

Report

Initial case:

Tp = 0

Fp = 0

Fn = 117

Step 1:

Regex written: `'([A-Za-z.]+)@([A-Za-z.]+\.)\.edu'`

This is regular expression match email pattern of type (someone)@(somewhere).edu

19 out 117 are matched properly with this regex.

Pattern matched	Files referred
balaji@stanford.edu	balaji
darlene@csl.stanford.edu kosecka@cs.gmu.edu	kunle
patrick.young@stanford.edu	psyoung
.....

True Positives (19):

```
{('balaji', 'e', 'balaji@stanford.edu'),  
 ('cheriton', 'e', 'cheriton@cs.stanford.edu'),  
 ('engler', 'e', 'engler@lcs.mit.edu'),  
 ('eroberts', 'e', 'eroberts@cs.stanford.edu'),  
 ('fedkiw', 'e', 'fedkiw@cs.stanford.edu'),  
 ('hanrahan', 'e', 'hanrahan@cs.stanford.edu'),  
 ('kosecka', 'e', 'kosecka@cs.gmu.edu'),  
 ('kunle', 'e', 'darlene@csl.stanford.edu'),  
 ('kunle', 'e', 'kunle@ogun.stanford.edu'),  
 ('nass', 'e', 'nass@stanford.edu'),  
 ('nick', 'e', 'nick.parlante@cs.stanford.edu'),  
 ('psyoung', 'e', 'patrick.young@stanford.edu'),  
 ('rinard', 'e', 'rinard@lcs.mit.edu'),  
 ('shoham', 'e', 'shoham@stanford.edu'),  
 ('thm', 'e', 'pkrokel@stanford.edu'),  
 ('widom', 'e', 'siroker@cs.stanford.edu'),  
 ('widom', 'e', 'widom@cs.stanford.edu'),  
 ('zelenski', 'e', 'zelenski@cs.stanford.edu'),  
 ('zm', 'e', 'manna@cs.stanford.edu')}
```

Summary after appending,

Tp = 19

Fp = 0

Fn = 98

Step 2:

Regex written: `'([A-Za-z.]+) @ ([A-Za-z.]+\.)\.edu'`

This is regular expression match email pattern of type (someone) @ (somewhere).edu

4 out of 117 are matched correctly with this regex.

Pattern matched	File referred
ashishg @ stanford.edu	Ashishg
.....

```
True Positives (4):
({'ashishg', 'e', 'ashishg@stanford.edu'},
 {'ashishg', 'e', 'rozm@stanford.edu'},
 {'ullman', 'e', 'ullman@cs.stanford.edu'},
 {'zelenski', 'e', 'zelenski@cs.stanford.edu'})
```

Summary after appending,

Tp = 22

Fp = 0

Fn = 95

Step 3

Regex written: '([A-Za-z.]+) @ ([A-Za-z.]+\.)\.edu'

This is regular expression match email pattern of type (someone) @ (somewhere).edu

1 out of 117 are matched correctly with this regex.

Pattern matched	File referred
dabo @ cs.stanford.edu	dabo
.....

```
True Positives (1):
({'dabo', 'e', 'dabo@cs.stanford.edu'})
```

Summary after appending,

Tp = 23

Fp = 0

Fn = 94

Step 4:

Regex written: '([A-Za-z.]+)@([A-Za-z.]+\.)\.EDU'

This is regular expression match email pattern of type (someone)@(somewhere).EDU

1 out of 117 are matched correctly with this regex.

Pattern matched	File referred
uma@cs.stanford.EDU	cheriton
.....

```
True Positives (1):
({'cheriton', 'e', 'uma@cs.stanford.edu'})
```

Summary after appending,

Tp = 24

Fp = 0

Fn = 93

Step 5

Regex written: '([A-Za-z.]+) AT ([A-Za-z.]+) DOT edu'

This is regular expression match email pattern of type (someone) AT (somewhere) DOT edu

1 out of 117 are matched correctly with this regex.

Pattern matched	File referred
subh AT stanford DOT edu	subh
.....

```
True Positives (1):  
{('subh', 'e', 'subh@stanford.edu')}
```

Summary after appending,

Tp = 25

Fp = 0

Fn = 92

Step 6

Regex written: '([A-Za-z.]+\s<at symbol>\s([A-Za-z.]+\s)\.edu'

This is regular expression match email pattern of type (someone) <at symbol> (somewhere).edu

2 out of 117 are matched correctly with this regex.

