a disorder known to have high penetrance, descendants of that patient that inherit the edited genes will more readily display observable differences in their phenotype.

1.1.3 Autosomal Dominant Inheritance

Disorders can be inherited differently. There are several patterns of inheritance: autosomal dominant, autosomal recessive, X-linked dominant, X-linked recessive, and mitochondrial [?].

Much work has been published in the last two years on new gene editing techniques, indicating that the disorders that gene editing is applicable to may not be as limited as was previously anticipated. For instance, mitochondrial DNA mutations, many of which are single-gene mutations, are amenable to a different technology – mitochondrial replacement. Mitochondrial replacement is a "recent technological development" that "[eliminates] the transmission of undesired defective mitochondria" for individuals with heritable mitochondrial diseases [12].

Furthermore, other single-gene disorders, such as Huntington disease, an autosomal dominant degenerative neurologic disease, are caused by a greater than normal number of so-called tri-nucleotide (CAG) repeats [7]. Prior molecular methods for gene editing were not suited to treating such a disorder, but newer methods are.

That being said, single-gene disorders remain easier candidates to target for gene editing. Single-gene disorders are more "easily tracked through families and the risk of them occurring in later generations can be predicted" [16]. On the other hand, disorders that are not caused by a single faulty gene are less fixable using the technologies currently under development. Current technology is stretched to its limits if aimed at correcting disorders resulting from insertion or deletion of large segments of DNA.

The questions remains: Among single-gene disorders, which are suitable candidates for gene editing based on severity and penetrance?

First, autosomal recessive mutations can be candidates for gene editing. However, many cases of recessive disorders result from abnormal genes inherited from the two parents with different mutations [17]. To correct autosomal recessive disorders, both genes would have to be corrected. On the other hand, if a disorder is autosomal dominant, then only one mutated gene from a parent is sufficient to pass on the

disorder. Therefore, it is more straightforward to correct one mutation than it is to correct two mutations.

Second, it is possible to conduct gene-editing on X-linked mutations. However, the need to do gene editing would be dependent on the sex of the offspring [17]. If the offspring is male, they would receive one X-chromosome from the mother and one Y-chromosome from the father; therefore, they would be affected if there was a mutation within the X-chromosome regardless of whether the mutation was dominant or recessive. If the offspring is female, they would only be affected by an X-linked mutation if it is a dominant mutation. The effect of the sex of the offspring complicates the process of detecting changes in gene editing.

Due to the current technical and analytical constraints on gene editing, autosomal dominant mutations are the most straightforward candidates for gene editing.

1.2 Discovering New Gene Editing Candidates

Medical researchers and academics at the Warren Alpert Medical School of Brown University have expressed interest in discovering novel candidates for gene editing.

Severe consequences, autosomal dominant inheritance, and/or high penetrance characterize diseases that are well-suited for gene editing. Given that public databases contain information about such characteristics for a multitude of disorders, it may be possible to discover new candidates for gene editing by allowing users to better query these characteristics.

Currently available genomic databases do not allow individuals to query these characteristics in a user-friendly way. This work aims to build an application that can serve as a public resource so that medical researchers can more easily discover

candidates for gene editing.

Chapter 2: Related Work

Even the most comprehensive, publicly available genomic databases pose enormous difficulties for those who wish to query for gene editing candidates based on relevant characteristics such as severe consequences, autosomal dominant inheritance, and/or high penetrance.

2.1 Overview of Public Genomic Databases

Most free, publicly available genomic databases contain information in the form of articles, each pertaining to a single gene or the implicated gene(s) of a single genetic disorder. There are over 2,000 databases, but there are no established standards for the content or formatting of those databases [15].

After surveying several databases, I have concluded that the most comprehensive and respected databases within the medical community include Genetic Testing Registry, ClinVar, OMIM, and GeneReviews. The following sections describe the difficulties of using these databases to query for characteristics relevant to gene editing.

2.1.1 Genetic Testing Registry

Genetic Testing Registry is largely focused on diagnostic tests [3]. If you search for a genetic disorder such as "hypoglycemia," you are brought to links to tests, conditions, genes, and and the laboratories involved in aforementioned tests (see Figure 2-1).

Clicking on one of the results leads to a page with fairly sparse information (see Figure 2-2). At best, the page will contain some information about a given disorder's

inheritance. However, rarely is there information about penetrance or mortality.

Therefore, it is not feasible for users to query Genetic Testing Registry for characteristics relevant to gene editing due to the sparse content within its articles.

