Text Analytics on Blockchain Data (Twitter Data)

1. **Code for extraction of BlockchainPressRelease.txt**

Code:

```
install.packages('openNLPmodels.en', repos='http://datacube.wu.ac.at/', type='source')
library(openNLPmodels.en)
library(NLP)
library(openNLP)

s=scan('BlockchainPressRelease.txt', what='character')
tokenizedS = annotate(s, list(Maxent_Sent_Token_Annotator(),
Maxent_Word_Token_Annotator()))

org_annotator=Maxent_Entity_Annotator(kind='organization')
person_annotator=Maxent_Entity_Annotator(kind='person')
date_annotator=Maxent_Entity_Annotator(kind='date')
percent_annotator=Maxent_Entity_Annotator(kind='percentage')
loc_annotator=Maxent_Entity_Annotator(kind='location')
money_annotator=Maxent_Entity_Annotator(kind='money')

swTokens = annotate(s, list(org_annotator, person_annotator, date_annotator, loc_annotator,
money_annotator), tokenizedS)
View(data.frame (swTokens))

org_annotator= Maxent_Entity_Annotator(kind='organization', probs=T)
swOrg=subset(swTokens,swTokens$features=='list(kind = "organization")')
swPerson=subset(swTokens,swTokens$features=='list(kind = "person")')
swDate=subset(swTokens,swTokens$features=='list(kind = "date")')
swLocation=subset(swTokens,swTokens$features=='list(kind = "location")')
swMoney=subset(swTokens,swTokens$features=='list(kind = "money")')

swOrg=subset(swTokens,substr(swTokens$features,1,26)=='list(kind = "organization"')

SWEntities=data.frame(Entity=character(), Position=numeric(), Type=character())

substring(paste(s, collapse=' '), 1117, 1124)

for (i in 1:nrow(as.data.frame(swOrg))) {
  SWEntities=rbind(SWEntities, cbind(substr(paste(s, collapse=' '),
swOrg$start[i],swOrg$end[i]),swOrg$start[i], 'Organization'))
}
for (i in 1:nrow(as.data.frame(swPerson))) {
  SWEntities=rbind(SWEntities, cbind(substr(paste(s, collapse=' '),
swPerson$start[i],swPerson$end[i]),swPerson$start[i], 'Person'))
}
for (i in 1:nrow(as.data.frame(swLocation))) {
```

```
  SWEntities=rbind(SWEntities, cbind(substr(paste(s, collapse=' '),
swLocation$start[i],swLocation$end[i]),swLocation$start[i], 'Location'))
}
for (i in 1:nrow(as.data.frame(swDate))) {
  SWEntities=rbind(SWEntities, cbind(substr(paste(s, collapse=' '),
swDate$start[i],swDate$end[i]),swDate$start[i], 'Date'))
}
for (i in 1:nrow(as.data.frame(swPercent))) {
  SWEntities=rbind(SWEntities, cbind(substr(paste(s, collapse=' '),
swPercent$start[i],swPercent$end[i]),swPercent$start[i], 'Phone'))
}
for (i in 1:nrow(as.data.frame(swMoney))) {
  SWEntities=rbind(SWEntities, cbind(substr(paste(s, collapse=' '),
swMoney$start[i],swMoney$end[i]),swMoney$start[i], 'Money'))
}
colnames(SWEntities)=c('Entity', 'Position', 'Type')
view(SWEntities)
```

Output of BlockchainPressRelease.txt

	Entity	Position	Type
1	Accelerate Enterprise Blockchain Adoption MOUNTAIN...	52	Organization
2	Currency Group	283	Organization
3	IBM	517	Organization
4	CME Group	549	Organization
5	New York Life	581	Organization
6	DCG Connect	2255	Organization
7	VP of Business Development	2380	Organization
8	DCG	2433	Organization
9	DCG Connect	3007	Organization
10	Stanford	3329	Organization
11	Harvard	3339	Organization
12	MIT	3348	Organization
13	University of Pennsylvania	3366	Organization
14	Digital Currency Group	3550	Organization
15	Srinivasan Sriram	1817	Person
16	Rebecca Liao	2366	Person
17	Rebecca Liao	3967	Person
18	Skuchain	320	Location
19	Stanford	3329	Location
20	May 22	117	Date
21	today	162	Date
22	2014	3518	Date
23	y for downstr	785	Money
24	facilitate ad	1279	Money
25	and to work w	1437	Money

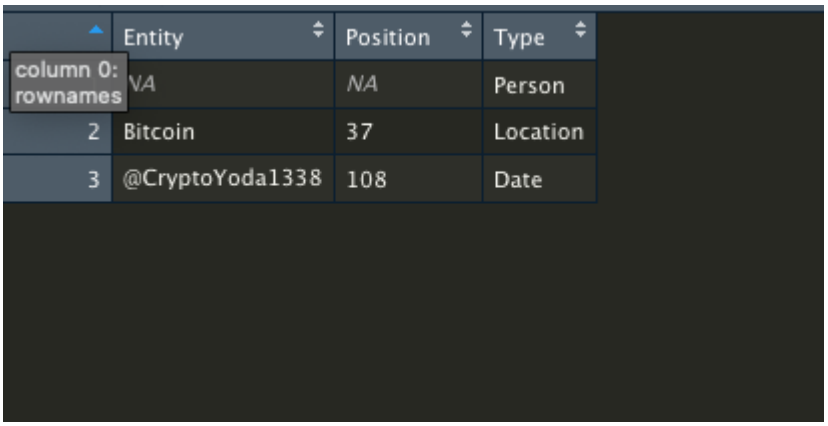
Code for extraction of BlockchainTweets.txt