Pattern matched	File referred
manning <at symbol> cs.stanford.edu	manning
.....

```
True Positives (2):  
{('manning', 'e', 'dbarros@cs.stanford.edu'),  
 ('manning', 'e', 'manning@cs.stanford.edu')}
```

Summary after appending,

Tp = 27

Fp = 0

Fn = 90

Step 7

Regex written: '([A-Za-z.]+)@([A-Za-z.]+\s)\.edu'

This is regular expression match email pattern of type (someone)@ (somewhere).edu

2 out of 117 are matched correctly with this regex.

Pattern matched	File referred
ada@graphics.stanford.edu	Levoy
.....

```
True Positives (2):
{('levoy', 'e', 'ada@graphics.stanford.edu'),
 ('levoy', 'e', 'melissa@graphics.stanford.edu')}
```

Summary after appending,

Tp = 29

Fp = 0

Fn = 88

Step 8

Regex written: **'([A-Za-z.]+)@([A-Za-z.]+)\.edu'**

This is regular expression match email pattern of type (someone)@(somewhere).edu

3 out of 117 are matched correctly with this regex.

Pattern matched	File referred
latombe@cs.stanford.edu	Latombe
.....

```
True Positives (3):
{('latombe', 'e', 'asandra@cs.stanford.edu'),
 ('latombe', 'e', 'latombe@cs.stanford.edu'),
 ('latombe', 'e', 'liliana@cs.stanford.edu')}
```

Summary after appending,

Tp = 32

Fp = 0

Fn = 85

Step 9

Regex written: **'([A-Za-z.]+) WHERE ([A-Za-z.]+) DOM edu'**

This is regular expression match email pattern of type (someone) WHERE (somewhere) DOM edu

1 out of 117 are matched correctly with this regex.

Pattern matched	File referred
engler WHERE stanford DOM edu	engler
.....

```
True Positives (1):
{('engler', 'e', 'engler@stanford.edu')}
```

Summary after appending,

Tp = 33

Fp = 0

Fn = 84

From this step onwards try to match phone patterns to reduce the fn

Step 10

Regex written: `'(\d{3})-(\d{3})-(\d{4})'`

This is regular expression match phone pattern of type ###-###-####

19 out of 117 are matched correctly with this regex.

Pattern matched	File referred
650-723-1131	cheriton
.....

True Positives (19):

```
{('cheriton', 'p', '650-723-1131'),  
 ('cheriton', 'p', '650-725-3726'),  
 ('eroberts', 'p', '650-723-3642'),  
 ('eroberts', 'p', '650-723-6092'),  
 ('hager', 'p', '410-516-8000'),  
 ('rajeev', 'p', '650-723-4377'),  
 ('rajeev', 'p', '650-723-6045'),  
 ('rajeev', 'p', '650-725-4671'),  
 ('subh', 'p', '650-724-1915'),  
 ('subh', 'p', '650-725-3726'),  
 ('subh', 'p', '650-725-6949'),  
 ('ullman', 'p', '650-494-8016'),  
 ('ullman', 'p', '650-725-2588'),  
 ('ullman', 'p', '650-725-4802'),  
 ('widom', 'p', '650-723-0872'),  
 ('widom', 'p', '650-723-7690'),  
 ('widom', 'p', '650-725-2588'),  
 ('zelenski', 'p', '650-723-6092'),  
 ('zelenski', 'p', '650-725-8596')}
```

Summary after appending,

Tp = 52

Fp = 0

Fn = 65

Step 11

Regex written: `'(?:\()([0-9]{3})(?:\))([0-9]{3})-([0-9]{4})'`

This is regular expression match phone pattern of type (###)###-####

8 out of 117 are matched correctly with this regex.

Pattern matched	File referred
(650)725-2340	Tim
.....

```
True Positives (8):
{('ashishg', 'p', '650-723-1614'),
 ('ashishg', 'p', '650-723-4173'),
 ('ashishg', 'p', '650-814-1478'),
 ('horowitz', 'p', '650-725-3707'),
 ('horowitz', 'p', '650-725-6949'),
 ('tim', 'p', '650-724-9147'),
 ('tim', 'p', '650-725-2340'),
 ('tim', 'p', '650-725-4671')}
```

Summary after appending,

Tp = 60

Fp = 0

Fn = 57

Step 12

Regex written: **'(?:[()](?:[0-9]{3})?(?:[()])?)*[0-9]{3}-[0-9]{4}'**

This is regular expression match phone pattern of types (###)###-#### and (###) ###-####
47 out of 117 are matched correctly with this regex.