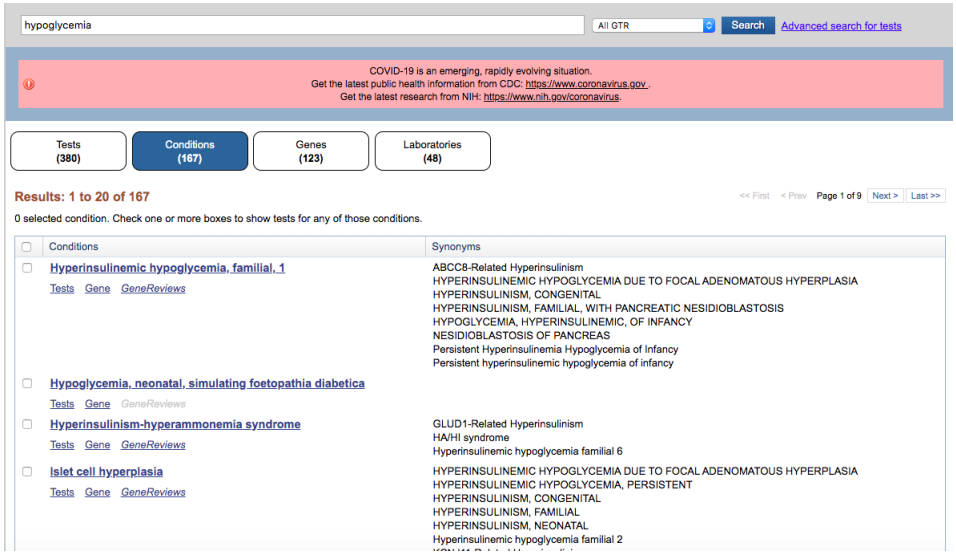

Figure 2-1: Search Results within Genetic Testing Registry

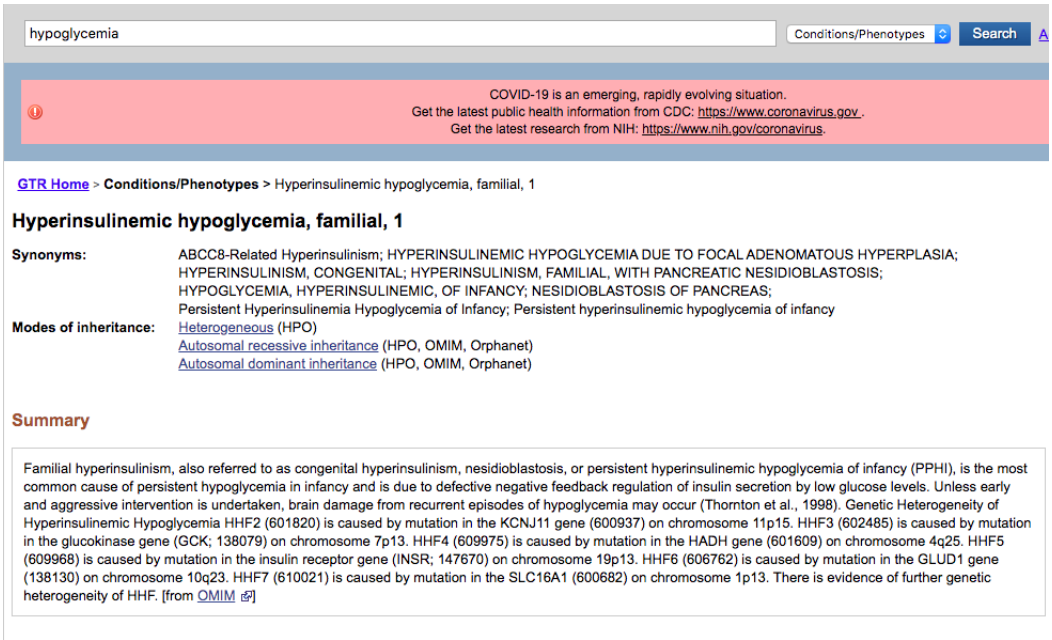

Figure 2-2: A Result Page within Genetic Testing Registry

2.1.2 ClinVar

ClinVar aggregates information about genetic variations and their relationships to human health [18]. There are approximately 6,000 entries, all of which are classified with regard to their clinical manifestations. Querying for certain characteristics will bring you to a list of search results, each of which describes a variant (see Figure 2-3)).

If you click on a variant, it brings you to tabular information, including links to GeneReviews and OMIM (for well characterized-disorders) (see Figure 2-4)).

ClinVar's submissions vary too greatly in complexity to be used as a source of information for users to query characteristics. Some submissions are a simple description of an "allele and its interpretation" 2-6)), while some provide "experimental evidence" regarding "the effect of the variation" on human health (see Figure 2-5)) [18]. As a result, if a user queries for diseases with a certain characteristic, ClinVar will likely fail to capture the breadth of diseases that have that characteristic.