Code:

```
library(openNLPmodels.en)
library(NLP)
library(openNLP)


s=scan('BlockchainTweets.txt', what='character')
tokenizedS = annotate(s, list(Maxent_Sent_Token_Annotator(),
Maxent_Word_Token_Annotator()))

org_annotator=Maxent_Entity_Annotator(kind='organization')
person_annotator=Maxent_Entity_Annotator(kind='person')
```

```
date_annotator=Maxent_Entity_Annotator(kind='date')
percent_annotator=Maxent_Entity_Annotator(kind='percentage')
loc_annotator=Maxent_Entity_Annotator(kind='location')
money_annotator=Maxent_Entity_Annotator(kind='money')

swTokens = annotate(s, list(org_annotator, person_annotator, date_annotator, loc_annotator,
money_annotator), tokenizedS)
View(data.frame (swTokens))

org_annotator= Maxent_Entity_Annotator(kind='organization', probs=T)
swOrg=subset(swTokens,swTokens$features=='list(kind = "organization")')
swPerson=subset(swTokens,swTokens$features=='list(kind = "person")')
swDate=subset(swTokens,swTokens$features=='list(kind = "date")')
swLocation=subset(swTokens,swTokens$features=='list(kind = "location")')
#swPhone=subset(swTokens,swTokens$features=='list(kind = "phone")')

swOrg=subset(swTokens,substr(swTokens$features,1,26)=='list(kind = "organization"')

SWEntities=data.frame(Entity=character(), Position=numeric(), Type=character())

substring(paste(s, collapse=' '), 1117, 1124)

for (i in 1:nrow(as.data.frame(swOrg))) {
  SWEntities=rbind(SWEntities, cbind(substr(paste(s, collapse=' '),
swOrg$start[i],swOrg$end[i]),swOrg$start[i], 'Organization'))
}
for (i in 1:nrow(as.data.frame(swPerson))) {
  SWEntities=rbind(SWEntities, cbind(substr(paste(s, collapse=' '),
swPerson$start[i],swPerson$end[i]),swPerson$start[i], 'Person'))
}
for (i in 1:nrow(as.data.frame(swLocation))) {
  SWEntities=rbind(SWEntities, cbind(substr(paste(s, collapse=' '),
swLocation$start[i],swLocation$end[i]),swLocation$start[i], 'Location'))
}
for (i in 1:nrow(as.data.frame(swDate))) {
  SWEntities=rbind(SWEntities, cbind(substr(paste(s, collapse=' '),
swDate$start[i],swDate$end[i]),swDate$start[i], 'Date'))
#}
#for (i in 1:nrow(as.data.frame(swPhone))) {
  #SWEntities=rbind(SWEntities, cbind(substr(paste(s, collapse=' '),
swPercent$start[i],swPercent$end[i]),swPercent$start[i], 'Phone'))
}
for (i in 1:nrow(as.data.frame(swMoney))) {
  SWEntities=rbind(SWEntities, cbind(substr(paste(s, collapse=' '),
swMoney$start[i],swMoney$end[i]),swMoney$start[i], 'Money'))
}
colnames(SWEntities)=c('Entity', 'Position', 'Type')
view(SWEntities)
```

Output of BlockchainTweets.txt



	Entity	Position	Type
column 0: rownames	NA	NA	Person
2	Bitcoin	37	Location
3	@CryptoYoda1338	108	Date

2. **Comparisons and Conclusion**
   The output of the BlockchainPressRelease.txt shows us the entity name, the position that it is in the text, and the entity type. We can observe that this method for this specific text is fairly accurate. However, we can also see that there are some errors, for example, all of the money extractions are inaccurate and it observed May 22 and 2017 as separate dates. It also inaccurately observed Skuchain as a location and was not able to observe Mountain View, Calif. as one.

   The output for BlockchainTweets.txt is very inaccurate. It was only able to observe three entities and they are all inaccurately categorized. @PayperExnet, @Bitcoin, @CryptoYoda1338, and @bloÖ are all entities that should have been recognized but were either inaccurately categorized or not picked up at all. This is likely because R's entity tagging performance does not included @'s as an entity type and none of them were categorized as a "Person". This tells us that there is still some ambiguity that exists in relation to processing natural language and this may make it difficult to get extremely accurate results.

3. **Accuracy and Improvement**
   While there was some accuracy with the code, we were able to observe some errors, this included inaccurately categorizing entities or not observing entities when they should be. As entity tagging is based on statistical machine learning, it is difficult for this process to be 100% accurate all the time. To improve its performance, it is recommended that the text is first preprocessed so it can be more accurately tagged, which as we learned in class is the best practice for NLP. For example, the removal of commas will help with observing dates more accurately.

   Another way to improve the accuracy of NLP algorithms is to train the algorithm to be more accurate with training data. Language is inherently ambiguous in nature and riddled with complexities in the form of expressions, grammar, accents, misspellings, symbols, emojis, etc which combined make it much more difficult to achieve greater accuracy. With better training data that's dynamic, the hope is that NLP and the algorithms used will be able to understand and detect the nuances that come with language.