EG_Gene_neighbors – concepts, triples - publications generation

The follow up activity for EntrezGene Gene_neighbors is to

- enrich existing gene concepts with concept measures (location information)
- generate triples from the gene_neighbors file with associated publications

Mapping collection.

First, the source file should be transferred to a mongo collection for further processing. Only add entries to the mongo collection if

existing: tax_id= 9606 AND <assembly>="Reference GRCh38 Primary Assembly" OR

new: tax_id = {**taxonomyID = {3702, 39947, 39946, 3708, 4577, 4113, 4006, 3983, 4565, 4081, 3712, 3711, 3707, 71323, 4513, 4558, 559292}**

September 2016 New: tax_id = taxonomyID= {10090, 10116, 7955, 8355, 7227, 6239, 4932, 562, 5061, 1488, 9823, 9913, 9915, 9925, 9940, 9031, 4530, 15368, 4565, 195583, 869827, 109376, 4084, 4111, 4072, 80379, 220668, 565040, 565042, 391904, 491077, 557433, 334390, 334413, 409438, 431946, 585535}

Expected final result overview

This source file will enrich concepts of ST=28 with additional measures, based on the following source file keys:

<chromosome>

<orientation>

<start_position>

<end_position>

<assembly>

This source file will result in 3 types of triples:

GeneID – adjacent_to - GeneIDs_on_left

GeneID – adjacent_to – GeneIDs_on_right

GeneID – location_of – overlapping_GeneIDs

Gene_neighbors is a tab-delimited file, with 1 line per entry. It is released/updated daily.

File location: ftp://ftp.ncbi.nih.gov/gene//DATA/gene_neighbors.gz

The file contains around 13M entries; however we only generate triples for entries for which the <assembly>="Reference GRCh38 Primary Assembly" ; For this set, the expected result is around 250k triples.

1) *Are data source schema's available? And if so, where can they be found? Is it possible to mark the fields we need to extract /use?*

Location of the schema:

ftp://ftp.ncbi.nih.gov/gene//README

It is a tab delimited file.

One line per GeneID and genomic placement

Column header line is the first line in the file. Genomic sequences in scope for reporting include all top-level sequences and curated genomic (NG_ accessions).

MODIFIED: May 21, 2007 to use '-' for empty fields. **When "-" is encountered, do not process the value!**

Key	Opt/Mand	Use	When absent	Value type	Modifications
tax_id	n/a	Not used	Continue	Number ; single value	
GeneID	M	Triple subject	Do not process triple	Number ; single value	Add prefix [geneid] ; kb= geneid
genomic_accession.version	n/a	Not used	Continue	string	
genomic_gi	n/a	Not used	Continue	Number; single value	
start_position	O	Concept measure	continue	number	
end_position	O	Concept measure	continue	Number	
Orientation	O	Concept measure	Continue	String	
Chromosome	O	Concept measure	Continue	String	
GeneIDs_on_left	O	Triple objects	Donot process triples	Multiple values; "|" separated	For each value: Add prefix [geneid] ; kb= geneid
Distance_to_left	O	Triple measure	continue	number	Assign only to first geneid_on_left
GeneIDs_on_right	O	Triple objects	Donot process triples	Multiple values; "|" separated	For each value: Add prefix [geneid] ; kb= geneid
Distance_to_right	O	Triple measure	Continue	Number	Assign only to geneid_on_right
Overlapping_ geneids	O	Triples objects	Donot process triples	Multiple values; "|" separated	For each value: Add prefix [geneid] ; kb= geneid

Assembly	M	CONCEPT MEASURE	Continue	String	

What we need to explain high level;

1) *How to find the subject term and associated measures*

The subject is the <GeneID> of the entry.

Solr query:
term: [geneid]<GeneID>

When the entry cannot be matched in Solr, donot process the entry and proceed to the next entry.

The following measures must be associated:

Source key	***Measure name***	***type***
<start_position>	*= “start position (0)”*	*number*
<end_position>	*= “end position (0)”*	*number*
<orientation>	*= “orientation”*	*string (value = “+” or “-“)*
<chromosome>	*= “chromosome”*	*string*
<assembly>	*= “assembly”*	*string*

2) *How to find the object term and associated measures (if applicable)*

There are 3 object keys (GeneIDs_on_left, GeneIDs_on_right; overlapping_geneIDs); each object key can have multiple values in the sourcefile, separated by “|” .
Triples need to be made with each value as object. So if a document contains 2 entries for each of the 3 object keys, the document generates 6 triples.

GeneIDs_on_left:

Solr query:
Term: [geneid]<GeneIDs_on_left>
When then entry cannot be matched in Solr, donot process the entry and proceed to the next entry.