Pattern matched	File referred
(650) 724-6354	bgirod
.....

True Positives (47):

```
{('ashishg', 'p', '650-723-1614'),
 ('ashishg', 'p', '650-723-4173'),
 ('ashishg', 'p', '650-814-1478'),
 ('bgirod', 'p', '650-723-4539'),
 ('bgirod', 'p', '650-724-3648'),
 ('bgirod', 'p', '650-724-6354'),
 ('dabo', 'p', '650-725-3897'),
 ('dabo', 'p', '650-725-4671'),
 ('hager', 'p', '410-516-5521'),
 ('hager', 'p', '410-516-5553'),
 ('hanrahan', 'p', '650-723-0033'),
 ('hanrahan', 'p', '650-723-8530'),
 ('horowitz', 'p', '650-725-3707'),
 ('horowitz', 'p', '650-725-6949'),
 ('kosecka', 'p', '703-993-1710'),
 ('kosecka', 'p', '703-993-1876'),
 ('kunle', 'p', '650-723-1430'),
 ('kunle', 'p', '650-725-3713'),
 ('kunle', 'p', '650-725-6949'),
 ('lam', 'p', '650-725-3714'),
 ('lam', 'p', '650-725-6949'),
 ('latombe', 'p', '650-721-6625'),
 ('latombe', 'p', '650-723-0350'),
 ('latombe', 'p', '650-723-4137'),
 ('latombe', 'p', '650-725-1449'),
 ('levoy', 'p', '650-723-0033'),
 ('levoy', 'p', '650-724-6865'),
 ('levoy', 'p', '650-725-3724'),
 ('levoy', 'p', '650-725-4089'),
 ('manning', 'p', '650-723-7683'),
 ('manning', 'p', '650-725-1449'),
 ('manning', 'p', '650-725-3358'),
 ('nick', 'p', '650-725-4727'),
 ('ok', 'p', '650-723-9753'),
 ('ok', 'p', '650-725-1449'),
 ('rinard', 'p', '617-253-1221'),
 ('rinard', 'p', '617-258-6922'),
 ('serafim', 'p', '650-723-3334'),
 ('serafim', 'p', '650-725-1449'),
 ('thm', 'p', '650-725-3383'),
 ('thm', 'p', '650-725-3636'),
 ('thm', 'p', '650-725-3938'),
 ('tim', 'p', '650-724-9147'),
 ('tim', 'p', '650-725-2340'),
 ('tim', 'p', '650-725-4671'),
 ('zm', 'p', '650-723-4364'),
 ('zm', 'p', '650-725-4671')}
```

Summary after appending,

Tp = 99

Fp = 0

Fn = 18

Step 13

Regex written: `'(?:[()]{0-9}{3})(?:[()]\s{0-9}{3})-([0-9]{4})'`

This is regular expression match phone pattern of type [###] ###-####

2 out of 117 are matched correctly with this regex.

Pattern matched	File referred
[650] 723-5499	Nass
.....

True Positives (2):

```
{('nass', 'p', '650-723-5499'), ('nass', 'p', '650-725-2472')}
```

Summary after appending,

Tp = 101

Fp = 0

Fn = 16

Step 14

Regex written: `'(\d{3}) (\d{3}) (\d{4})'`

This is regular expression match phone pattern of type ### ### ####

2 out of 117 are matched correctly with this regex.

Pattern matched	File referred
650 725 9046	Pal
.....

True Positives (2):

```
{('pal', 'p', '650-725-9046'), ('jurafsky', 'p', '650-723-5666')}
```

Summary after appending,

Tp = 103

Fp = 0

Fn = 14

Step 15

Regex written: `'(\d{3}) ([0-9]{3})-([0-9]{4})'`

This is regular expression match phone pattern of type ### ###-####

2 out of 117 are matched correctly with this regex.

Pattern matched	File referred
650 725-1449	shoham
.....