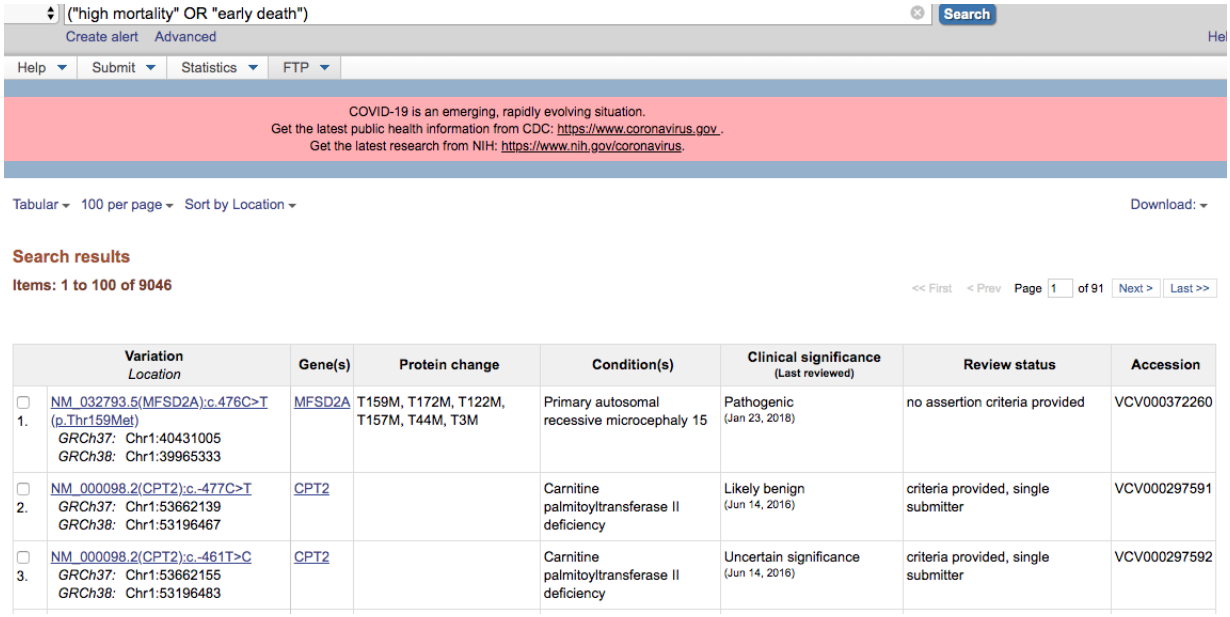

Figure 2-3: Search Results within ClinVar

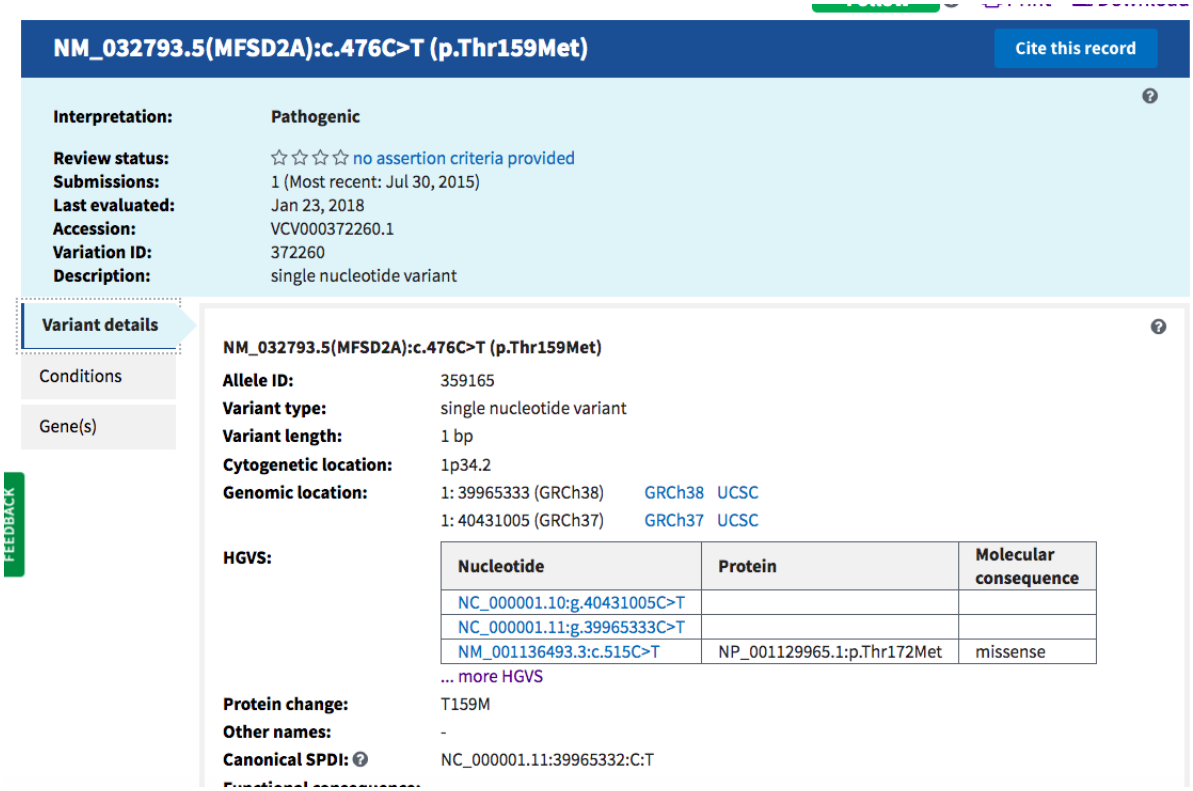

Figure 2-4: Result Page within ClinVar

Figure 2-5: Detailed Submission from ClinVar

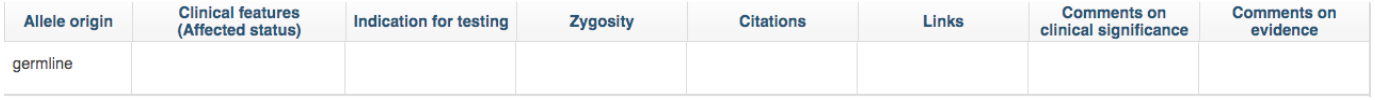

Figure 2-6: Meager Submission from ClinVar

2.1.3 OMIM

OMIM is intended to be a comprehensive compendium of human genes and genetic phenotypes. It is freely available and updated daily. OMIM contains full-text, referenced overviews of all known Mendelian disorders and over 15,000 genes, focusing on the relationship between phenotype and genotype [10]. My inspection revealed that there is no standardized format for entries. Phenotype-genotype information is provided in a tabular format at the top of each entry.

There is a search engine in OMIM that one can use in addition to filters. However, the search engine lacks the appropriate filters that allow users to query for results with a certain type of penetrance, inheritance, or mortality. For example, if a user wishes to search for disease with high mortality, OMIM's search engine lacks a filter that queries for this characeristic.

From a user experience perspective, this makes filters unreliable. A user may take time attempting to look through the filters only to discover that filters don't work, making the process more inefficient.

Because users cannot use filters to expedite their search, they are forced to hand-write queries. As a result, users may write queries that inaccurately capture the characteristics that they are trying to look for.

To construct a query, the user needs to think of all possible keywords and phrases that could indicate the characteristic that they are querying for. This is because the search engine does not automatically search for related keywords or phrases that may be related what the user inputs.