Example:
Sourcefile:
<geneID>=6011
<geneIDs_on_left>=496|7027

Solrquery-subject: [geneid]6011
Solrquery-object1: [geneid]496

Solrquery-object2:[geneid]7027

Resulting in 2 triples

GeneIDs_on_right:

Solr query:

Term: [geneid]<GeneIDs_on_right>

When then entry cannot be matched in Solr, donot process the entry and proceed to the next entry.

Example:

Sourcefile:

<geneID>=6011

<geneIDs_on_right>=348013|100130386

Solrquery-subject: [geneid]6011

Solrquery-object1: [geneid]348013

Solrquery-object2:[geneid]100130386

resulting in 2 triples

Overlapping geneIDs

Solr query:

Term: [geneid]<overlapping_geneids>

When then entry cannot be matched in Solr, donot process the entry and proceed to the next entry.

Example:

Sourcefile:

<geneID>=4698

<overlapping_geneids>=142685|102724567

Solrquery-subject: [geneid]4698

Solrquery-object1: [geneid]142685

Solrquery-object2:[geneid]102724567

Resulting in 2 triples

3) How to find the triple mapping and associated measures (if applicable)

When object =<Geneids_on_left>, use “adjacent_to” as predicate

- Use <distance_to_left> as measure "distance", but ONLY FOR THE FIRST VALUE of the GeneIDs_on_left> key. The other values donot get a measure.

Example:

Solr query:

Term: [geneid]<GeneIDs_on_left>

Example:

Sourcefile:

<geneID>=6011

<geneIDs_on_left>=496|7027

<distance_to_left>=26896

leads to the following triples:

uuid([geneid]6011)- adjacent to – uuid([geneid]496) [distance=26896]

uuid([geneid]6011) – adjacent to – uuid([geneid]7027)

When object =<Geneids_on_right>, use "adjacent_to" as predicate

- Use <distance_to_right> as measure "distance", but ONLY FOR THE FIRST VALUE of the GeneIDs_on_left> key. The other values donot get a measure.

See example above.

When object =<overlapping_geneids>, use "location_of" as predicate

There is no measure for this triple.

4) How to find the publication and associated measures (if applicable)

A Gene reference (information is based on GeneID) will be generated for each triple see detail below

In detail we expect the following information (without ambiguity);

2) *Where can we find the data source(s)?*

File location: ftp://ftp.ncbi.nih.gov/gene//DATA/gene_info.gz

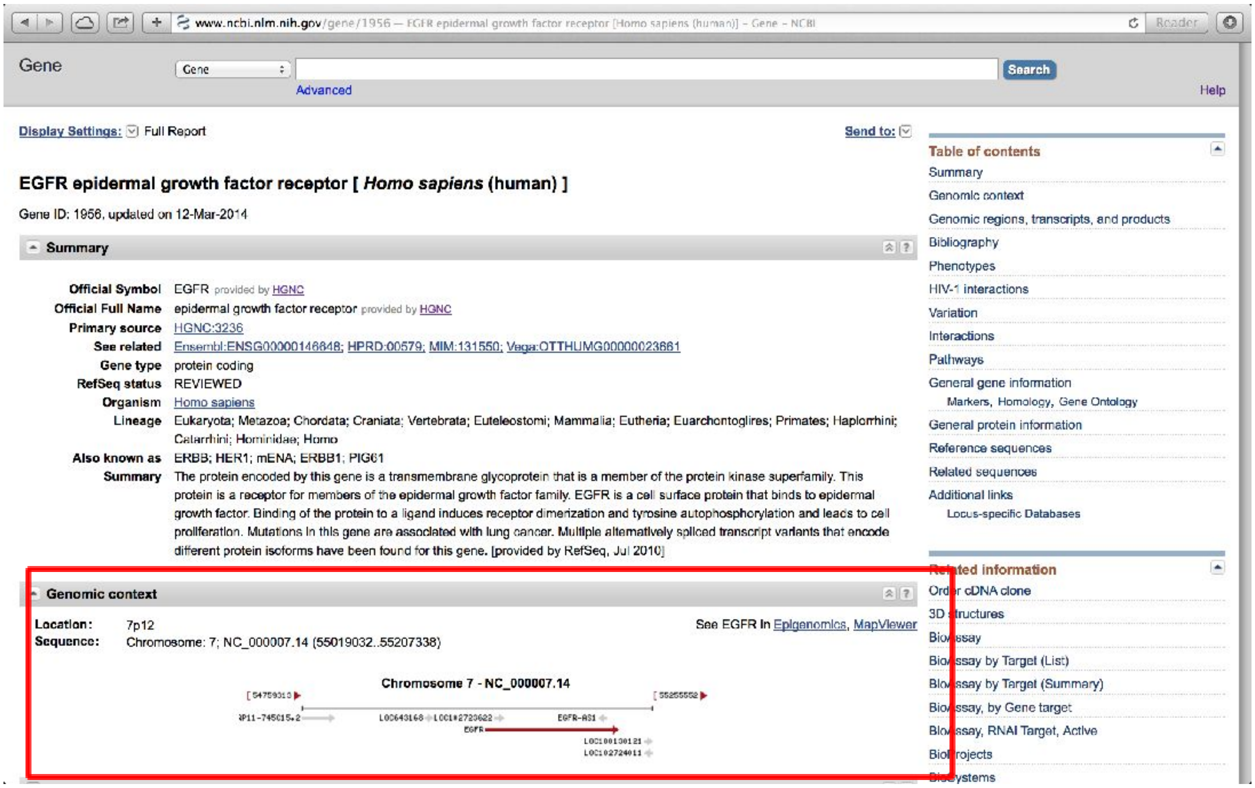
3) *Which tools / urls are available to validate / test content?*