True Positives (2):

```
{('shoham', 'p', '650-723-3432'), ('shoham', 'p', '650-725-1449')}
```


Summary after appending

Tp = 105

Fp = 0

Fn = 12

Thus from step 1 to 9 appended regex for email to reduce fn and from step 10 onwards appended regex for phone numbers to reduce fn

The final summary is

Tp = 105

Fp = 0

Fn = 12

The complete output is

True Positives (105):

```
{('ashishg', 'e', 'ashishg@stanford.edu'),  
 ('ashishg', 'e', 'rozm@stanford.edu'),  
 ('ashishg', 'p', '650-723-1614'),  
 ('ashishg', 'p', '650-723-4173'),  
 ('ashishg', 'p', '650-814-1478'),  
 ('balaji', 'e', 'balaji@stanford.edu'),  
 ('bgirod', 'p', '650-723-4539'),  
 ('bgirod', 'p', '650-724-3648'),  
 ('bgirod', 'p', '650-724-6354'),  
 ('cheriton', 'e', 'cheriton@cs.stanford.edu'),  
 ('cheriton', 'e', 'uma@cs.stanford.edu'),  
 ('cheriton', 'p', '650-723-1131'),  
 ('cheriton', 'p', '650-725-3726'),  
 ('dabo', 'e', 'dabo@cs.stanford.edu'),  
 ('dabo', 'p', '650-725-3897'),  
 ('dabo', 'p', '650-725-4671'),  
 ('engler', 'e', 'engler@lcs.mit.edu'),  
 ('engler', 'e', 'engler@stanford.edu'),  
 ('eroberts', 'e', 'eroberts@cs.stanford.edu'),  
 ('eroberts', 'p', '650-723-3642'),  
 ('eroberts', 'p', '650-723-6092'),  
 ('fedkiw', 'e', 'fedkiw@cs.stanford.edu'),  
 ('hager', 'p', '410-516-5521'),  
 ('hager', 'p', '410-516-5553'),  
 ('hager', 'p', '410-516-8000'),  
 ('hanrahan', 'e', 'hanrahan@cs.stanford.edu'),  
 ('hanrahan', 'p', '650-723-0033'),  
 ('hanrahan', 'p', '650-723-8530'),  
 ('horowitz', 'p', '650-725-3707'),  
 ('horowitz', 'p', '650-725-6949'),  
 ('jurafsky', 'p', '650-723-5666'),
```

('kosecka', 'e', 'kosecka@cs.gmu.edu'),
('kosecka', 'p', '703-993-1710'),
('kosecka', 'p', '703-993-1876'),
('kunle', 'e', 'darlene@csl.stanford.edu'),
('kunle', 'e', 'kunle@ogun.stanford.edu'),
('kunle', 'p', '650-723-1430'),
('kunle', 'p', '650-725-3713'),
('kunle', 'p', '650-725-6949'),
('lam', 'p', '650-725-3714'),
('lam', 'p', '650-725-6949'),
('latombe', 'e', 'asandra@cs.stanford.edu'),
('latombe', 'e', 'latombe@cs.stanford.edu'),
('latombe', 'e', 'liliana@cs.stanford.edu'),
('latombe', 'p', '650-721-6625'),
('latombe', 'p', '650-723-0350'),
('latombe', 'p', '650-723-4137'),
('latombe', 'p', '650-725-1449'),
('levoy', 'e', 'ada@graphics.stanford.edu'),
('levoy', 'e', 'melissa@graphics.stanford.edu'),
('levoy', 'p', '650-723-0033'),
('levoy', 'p', '650-724-6865'),
('levoy', 'p', '650-725-3724'),
('levoy', 'p', '650-725-4089'),
('manning', 'e', 'dbarros@cs.stanford.edu'),
('manning', 'e', 'manning@cs.stanford.edu'),
('manning', 'p', '650-723-7683'),
('manning', 'p', '650-725-1449'),
('manning', 'p', '650-725-3358'),
('nass', 'e', 'nass@stanford.edu'),
('nass', 'p', '650-723-5499'),
('nass', 'p', '650-725-2472'),
('nick', 'e', 'nick.parlante@cs.stanford.edu'),
('nick', 'p', '650-725-4727'),
('ok', 'p', '650-723-9753'),
('ok', 'p', '650-725-1449'),
('pal', 'p', '650-725-9046'),
('psyoung', 'e', 'patrick.young@stanford.edu'),
('rajeev', 'p', '650-723-4377'),
('rajeev', 'p', '650-723-6045'),
('rajeev', 'p', '650-725-4671'),
('rinard', 'e', 'rinard@lcs.mit.edu'),
('rinard', 'p', '617-253-1221'),
('rinard', 'p', '617-258-6922'),
('serafim', 'p', '650-723-3334'),
('serafim', 'p', '650-725-1449'),
('shoham', 'e', 'shoham@stanford.edu'),
('shoham', 'p', '650-723-3432'),

```
( 'shoham', 'p', '650-725-1449'),
( 'subh', 'e', 'subh@stanford.edu'),
( 'subh', 'p', '650-724-1915'),
( 'subh', 'p', '650-725-3726'),
( 'subh', 'p', '650-725-6949'),
( 'thm', 'e', 'pkrokel@stanford.edu'),
( 'thm', 'p', '650-725-3383'),
( 'thm', 'p', '650-725-3636'),
( 'thm', 'p', '650-725-3938'),
( 'tim', 'p', '650-724-9147'),
( 'tim', 'p', '650-725-2340'),
( 'tim', 'p', '650-725-4671'),
( 'ullman', 'e', 'ullman@cs.stanford.edu'),
( 'ullman', 'p', '650-494-8016'),
( 'ullman', 'p', '650-725-2588'),
( 'ullman', 'p', '650-725-4802'),
( 'widom', 'e', 'siroker@cs.stanford.edu'),
( 'widom', 'e', 'widom@cs.stanford.edu'),
( 'widom', 'p', '650-723-0872'),
( 'widom', 'p', '650-723-7690'),
( 'widom', 'p', '650-725-2588'),
( 'zelenski', 'e', 'zelenski@cs.stanford.edu'),
( 'zelenski', 'p', '650-723-6092'),
( 'zelenski', 'p', '650-725-8596'),
( 'zm', 'e', 'manna@cs.stanford.edu'),
( 'zm', 'p', '650-723-4364'),
( 'zm', 'p', '650-725-4671')}
```

