now process

(1) our result:

0. import original content tag to table CONTENT\_TAG.

1. find all cui,AUI that is relevant to diabetes

/\*find all cui,AUI that is relevant to diabetes\*/

drop table if exists tmp\_diabetes;

create table tmp\_diabetes as select distinct CUI,AUI from MRCONSO where LAT='ENG' AND (str like '%diabetes%' or str like '%type 1 diabetes%' or str like '%type 2 diabetes%') ;

/\*4011 results\*/

2.  find all cui+aui which belong to  T047,  as part of the seeds

create table tmp\_diabetes\_T047 as

select distinct td.CUI,td.AUI from tmp\_diabetes as td

inner join MRSTY as sty

on (sty.TUI = 'T047' and sty.CUI = td.CUI)

;

/\*2662 results\*/

3. find all relationship that is relevant to diabetes and T047, as the seeds too.

/\*find all relationship that is relevant to diabetes and T047, as the seeds\*/

drop tables if exists tmp\_rel\_diabetes\_T047;

create table tmp\_rel\_diabetes\_T047 as

select distinct \*

from (

select distinct td.cui,td.aui,r.cui2 as rel\_cui, r.aui2 as rel\_aui

from mrrel r

inner join tmp\_diabetes\_T047 td

on td.aui = r.aui1 and td.cui = r. cui1

union

select distinct td.cui,td.aui,r.cui1 as rel\_cui, r.aui1 as rel\_aui

from mrrel r

inner join tmp\_diabetes\_T047 td

on td.aui = r.aui2 and td.cui = r. cui2

) as tmp

;

/\*8745 results\*/

4. find all tags that has relationship with diabetes and T047 seeds

/\*find all tags that has relationship with diabetes and T047 seeds\*/

drop table if exists content\_tag\_diabetes\_T047;

create table content\_tag\_diabetes\_T047 as

(select distinct c.blogId, c.target, c.wordIndex, c.cui, td.cui as rel\_cui, td.aui as rel\_aui, 0 as rel\_flag

from CONTENT\_TAG c

inner join tmp\_diabetes\_T047 td

on c.cui = td.cui COLLATE utf8\_unicode\_ci

) union (

select distinct c.blogId, c.target, c.wordIndex, c.cui, r.cui as rel\_cui, r.aui as rel\_aui, 1 as rel\_flag

from CONTENT\_TAG c

inner join tmp\_rel\_diabetes\_T047 r

on c.cui = r.rel\_cui COLLATE utf8\_unicode\_ci

);

/\*718169\*/

5.  reduce the data based on different level.

/\* reduce the data based on unique target/cui/rel\_cui\*/

drop table if exists content\_tag\_diabetes\_T047\_unique;

create table content\_tag\_diabetes\_T047\_unique as

select cd.\* from content\_tag\_diabetes\_T047 as cd

inner join

(select distinct id from content\_tag\_diabetes\_T047 group by blogId,target,wordIndex,rel\_cui order by stt) as temp

on cd.id = temp.id

;

/\*483903\*/

/\* reduce the data based on unique target/wordIndex/cui\*/

drop table if exists content\_tag\_diabetes\_T047\_unique2;

create table content\_tag\_diabetes\_T047\_unique2 as

select cd.\* from content\_tag\_diabetes\_T047\_unique as cd

inner join

(select distinct id from content\_tag\_diabetes\_T047\_unique group by blogId,target,wordIndex) as temp

on cd.id = temp.id

;

/\*11354 \*/

6.  create output data. Process it step by step rather than all in one sql for performance concern.

drop table if exists content\_tag\_diabetes\_T047\_unique2\_output;

create table content\_tag\_diabetes\_T047\_unique2\_output as

select distinct ct.blogId,

ct.target,

ct.cui,

org.umlsStr,

ct.wordIndex,

org.sentence,

ct.rel\_cui,

ct.rel\_aui,

ct.rel\_flag,

ct.STT

from content\_tag\_diabetes\_T047\_unique2 as ct

inner join (select \* from content\_tag as ct2

group by blogId, target,wordIndex,sentence) org

on org.blogId=ct.blogId and org.target = ct.target and org.wordIndex=ct.wordIndex

;