For example, if they want to search for diseases with high mortality, it would not be sufficient to search "high mortality", because OMIM does not automatically search for documents with related keywords (see Figure 2-7). Therefore, it is necessary for the user to be creative and to think of the different ways that the characteristic "high mortality" could be conveyed within articles. "Early death" and "high death rate" are phrases that might indicate high mortality, but they would not be captured by the search term "high mortality." As a result, merely searching the keyword "high

mortality" would not fully capture the diversity of phrases that might mean high mortality.

A user can attempt to remedy this by stacking more phrases and terms within a query using boolean operators such as "OR" or "AND", which might allow them to attain more relevant search rules (see Figure 2-8). However, even if a user attempts to expand their query to capture more relevant terms or phrases, they could incorrectly include certain phrases or terms that cause false positives. For example, if they include the term "death" in their query, search results containing "cell death" might show up. These results would contain false positives because "cell death" does not actually indicate the characteristic of "high mortality."

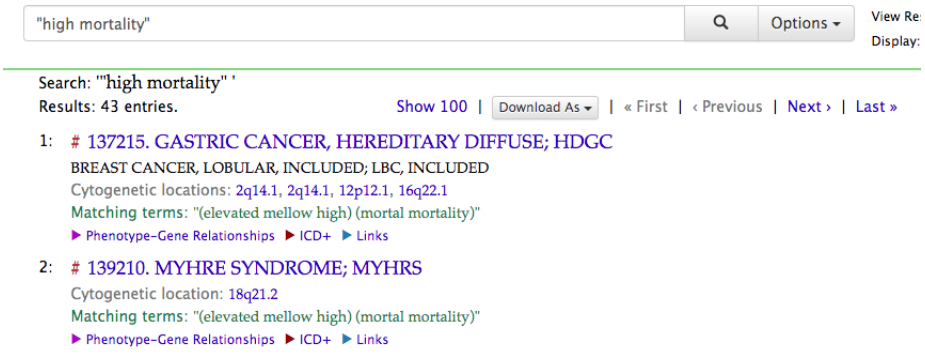

Figure 2-7: Limited Search Query within OMIM

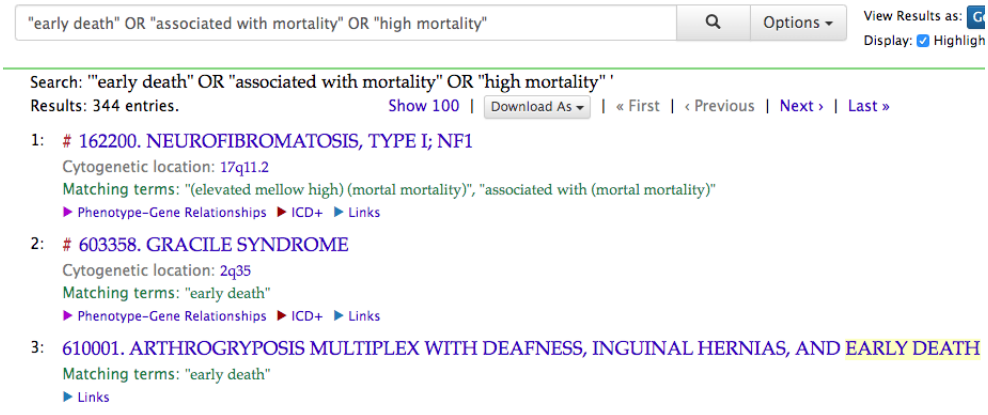

Figure 2-8: More Expansive Search Query within OMIM

2.1.4 GeneReviews

GeneReviews is an "international point-of-care resource for busy clinicians" that provides "clinically relevant and medically actionable information for inherited conditions" [1]. GeneReviews currently contains over 700 entries, each of which focuses on a single gene or phenotype that results from single-gene disorders.

According to my investigation, GeneReviews contains relatively comprehensive information about penetrance, mortality, and mode of inheritance. That being said, like OMIM, GeneReviews lacks appropriate filters that should allow users to query for these characteristics.

As a result, users have to handwrite the query (see Figure 2-9). Users can capture a wider range of relevant results by including more keywords and phrases in their search (see Figure 2-10), but it may still be possible that their query is constructed erroneously.