False Positives (0):

set()

False Negatives (12):

```
{( 'dlwh', 'e', 'dlwh@stanford.edu'),
( 'hager', 'e', 'hager@cs.jhu.edu'),
( 'jks', 'e', 'jks@robotics.stanford.edu'),
( 'jrafsky', 'e', 'jrafsky@stanford.edu'),
( 'lam', 'e', 'lam@cs.stanford.edu'),
( 'ouster', 'e', 'ouster@cs.stanford.edu'),
( 'ouster', 'e', 'teresa.lynn@stanford.edu'),
( 'pal', 'e', 'pal@cs.stanford.edu'),
( 'serafim', 'e', 'serafim@cs.stanford.edu'),
( 'subh', 'e', 'uma@cs.stanford.edu'),
( 'ullman', 'e', 'support@gradiance.com'),
( 'vladlen', 'e', 'vladlen@stanford.edu')}
```

Summary: tp=105, fp=0, fn=12

To list of all the examples that did not match with the current regular expression with two extracted parts, ending in .edu.

Email	file
-------	------

d-l-w-h-@-s-t-a-n-f-o-r-d-.-e-d-u	dlwh
hager at cs dot jhu dot edu	hager
jks at robotics;stanford;edu	jks
obfuscate('stanford.edu','jurafsky');	jurafsky
lam at cs.stanford.edu	lam
teresa.lynn (followed by "@stanford.edu") ouster (followed by “@cs.stanford.edu”)	ouster
pal at cs stanford edu	pal
serafim at cs dot stanford dot edu	serafim
uma at cs dot stanford dot edu	subh
support at gradiance dt com	ullman
vladlen at <!-- die!--> stanford <!-- spam pigs!--> dot <!-- die!--> edu	vladlen

Lets consider different types,

Type 1:

d-l-w-h-@-s-t-a-n-f-o-r-d-.-e-d-u

This email is not matched due to the presence of ‘-’ between each letter. We need to use string replace function to match this email correctly with the email in the Gold file.

Type 2:

hager at cs dot jhu dot edu

serafim at cs dot stanford dot edu

uma at cs dot stanford dot edu

pal at cs stanford edu

For the above mentioned emails we need more than 2 parenthesis(in specific 3) to identify or match

Type 3:

jks at robotics;stanford;edu :

This email is not matched due to the presence of ‘;’ between robotics and stanford. We need more than 2 parenthesis(in specific 3) to match this one or we have to use string replace function to convert it and match with the Gold file.

Type 4:

obfuscate('stanford.edu','jurafsky');

This is difficult pattern to match as the email id is written in reverse manner and without any usage of ‘@’ or ‘at’ or ‘AT’ symbols.

Type 5:

pal at cs stanford edu

This is not matched due to the presence of space “ ” between cs and stanford instead of a ‘.’. We need more than 2 parenthesis(in specific 3) to match this one or we have to use string replace function to convert it and match with the Gold file.