http://www.ncbi.nlm.nih.gov/gene

4) *Which example screen dumps can we share to visualize a subject, object, predicate, publication? Can we validate against an example screen dump (perhaps based on tools/urls)?*

Example EGFR ; gene ID 1956

Information in this step included in General Gene information > "genomic context"



www.ncbi.nlm.nih.gov/gene/1956 — EGFR epidermal growth factor receptor [Homo sapiens (human)] - Gene - NCBI
Reader
Gene
Gene
Search
Advanced
Help
Display Settings:
Full Report
Send to:

EGFR epidermal growth factor receptor [*Homo sapiens* (human)]

Gene ID: 1956, updated on 12-Mar-2014

Summary

Official Symbol	EGFR provided by HGNC
Official Full Name	epidermal growth factor receptor provided by HGNC
Primary source	HGNC:3236
See related	Ensembl:ENSG00000146648; HPRD:00579; MIM:131550; Vega:OTTHUMG00000023661
Gene type	protein coding
RefSeq status	REVIEWED
Organism	Homo sapiens
Lineage	Eukaryota; Metazoa; Chordata; Craniata; Vertebrata; Euteleostomi; Mammalia; Eutheria; Euarchontoglires; Primates; Haplorrhini; Catarrhini; Hominidae; Homo
Also known as	ERBB; HER1; mENA; ERBB1; PIG61
Summary	The protein encoded by this gene is a transmembrane glycoprotein that is a member of the protein kinase superfamily. This protein is a receptor for members of the epidermal growth factor family. EGFR is a cell surface protein that binds to epidermal growth factor. Binding of the protein to a ligand induces receptor dimerization and tyrosine autophosphorylation and leads to cell proliferation. Mutations in this gene are associated with lung cancer. Multiple alternatively spliced transcript variants that encode different protein isoforms have been found for this gene. [provided by RefSeq, Jul 2010]

Genomic context

Location: 7p12

Sequence: Chromosome: 7; NC_000007.14 (55019032..55207338)

See EGFR in Epigenomics, MapViewer

Table of contents

Related information

Order cDNA clone
3D structures
BioAssay
BioAssay by Target (List)
BioAssay by Target (Summary)
BioAssay, by Gene target
BioAssay, RNAi Target, Active
BioProjects
BioSystems

5) *Are data source schema's available? And if so, where can they be found? Is it possible to mark the fields we need to extract /use?*

Location of the schema:

ftp://ftp.ncbi.nih.gov/gene//README

6) *Is the data source an authority?*

Yes, this is the authority for all genes.

a. *If so, do we expect SOLR create events? (new UUID)*

   No, only use existing UUID's

b. *If so, do we expect SOLR update events? (additional synonym to existing UUID)*

   No, only create concept measures for existing concepts, and create new triples and publications.

7) *What is the new element (subject, object, predicate, triple, publication) to be introduced?*

- *Concept measures*
- *Triples*
- *Publications associated to the triples.*

Concept Generation

When iterating Solr, identify genes by [geneid] (kb=geneid)

Do lookup in mongo collection gene_neighbors to identify correct document and fetch measures.

Triple generation

For each document in the gene_neigbor mongo collection, generate triples according to above rules.

Expecting around 250k new triples.

Access parameter

Source Mongo collection	**RD**	**RT**
entrezgene	0	57

Publication generation

For each triple generated, 1 publication is created with the following characteristics:

Publicationtype= curated database

Scientific value =5

Institution= NCBI

Publicationtitle= NCBI-gene/<geneid>

Publicationdate= file date for the source file

URL = www.ncbi.nlm.nih.gov/gene/?term=<geneid>

Parameter	type	Input	Example
Publication type	Fixed string	“curated database”	curated database
Scientific value	Fixed number	5	5
Institution	Fixed string	“NCBI”	NCBI
Publication title	Variable suffix	“NCBI-gene/”<geneID>	NCBI-gene/1956
Publication date	date	Date of the gene_neighbor file	2014-02-17
URL	Variable suffix	www.ncbi.nlm.nih.gov/gene/?term=<geneid>	www.ncbi.nlm.nih.gov/gene/?term=1956