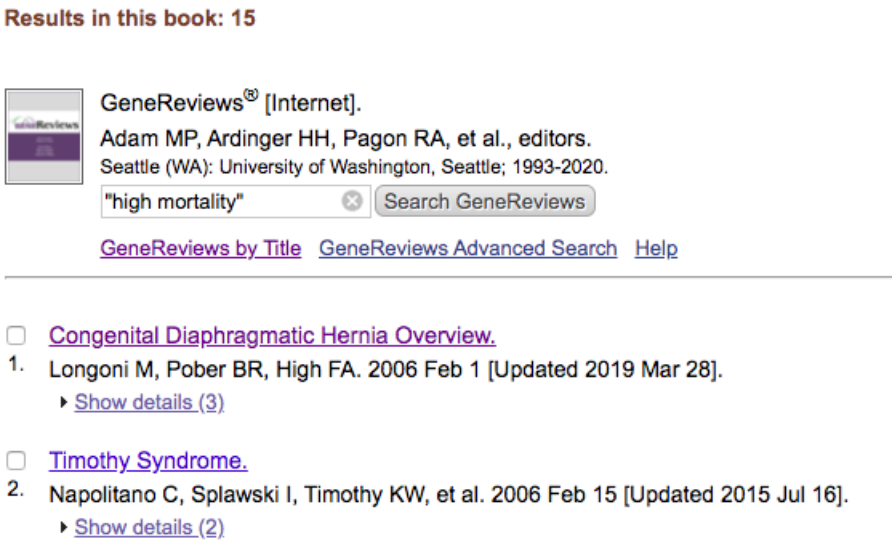

Figure 2-9: Limited Search Query within GeneReviews

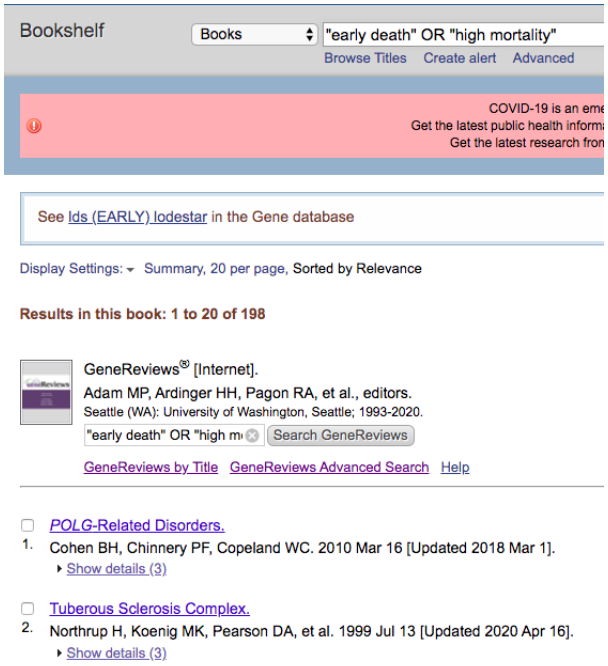

Figure 2-10: Expanded Search Query within GeneReviews

Chapter 3: Method

3.1 Designing A Solution

3.1.1 Inputs and Outputs

Identifying candidates for gene editing remains the primary purpose. Although potential uses extend beyond identifying gene editing candidates, I designed a resource that could also be used beyond identifying gene editing candidates.

As a result, I created an application that can allows users to filter by mode of inheritance, level of mortality, and type of penetrance, beyond characteristics specific to gene editing candidates.

My application could be useful with varied inputs in the analysis of genetic disorders in healthcare. For example, consider a healthcare professional who is caring for a child who has severe seizures. Family history shows that his father, siblings, and father's relatives have inconsistent symptoms. Some have even suffered early death. This healthcare professional can use the application to identify genetic disorders that display autosomal dominant inheritance, low penetrance, and high mortality. The application can provide a list of diseases that have these characteristics in order to assist the healthcare professional in diagnosing the hereditary disease. Therefore, the application should include multiple types of penetrance as inputs, not only high penetrance.

Similarly, if a healthcare professional wants to query disorders that they suspect are autosomal recessive or low mortality, an application should present characteristics that include different modes of inheritance and different levels of mortality.

Heredity across species is another areas in which users might find it valuable to input multiple modes of inheritance, mortality, and penetrance when querying diseases. Suppose a genetic scientist is studying a certain gene. He deletes that gene in mice. Their descendants inherit two mutated genes, one from each parent. A significant percentage of descendants die, indicating high mortality, but he's not sure if he can conclude that there is a high penetrance. The genetic scientist can use the application to identify genetic disorders that are autosomal recessive, display high mortality, and unknown penetrance. The application can provide diseases that will help him identify genetic disorders in humans that display a similar phenotype to identify defects attributable to that gene.

As a result, I decided to include the following options for a mortality filter: high mortality, low mortality, and ignore mortality. The options for the penetrance filter are high penetrance, low penetrance, variable penetrance, and ignore penetrance. The options for inheritance are autosomal dominant, autosomal recessive, and ignore inheritance.

3.1.2 The Application

I created an application from NodeJS and Python. The user starts up the app, selects the inputs given the radio buttons, and clicks the search button (see Figure 3-1). They wait for the results to appear while the app displays a message asking them to wait (see Figure 3-2).

Finally, the search results appear, displaying the disorder ID, name, and relevance score (which will be discussed later) (see Figure 3-3). Each disorder ID is hyperlinked, allowing the user to click on a disease they find interesting in order to investigate. A maximum of 150 results are displayed. The results are ranked in order of relevance score, from highest to lowest. The user can query again by entering different inputs and clicking "search."