/\*11355\*/

alter table content\_tag\_diabetes\_T047\_unique2\_output add (`rel\_str` text);

update content\_tag\_diabetes\_T047\_unique2\_output as ct set rel\_str = (select STR from umls.MRCONSO where CUI=ct.rel\_cui and AUI=ct.rel\_aui);

select \* from content\_tag\_diabetes\_T047\_unique2\_output order by blogId,sentence

into outfile 'c:\\fsu\\content\_tag\_our\_T047\_unique.csv' fields terminated by ',' enclosed by '"' lines terminated by '\n';

(2) ytex result

0. create basic ytex result table using ytex output

create table content\_tag\_ytex as

select a.anno\_text, d.instance\_id, c.\* from v\_document\_cui\_sent c

inner join v\_annotation a on c.anno\_base\_id = a.anno\_base\_id

inner join v\_document d on d.document\_id = c.document\_id;

1. find all tags that has relationship with diabetes and T047, use the same seeds as our result.

/\*find all tags that has relationship with diabetes and T047\*/

drop table if exists content\_tag\_diabetes\_ytex\_T047;

create table content\_tag\_diabetes\_ytex\_T047 as

(select distinct c.instance\_id as blogId, c.anno\_text as target, c.anno\_base\_id as wordIndex, c.code as cui, td.cui as rel\_cui, td.aui as rel\_aui, 0 as rel\_flag

from ytex.content\_tag\_ytex c

inner join umls.tmp\_diabetes\_T047 td

on c.code = td.cui COLLATE utf8\_unicode\_ci

) union (

select distinct c.instance\_id as blogId, c.anno\_text as target, c.anno\_base\_id as wordIndex, c.code as cui, r.cui as rel\_cui, r.aui as rel\_aui, 1 as rel\_flag

from ytex.content\_tag\_ytex c

inner join umls.tmp\_rel\_diabetes\_T047 r

on c.code = r.rel\_cui COLLATE utf8\_unicode\_ci

);

/\*219203 results\*/

2. reduce the result at different level.

/\*add the stt column, because we want to pick the 'preferred name'.\*/

alter table content\_tag\_diabetes\_ytex\_T047 add (`STT` varchar(3) default null);

alter table content\_tag\_diabetes\_ytex\_T047 add (`id` int auto\_increment primary key );

update content\_tag\_diabetes\_ytex\_T047 as cd

inner join umls.MRCONSO as con

on cd.rel\_cui = con.cui and cd.rel\_aui = con.AUI

set cd.stt = con.stt;

/\* reduce the data based on unique target/cui/rel\_cui\*/

drop table if exists content\_tag\_diabetes\_ytex\_T047\_unique;

create table content\_tag\_diabetes\_ytex\_T047\_unique as

select cd.\* from content\_tag\_diabetes\_ytex\_T047 as cd

inner join

(select distinct id from content\_tag\_diabetes\_ytex\_T047 group by blogId,target,wordIndex,rel\_cui order by stt) as temp

on cd.id = temp.id

;

/\*106004\*/

/\* reduce the data based on unique target/cui\*/

drop table if exists content\_tag\_diabetes\_ytex\_T047\_unique2;

create table content\_tag\_diabetes\_ytex\_T047\_unique2 as

select cd.\* from content\_tag\_diabetes\_ytex\_T047\_unique as cd

inner join

(select distinct id from content\_tag\_diabetes\_ytex\_T047\_unique group by blogId,target,wordIndex) as temp

on cd.id = temp.id

;

/\*5800 \*/

3. create output

drop table if exists content\_tag\_ytex\_T047\_unique\_output;

create table content\_tag\_ytex\_T047\_unique\_output as

select distinct ct.blogId,

ct.target,

ct.cui,

org.cui\_text as umlsStr,

ct.wordIndex,

org.sentence\_text as sentence,

ct.rel\_cui,

ct.rel\_aui,

ct.rel\_flag,

ct.STT

from content\_tag\_diabetes\_ytex\_T047\_unique2 as ct

inner join (select \* from content\_tag\_ytex as cy

group by instance\_id, anno\_text,anno\_base\_id,sentence\_text) org

on org.instance\_id=ct.blogId and org.anno\_text = ct.target and org.anno\_base\_id=ct.wordIndex

;