Type 6:

support at gradiance dt com

This is not matched because its ending with 'com' address (not .edu type) and also 'dt' inplace of '.' We need to compose a different for loop using String replace function with .com type to have this matched perfectly.

for cpat **in** compatterns:

each cpat has 2 sets of parentheses so each match will have 2 items in a list

matches = re.findall(cpat,line)

for m **in** matches:

string formatting operator % takes elements of list m

and inserts them in place of each %s in the result string

email = '%s@%s.com' % m

res.append((name,'e',email))

Type 7:

teresa.lynn (followed by "@stanford.edu")

ouster (followed by "“@cs.stanford.edu”")

This is the difficult pattern to match id is in different format and presence of (,), &, ; symbols. We need more than 2 parenthesis to match this one or we have to use string replace function to convert it and match with the Gold file.

Type 8

Step 16

Regex written: '([A-Za-z.]+) [at]+ ([A-Za-z.]+)\.edu'

This is regular expression match email pattern of types (someone) at (somewhere).edu

Pattern matched	File referred
lam at cs.stanford.edu	Lam
.....

Summary after appending,

Tp = 106

Fp = 2

Fn = 11

Pattern at fp	Files
Server at cs.stanford.edu Port 80	Jure
Server at infolab.stanford.edu Port 80	Plotkin

```

False Positives (2):
({'jure', 'e', 'server@cs.stanford.edu'},
 {'plotkin', 'e', 'server@infolab.stanford.edu'})
False Negatives (11):
({'dlwh', 'e', 'dlwh@stanford.edu'},
 {'hager', 'e', 'hager@cs.jhu.edu'},
 {'jks', 'e', 'jks@robotics.stanford.edu'},
 {'jurafsky', 'e', 'jurafsky@stanford.edu'},
 {'ouster', 'e', 'ouster@cs.stanford.edu'},
 {'ouster', 'e', 'teresa.lynn@stanford.edu'},
 {'pal', 'e', 'pal@cs.stanford.edu'},
 {'serafim', 'e', 'serafim@cs.stanford.edu'},
 {'subh', 'e', 'uma@cs.stanford.edu'},
 {'ullman', 'e', 'support@gradiance.com'},
 {'vladlen', 'e', 'vladlen@stanford.edu'})
Summary: tp=106, fp=2, fn=11

```

this email pattern increases tp by 1, decreases the fn by 1 but increases the fp by 2.

It matches the pattern in lam file ie. lam at cs.stanford.edu correctly but fails to match Server at cs.stanford.edu Port 80 and Server at infolab.stanford.edu Port 80 of jure and plotkin file respectively.

To match such pattern correctly we need more than 2 domains (to be specific domain after .edu extension)

Type 9:

vladlen at <!-- die!--> stanford <!-- spam pigs!--> dot <!-- die!--> edu

This is difficult pattern to match due to the presence of <****> . We need more than 2 parenthesis to match this one or we have to use string replace function to convert it and match with the Gold file.

For the difficult obscured emails

Method 1

USING THE 'ONCLICK' EVENT

You can create a regular mailto hyperlink for your email address but replace some of the characters – like the dot | DOT | dt by '.' And at | AT by '@' . Then add an onclick event to this hyperlink that will substitute the text with the actual symbols.

```

<a href = "mailto:johnATstanfordDOTedu"
onclick = "this.href=this.href
.replace(/AT/, '@')
.replace(/DOT/, '.')"

```

By this method we can capture most of the email patterns in spamlord through regular expressions

Method 2:

Random Array

Split your email address into multiple parts and create an array in JavaScript out of these parts. Next join these parts in the correct order and use the .innerHTML property to add the email address to the web page.

```
<span id= "email"></span>
```

```
<script>
```

```
    var parts = ["john","abc","com","&#46","&#64"];
    var email = parts[0]+ parts[4]+ parts[1]+parts[3]+parts[2];
    document.getElementById("email").innerHTML=email;
```

```
</script>
```

By this method we can capture most of the email patterns in spamlord.

(there are other open source tools like WORDPRESS which uses the built-in antispambot() function to encode email address. The function will encode the characters in address to their HTML character entity (the letter a becomes a and the @ symbol becomes @) though they will render correctly in the browser.

```
<?php echo antispambot("john@abc.com"); ?>
```

Referred the link <https://www.sitepoint.com/community/t/hiding-mailto-email-address-link-from-spam-bot-scrapers/276368>)