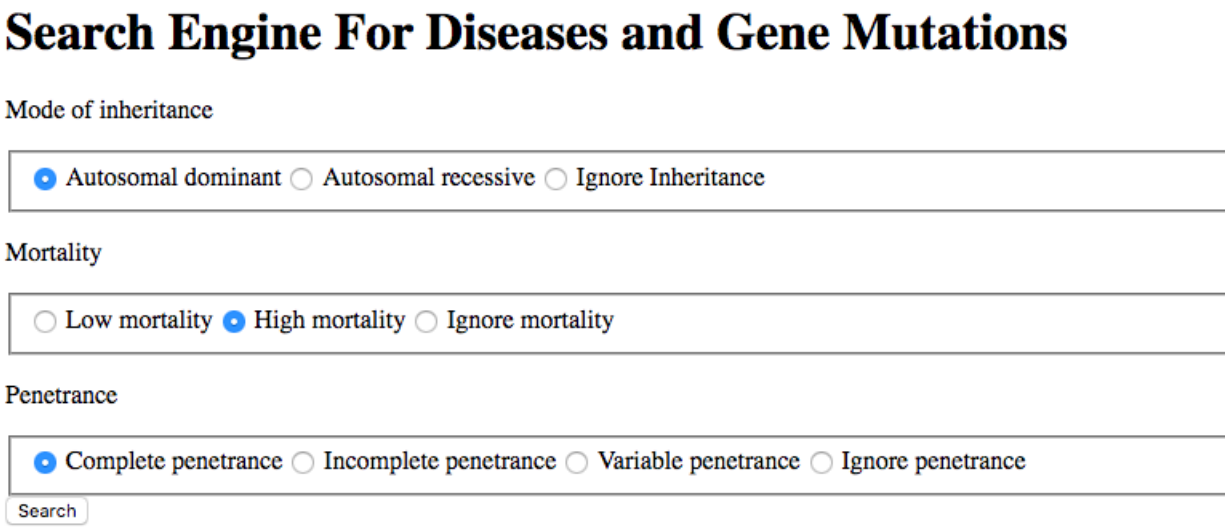

Figure 3-1: User Selects Inputs

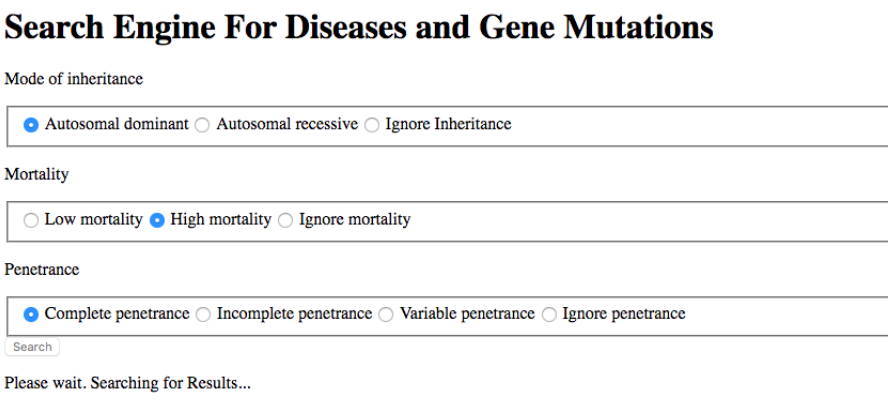

Figure 3-2: Waiting for Search Results

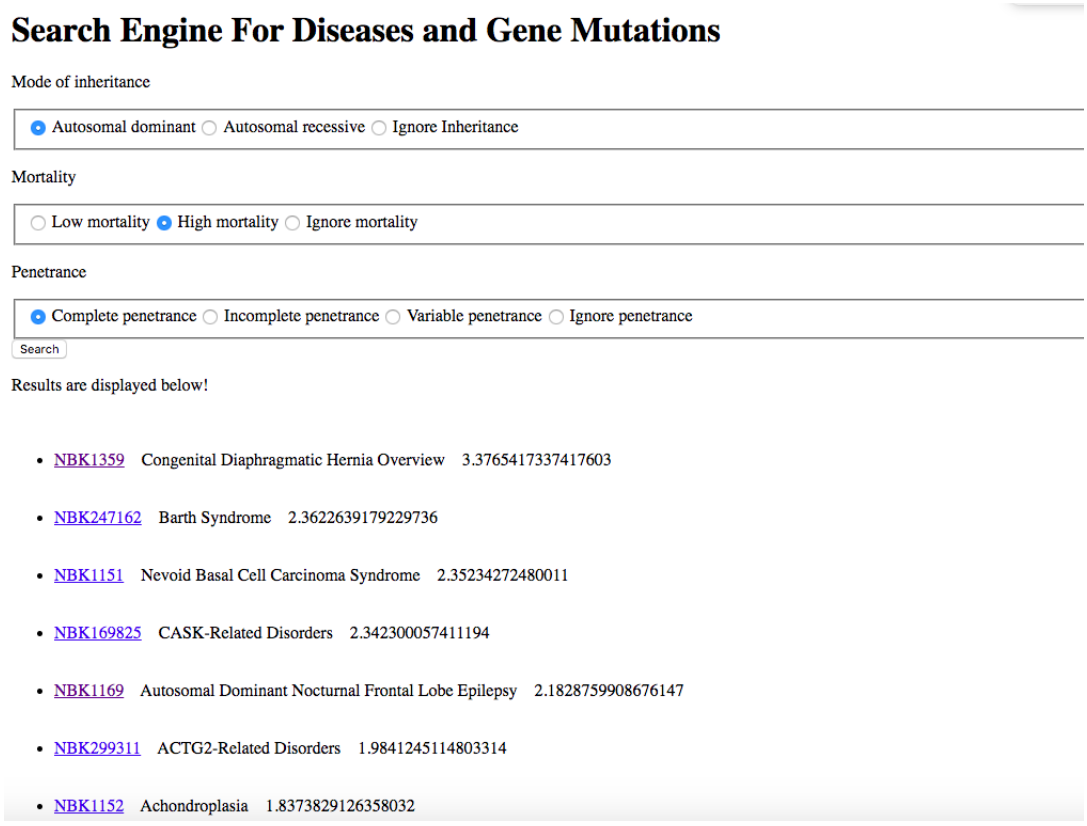

Figure 3-3: Search Results

3.2 Information Extraction

3.2.1 Where should the relevant information come from?

Before I created a search engine that allowed users to query for relevant information, I had to first decide what the source of that information would be. Based on my investigation, the two most comprehensive databases that would have information about different modes of inheritance, penetrance, and mortality are OMIM and Gene Reviews. Furthermore, these two databases are relatively more comprehensible than others in terms of how much information on each gene mutation or disease articles contained.

I decided to use GeneReviews as a source of information, because its articles are generally longer and more comprehensive than OMIM's. In OMIM, articles that are only one paragraph are common. Some articles in OMIM have no information on inheritance. The majority of shorter entries don't address questions of penetrance or mortality.

3.2.2 Extracting Relevant Information

First, I scraped the text from every GeneReviews article using BeautifulSoup. (Each GeneReviews article discusses a different genetic disorder or disease.) Then, in collaboration with peers at the Brown Medical School, I inferred the most common keywords and phrases that signified an article contained information regarding inheritance, penetrance, and mortality. Sentences that contained a matching keyword or phrase were stored in files with a unique id that corresponded to the GeneReviews article they were found in.