/\*5800\*/

alter table content\_tag\_ytex\_T047\_unique\_output add (`rel\_str` text);

update content\_tag\_ytex\_T047\_unique\_output as ct set rel\_str = (select STR from umls.MRCONSO where CUI=ct.rel\_cui and AUI=ct.rel\_aui);

select \* from content\_tag\_ytex\_T047\_unique\_output order by blogId,sentence

into outfile 'c:\\fsu\\content\_tag\_ytex\_T047\_unique.csv' fields terminated by ',' enclosed by '"' lines terminated by '\n';

(3) analyzing the difference between our result and umls result

0. add the primary key 'id' for further process.

alter table ytex.content\_tag\_ytex\_T047\_unique\_output add (id int primary key auto\_increment);

alter table umls.content\_tag\_diabetes\_T047\_unique2\_output add (id int primary key auto\_increment);

1. find the overlap terms

/\* find the overlap content tags both in ytex's result and our result. (blogid, cui, target are same)\*/

drop table if exists content\_tag\_compare\_same;

create table content\_tag\_compare\_same as

select distinct yctu.\* from ytex.content\_tag\_ytex\_T047\_unique\_output as yctu

inner join (

select distinct yctu2.id from ytex.content\_tag\_ytex\_T047\_unique\_output as yctu2

inner join umls.content\_tag\_diabetes\_T047\_unique2\_output as uctu

on yctu2.blogId = uctu.blogId /\*and yctu2.cui = uctu.cui COLLATE utf8\_unicode\_ci\*/ and yctu2.target = uctu.target COLLATE utf8\_unicode\_ci

) as temp

on yctu.id = temp.id

order by yctu.blogId, yctu.sentence

;

/\*5529\*/

select \* from content\_tag\_compare\_same

into outfile 'c:\\fsu\\content\_tag\_compare\_same.csv' fields terminated by ',' enclosed by '"' lines terminated by '\n'; ;

drop table if exists content\_tag\_compare\_same2;

create table content\_tag\_compare\_same2 as

select distinct uctu.\* from umls.content\_tag\_diabetes\_T047\_unique2\_output as uctu

inner join (

select distinct uctu2.id from umls.content\_tag\_diabetes\_T047\_unique2\_output as uctu2

inner join ytex.content\_tag\_ytex\_T047\_unique\_output as yctu

on uctu2.blogId = yctu.blogId /\*and uctu2.cui = yctu.cui COLLATE utf8\_unicode\_ci\*/ and uctu2.target = yctu.target COLLATE utf8\_unicode\_ci

) as temp

on uctu.id = temp.id

order by uctu.blogId, uctu.sentence

;

/\*5581\*/

select target, count(\*) as cnt from content\_tag\_compare\_same group by target order by cnt desc;

select \* from content\_tag\_compare\_same2

into outfile 'c:\\fsu\\content\_tag\_compare\_same2.csv' fields terminated by ',' enclosed by '"' lines terminated by '\n'; ;

2. find the terms exists in ytex only

/\* find the content terms only in ytex's result \*/

drop table if exists content\_tag\_compare\_only\_ytex;

create table content\_tag\_compare\_only\_ytex as

select yctu.\* from ytex.content\_tag\_ytex\_T047\_unique\_output as yctu

where yctu.id not in (

select distinct same.id from ytex.content\_tag\_compare\_same as same

)

order by yctu.blogId, yctu.sentence

;

/\*271\*/

select target, count(\*) as cnt from content\_tag\_compare\_only\_ytex group by target order by cnt desc;

/\*45 result\*/

select \* from content\_tag\_compare\_only\_ytex

into outfile 'c:\\fsu\\content\_tag\_compare\_only\_ytex.csv' fields terminated by ',' enclosed by '"' lines terminated by '\n'; ;

3. find the terms exists in ours only

/\* find the content terms only in our result \*/

drop table if exists content\_tag\_compare\_only\_our;

create table content\_tag\_compare\_only\_our as

select uctu.\* from umls.content\_tag\_diabetes\_T047\_unique2\_output as uctu

where uctu.id not in (

select distinct same.id from ytex.content\_tag\_compare\_same2 as same

)

order by uctu.blogId, uctu.sentence

;

/\*5774\*/

select target, count(\*) as cnt from content\_tag\_compare\_only\_our group by target order by cnt desc;

/\*235 result\*/

select \* from content\_tag\_compare\_only\_our

into outfile 'c:\\fsu\\content\_tag\_compare\_only\_our.csv' fields terminated by ',' enclosed by '"' lines terminated by '\n'; ;