The idea is (I kind of improvised Jeff's idea):

1. for each author in disambig patref table, we add a coauthor column.
2. for each author in disambig reference table, we add a coauthor column too.
3. and then, for each author match entry from both tables (again we need to decide the criteria for that), we compare two co-author columns, to see how many coauthor match do they have. The more coauthor matches, the more confident we are for the author match

```
[38]    #Create coauthor list1
        fifty.names <-
        c("Georgina","Gustavo","Simonne","Birgit","Santos","Jaymie","Jed","St
        ephaine","Felisha","Linn","Eddy","Sherwood","Katrice"

        ,"Luella","Loreta","Verlie","Dirk","Candi","Pattie","Tomeka","Karry",
        "Phebe","My","Roseline","Mae","Adelia","Audie"

        ,"Yolanda","Charissa","Lupita","Neely","Mireya","Erich","Marcus","Mel
        vina","Diana","Saturnina","Efrain","Mickie","Abram"

        ,"Verda","Carmelina","Barb","Deon","Faustina","Nilsa","Geralyn","Heid
        y","John","Mercedes")

        set.seed(5)
        num.list.items <- 10
        list.auths1 <- replicate(10,"Georgina")
        list.auths1
        list.co.auths1 <- lapply(1:num.list.items, function(x)
        sample(fifty.names, size = sample(2:8, size = 1)))
        list.co.auths1
```

'Georgina'   'Georgina'   'Georgina'   'Georgina'   'Georgina'   'Georgina'
'Georgina'   'Georgina'   'Georgina'   'Georgina'

1.   'Melvina'   'Faustina'   'Luella'
2.   'Diana'   'Adelia'
3.   'Heidy'   'Jaymie'   'Luella'   'Roseline'   'Loreta'   'Adelia'
      'Sherwood'
4.   'Tomeka'   'Deon'   'Audie'
5.   'Faustina'   'Diana'   'Eddy'   'Heidy'   'Jed'   'Phebe'   'Tomeka'
6.   'Stephaine'   'Geralyn'   'Phebe'   'Simonne'   'Katrice'   'Gustavo'
      'Georgina'   'Karry'
7.   'Lupita'   'Tomeka'   'John'   'Mickie'   'Eddy'   'Efrain'
8.   'Geralyn'   'Audie'   'Saturnina'   'Birgit'   'Heidy'
9.   'Tomeka'   'Yolanda'   'Faustina'   'Nilsa'   'Barb'   'Candi'
10.   'Katrice'   'Linn'   'Jed'

```
[39]    #Create a pseudo coautho list 2
        list.co.auths2<-list()
        list.co.auths2[[1]]<-"Melvina"
        list.co.auths2[[2]]<-c("Melvina","Luella")
        list.co.auths2[[3]]<-c("Melvina","Luella","Faustina")
        list.co.auths2[[4]]<-c("Melvina","Lebron")
        list.co.auths2[[5]]<-c("Lebron")
```

```
[40]    #Let's match the coauthor list
        #1. 'Melvina' 'Faustina' 'Luella' vs "Melvina"
        tmp<-sapply(X = list.co.auths1[[1]], FUN = function (X) { grep(X,
        list.co.auths2[1], ignore.case = TRUE) }, simplify = "array")
        Reduce("+",unlist(tmp))
        #output: 1

        #2. 'Melvina' 'Faustina' 'Luella' vs "Melvina","Luella"
        tmp<-sapply(X = list.co.auths1[[1]], FUN = function (X) { grep(X,
        list.co.auths2[2], ignore.case = TRUE) }, simplify = "array")
        Reduce("+",unlist(tmp))
        #output: 2

        #3. 'Melvina' 'Faustina' 'Luella' vs "Melvina","Luella","Faustina"
        tmp<-sapply(X = list.co.auths1[[1]], FUN = function (X) { grep(X,
        list.co.auths2[3], ignore.case = TRUE) }, simplify = "array")
        Reduce("+",unlist(tmp))
        #output: 3

        #4. 'Melvina' 'Faustina' 'Luella' vs "Melvina","Lebron"
        tmp<-sapply(X = list.co.auths1[[1]], FUN = function (X) { grep(X,
        list.co.auths2[4], ignore.case = TRUE) }, simplify = "array")
        Reduce("+",unlist(tmp))
        #output: 1

        #5. 'Melvina' 'Faustina' 'Luella' vs "Lebron"
        tmp<-sapply(X = list.co.auths1[[1]], FUN = function (X) { grep(X,
        list.co.auths2[5], ignore.case = TRUE) }, simplify = "array")
        Reduce("+",unlist(tmp))
        #output: null
```

```
        1
        2
        3
        1
        NULL
```

SO YEAH IT'S WORKING, ALL WE NEED TO DISCUSS ARE

1. COAUTHOR LISTS FOR BOTH TABLE (i'mma write the code...)
2. Determiner what an author match is (FName+LName? FName+LName+MI?)