3.2.3 Structuring Information for Retrieval

For every GeneReviews article, a JSON object was created with the fields "id", "name", "mortality", "penetrance", and "inheritance". The "mortality", "penetrance", and "in-

heritance" fields contain sentences from the files that were written to earlier. For every JSON object, PyLucene creates a document. This process is called "indexing", or making the information searchable. PyLucene is a Python extension for accessing Java Lucene. Its goal is to allow you to use PyLucene's text indexing and searching capabilities from Python. It is recommended for its flexible search engine functionality than Apache Solr, its alternative.

Essentially, one document corresponds to one JSON, which corresponds to one gene article scraped from Gene Reviews.

For every JSON object in the inputted list, a document with fields "id", "name", "mortality", "penetrance", and "inheritance" is created. All sentences related to a given field (penetrance, inheritance, or mortality) are joined together and added to that field.

3.3 Information Retrieval

3.3.1 Constructing Queries from User Input

Suppose a user selects an input in the application such as "high mortality". Consequently, the backend has an existing query corresponding to that input. In PyLucene, a query is a series of phrases and terms [11].

The query has been constructed from common phrases and terms in GeneReviews that indicates a genetic disorder or disease has that characteristic. As a result, the user does not have to construct a burdensome query in GeneReview's currently existing search engine. For example, the likely phrases that indicate "high mortality" are "increases risk of death", "early death", and "can result in death".

PyLucene uses boolean logic in the query specification so that it can score documents that contain at least one of these phrases and terms [13].

3.3.2 Scoring Documents

Suppose the user enters three inputs: autosomal dominant, low penetrance, and low mortality. There will be three queries that correspond to each of these inputs, respectively. PyLucene scores documents with respect to each query. Then, it creates a relevance score fore each document.

PyLucene uses the Vector Space Model (VSM) of information retrieval to determine the relevance of given document that matches keywords or phrases in a query [13]. The more relevant a document is, the higher the score it receives relative to a query. In VSM, the more frequently a term or phrase within the query appears in a given document relative to the number of times the term or phrase appears in all the documents in the collection, the higher the score the document receives with respect to the query.

Then, the scores for all documents for each search field is normalized with respect to each other. For one document, the scores for each field with respect to that document that is summed into a single relevance score. Finally, PyLucene ranks documents from highest to lowest by relevance score. The top 150 documents that have higher relevance scores are the ones displayed as search results for the user.

Chapter 4: Future Steps

In the future, we can gather and utilize performance metrics collected from user testing in order to improve the overall relevance of search results.

4.1 Performance Metrics

4.1.1 Precision

One performance metric is precision, or the fraction of relevant instances among the retrieved instances. We can measure precision of the results returned by the search engine by asking the users to report how many of the top N results, or top N documents, were actually diseases that fit the characteristics in the search query.

The denominator within our precision metric represents the top N results. We can adjust the denominator, say, if instead of measuring the precision of the top 10 results we want to measure the precision of the top 20.

When measuring precision, keep in mind the specialized nature of our users' professions. A user might not be aware of every disease that show up in the results. We can measure precision by first asking them to point out the diseases they're aware of and how relevant they are to the search criteria, based off of our user's past knowledge. As for the results that the user is not aware of, the user will assess whether the disease has the query's characteristics by skimming the web-page linked to that disease on GeneReviews. The user will then infer whether the search engine has correctly inferred that the disease fits that result.

Adding together the number of results they recognize as relevant based on prior experience and the number of results that are relevant based on the user's inferences, we can then divide the sum by the top N results.

4.1.2 Recall

A second performance metric is recall. The recall metric is the number of expected results that are displayed divided by the total number of results expected. Given the search criteria, recall measures how well the search engine captured results that are expected to appear.

In order to accurately attain the total number of expected results, we would have to ask a geneticist how many diseases they would expect to have the characteristics

that were searched for. A geneticist can identify the characteristics of a much broader range of diseases than the average user. Most other prospective users do not know the genetic traits of a wide enough range of diseases.

4.2 Trade-offs Between Performance Metrics

There may be a trade-off between improving precision and improving recall. To improve recall, we can display more results. However, this may result in more false positives among the top results, decreasing precision. On the other hand, to improve precision, we can decrease the number of results retrieved. However, there may be less expected results displayed, decreasing recall.

Currently, the maximum number of results displayed in the application is 150 results. If there is indeed a trade-off between improving and recall, we can adjust the number of results displayed to the user depending on which users care more about - recall or precision. During user testing, it is necessary to evaluate the extent to which users care about precision as opposed to how much users care about recall.

4.3 Improving Performance Metrics

4.3.1 Adding sources in information extraction

We can draw information from additional sources, such as other databases, in order to improve the relevance and the precision. One potential idea is to extract information from OMIM, as OMIM contains more entries than GeneReviews. Perhaps there are important gene editing candidates that lie within OMIM's entries. As a result, more relevant results may appear overall and in the top results if we extract information not just from GeneReviews' articles but also from OMIM's entries.

4.3.2 Changing the terms and phrases within information extraction

If we become cognizant of more terms and phrases that signify that a GeneReviews article contains information about mortality, penetrance, or inheritance, we can capture more articles that are related to those characteristics. As a result, we can improve recall.

4.3.3 Changing the terms and phrases within a query

Once we discover more phrases and terms indicating that a certain disorder is likely to have a certain characteristic, we can change the phrases and terms within the query corresponding to that characteristic so that documents containing those relevant terms are eligible for scoring in the first place. This improves recall as more relevant results will show up.