3. Determiner what a co-author match is (FName+LName? FName+LName+MI?)

[41]
```
#appendix: jeff's code

# given a long vector of names
# and a list of vectors of names
# find the location in the long vec of each of the names in each of
the short vecs

fifty.names <-
c("Georgina","Gustavo","Simonne","Birgit","Santos","Jaymie","Jed","St
ephaine","Felisha","Linn","Eddy","Sherwood","Katrice"

,"Luella","Loreta","Verlie","Dirk","Candi","Pattie","Tomeka","Karry",
"Phebe","My","Roseline","Mae","Adelia","Audie"

,"Yolanda","Charissa","Lupita","Neely","Mireya","Erich","Marcus","Mel
vina","Diana","Saturnina","Efrain","Mickie","Abram"

,"Verda","Carmelina","Barb","Deon","Faustina","Nilsa","Geralyn","Heid
y","John","Mercedes")

# first, lets make a random list of vectors of names
num.list.items <- 10
list.auths <- sample(fifty.names, size = num.list.items)
list.auths
list.co.auths <- lapply(1:num.list.items, function(x)
sample(fifty.names, size = sample(2:8, size = 1)))
list.co.auths


for (i in 1:num.list.items) {
  # now find those co-auths in a bigger list and get the index of
their positions
  indexes <- sapply(X = list.co.auths[[i]], FUN = function (X) {
grep(X, fifty.names, ignore.case = TRUE) }, simplify = "array")
  print(paste0("Auth (", i, "): ", list.auths[i], ":: co-Auths: ",
paste(list.co.auths[[i]], collapse = ","), ":: Matched Names: ",
paste(fifty.names[indexes], collapse = ",")))
  # now we might do something with the matched values
}
```

'Mireya'   'Felisha'   'Carmelina'   'Geralyn'   'Nilsa'   'Verda'   'Abram'
'Jed'   'Deon'   'Marcus'

1. 'Diana'   'Stephaine'   'Verda'   'Gustavo'   'Barb'   'Geralyn'
   'Neely'
2. 'Verlie'   'Geralyn'   'Abram'   'Linn'   'Santos'
3. 'Simonne'   'Heidy'   'Efrain'   'Verlie'   'Charissa'
4. 'Dirk'   'Linn'   'Tomeka'   'Verlie'   'Efrain'   'Eddy'
5. 'Roseline'   'My'   'Faustina'   'Marcus'   'Audie'   'Carmelina'

6.   'Tomeka'   'Luella'   'Mae'   'Phebe'   'Loreta'   'Stephaine'   'My'
7.   'Loreta'   'My'   'Pattie'   'John'   'Santos'   'Audie'   'Felisha'
8.   'Mercedes'   'Abram'
9.   'Loreta'   'Simonne'   'Marcus'   'Diana'   'Eddy'   'Verlie'   'Dirk'
    'Stephaine'
10.   'Barb'   'Pattie'

```
[1] "Auth (1): Mireya:: co-Auths:
Diana,Stephaine,Verda,Gustavo,Barb,Geralyn,Neely:: Matched Names:
Diana,Stephaine,Verda,Gustavo,Barb,Geralyn,Neely"
[1] "Auth (2): Felisha:: co-Auths: Verlie,Geralyn,Abram,Linn,Santos::
Matched Names: Verlie,Geralyn,Abram,Linn,Santos"
[1] "Auth (3): Carmelina:: co-Auths:
Simonne,Heidy,Efrain,Verlie,Charissa:: Matched Names:
Simonne,Heidy,Efrain,Verlie,Charissa"
[1] "Auth (4): Geralyn:: co-Auths:
Dirk,Linn,Tomeka,Verlie,Efrain,Eddy:: Matched Names:
Dirk,Linn,Tomeka,Verlie,Efrain,Eddy"
[1] "Auth (5): Nilsa:: co-Auths:
Roseline,My,Faustina,Marcus,Audie,Carmelina:: Matched Names:
Roseline,My,Faustina,Marcus,Audie,Carmelina"
```

[ ]