4.3.4 Boosting certain terms and phrases within a query

We can also apply query boosting. In query boosting, documents containing certain phrases or terms within a given query will have a higher relevance score relative to other documents that have other phrases or terms within that query. If you boost a certain phrase or term, the document containing it will have its score multiplied by by a given boost factor. As a result, the search result represented by that document will rank higher. Overall, this can improve precision because documents containing more relevant terms within the query will rank higher.

4.3.5 Boosting a field

Field boosting multiples the scores of all documents with any given match to a query within a certain field by a boost factor specific to that field [4]. As a result, documents that have matches for a certain field will score higher relative to other documents.

Suppose we hypothesize it would improve precision to regard the characteristic "low penetrance" to be more important than the characteristic "autosomal dominant." Therefore, documents that match the query corresponding to "low penetrance" should have a higher relevance score than documents matching the query corresponding to "autosomal dominant." As a result, we can apply a boost to the field "low penetrance" so that a given document containing a match with the query corresponding to "low penetrance" has a higher score relative to documents that have a query match for "autosomal dominant."

Field boosting can improve precision. Given all the characteristics a user inputs, prioritizing certain characteristics over others will allow documents with those characteristics to rank higher, thereby improving precision.

Chapter 5: Conclusion

To better identify the potential universe of gene editing candidates, I built an application that allows users to filter by different types of characteristics pertaining to mode of inheritance, mortality, and penetrance.

A preliminary examination of the search results by collaborators in the Brown Medical School demonstrates that search criteria are appropriately recognized. Given fuller user evaluation, we can improve the way in which information extraction and retrieval are performed so that the overall results displayed to the user are more relevant.

Bibliography

[1] Adam MP, Ardinger HH, Pagon RA, et al., editors. GeneReviews® [Internet]. Seattle (WA): University of Washington, Seattle; 1993-2020. Available from: https://www.ncbi.nlm.nih.gov/books/NBK1116/.

[2] "Familial Dilated Cardiomyopathy." Genetics Home Reference, National Institutes of Health, 12 May 2020, https://ghr.nlm.nih.gov/condition/familial-dilated-cardiomyopathy.

[3] "Frequently Asked Questions (FAQ) for the NIH Genetic Testing Registry (GTR)." National Center for Biotechnology Information, U.S. National Library of Medicine, https://www.ncbi.nlm.nih.gov/gtr/docs/faq/.

[4] Gospodnetic, Otis, and Erik Hatcher. Lucene in Action. Greenwich, CT: Manning Publications, 2005. Internet resource.

[5] Griffiths AJF, Miller JH, Suzuki DT, et al. An Introduction to Genetic Analysis. 7th edition. New York: W. H. Freeman; 2000. Penetrance and expressivity. Available from: https://www.ncbi.nlm.nih.gov/books/NBK22090/.

[6] "How Is Genome Editing Used?" Genome.gov, National Institutes of Health, https://www.genome.gov/about-genomics/policy-issues/Genome-Editing/How-genome-editing-is-used.

[7] "Huntington Disease." Genetics Home Reference, National Institutes of Health, https://ghr.nlm.nih.gov/condition/huntington-disease.

[8] Ledford, Heidi. "CRISPR treatment inserted directly into the body for first time." Nature News, Nature Publishing Group, 5 Mar. 2020, https://www.nature.com/articles/d41586-020-00655-8.

[9] Ohiri, Joyce C, and Elizabeth M McNally. "Gene Editing and Gene-Based Therapeutics for Cardiomyopathies." Heart failure clinics vol. 14,2 (2018): 179-188. doi:10.1016/j.hfc.2017.12.006.

[10] "Online Mendelian Inheritance in Man (OMIM)." OMIM, https://omim.org/help/faq.

[11] "QueryParser (Lucene 7.3.1 API)." Apache Lucene, 14 May 2018, https://lucene.apache.org/core/7_3_1/queryparser/org/apache/lucene/queryparser/classic/QueryParser.html.

[12] Tachibana, Masahito, et al. "Mitochondrial Replacement Therapy and Assisted Reproductive Technology: A Paradigm Shift toward Treatment of Genetic Diseases in Gametes or in Early Embryos." Reproductive Medicine and Biology, John Wiley and Sons Inc., 19 Sept. 2018, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6194288/.

[13] "TFIDFSimilarity (Lucene 7.4.0 API)." Apache Lucene, 22 June 2018, https://lucene.apache.org/core/7_4_0/core/org/apache/lucene/search/similarities/TFIDFSimilarity.html.

[14] Tilemann, L., Ishikawa, K., Weber, T., & Hajjar, R. J. (2012). Gene therapy for heart failure. Circulation research, 110(5), 777–793. https://doi.org/10.1161/CIRCRESAHA.111.252981.

[15] Vihinen, Mauno, et al. "Guidelines for Establishing Locus Specific Databases." Wiley Online Library, John Wiley; Sons, Ltd, 9 Dec. 2011, onlinelibrary.wiley.com/doi/full/10.1002/humu.21646.

[16] "What Are Single Gene Disorders?" YourGenome, The Public Engagement Team at the Wellcome Genome Campus, 25 Jan. 2016, https://www.yourgenome.org/facts/what-are-single-gene-disorders.

[17] "What Are the Different Ways in Which a Genetic Condition Can Be Inherited?" Genetics Home Reference, National Institutes of Health, 12 May 2020, ghr.nlm.nih.gov/primer/inheritance/inheritancepatterns.

[18] "What Is ClinVar?" National Center for Biotechnology Information, U.S. National Library of Medicine, https://www.ncbi.nlm.nih.gov/clinvar/intro/.