In [1]:

```
# To import the required libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import tweepy
import csv
import string
import re
import nltk
#nltk.download('vader_lexicon')

from textblob import TextBlob
from nltk.sentiment.vader import SentimentIntensityAnalyzer
from wordcloud import WordCloud, STOPWORDS
from PIL import Image
```

In [2]:

```
#Twitter Authentication
consumerKey = "VqSDkT8l1yUclHfKEHjo6bYpk"
consumerSecret = "qM4lMUsuzlhXGAreyhPyEgin4rHOduAuT3MJGkxMulquIz5zmX"
accessToken = "1275263936590479360-jQRiw2U5dCZ5saB1AJzlE64uwHeL1x"
accessTokenSecret = "ncYLGFI3n7JKbWssLIVj7tehJVDW1oDAxR6BX7y9xTKll"

auth = tweepy.OAuthHandler(consumerKey, consumerSecret)
auth.set_access_token(accessToken, accessTokenSecret)
api = tweepy.API(auth)
```

In [3]:

```
#Functions to extract tweets and calculate percentage
def extract_tweets(hashtags, numTweets):
    data = tweepy.Cursor(api.search_tweets, q=hashtags, lang="en").items(numTweets)
    return data

def percent(actual,total):
    return 100 * float(actual)/float(total)
```

In [4]:

```
#Most common hashtags relevant to our topic ('Roe V Wade')
hashtags = "#RoeVWade OR #RoeVsWade OR #Roe_v_Wade OR #RoeOverturned OR #RoeVWadeOverturned OR #RoeVersusWade OR #ProLife
numTweets = 3200

#Function Call to extract tweets based on the hashtags
```

```
tweets = extract_tweets(hashtags, numTweets)

#Variables to store positive, negative and neutral tweets and the corresponding count of tweets
pos  = 0
pos_list = []
neg = 0
neg_list = []
neut = 0
neut_list = []
polarity = 0
tw_list = []
```

In [5]:

```
#To analyze the tweets using SentimentIntensityAnalyzer
#and assign the tweets to appropriate categories based on the polarity scores
#Also to write the tweets to a csv file
csvFile = open('RoeVWade_Overturned_Tweets', 'a')
csvWriter = csv.writer(csvFile)

for tweet in tweets:
    csvWriter.writerow([tweet.created_at, tweet.text.encode('utf-8')])
    tw_list.append(tweet.text)
    analysis = TextBlob(tweet.text)
    score = SentimentIntensityAnalyzer().polarity_scores(tweet.text)
    negative = score['neg']
    neutral = score['neu']
    positive = score['pos']
    comp = score['compound']
    polarity += analysis.sentiment.polarity

    if negative > positive:
        neg_list.append(tweet.text)
        neg += 1

    elif positive > negative:
        pos_list.append(tweet.text)
        pos += 1

    elif positive == negative:
        neut_list.append(tweet.text)
        neut += 1
csvFile.close()
```

In [6]:

```
#To find the number of tweets reflecting each sentiment
#and to calculate the percentage of positive, negative and neutral tweets
pos = percent(pos, numTweets)
neg = percent(neg, numTweets)
neut = percent(neut, numTweets)
polarity = percent(polarity, numTweets)
pos = format(pos, '.1f')
neg = format(neg, '.1f')
neut = format(neut, '.1f')

tw_list = pd.DataFrame(tw_list)
neut_list = pd.DataFrame(neut_list)
neg_list = pd.DataFrame(neg_list)
pos_list = pd.DataFrame(pos_list)
print("Total Number of tweets: ",len(tw_list))
print("Positive Tweets: ",len(pos_list))
print("Negative Tweets: ", len(neg_list))
print("Neutral Tweets: ",len(neut_list))
```

```
Total Number of tweets:  1242
Positive Tweets:  509
Negative Tweets:  385
Neutral Tweets:  348
```

In [7]:

```
#To remove duplicate tweets and extract the text of the unique ones
tw_list.drop_duplicates(inplace = True)
tweets_list = pd.DataFrame(tw_list)
tweets_list['text'] = tweets_list[0]

#Removing RT, Punctuation and hash
remove_rt = lambda x: re.sub('RT @\w+: ',' ',x)
rt = lambda x: re.sub('(#[A-Za-z0-9]+)|(@[0-9A-Za-z \t])(\w+:\/\/\S+)',' ',x)
tweets_list['text'] = tweets_list.text.map(remove_rt).map(rt)
tweets_list['text'] = tweets_list.text.str.lower()
tweets_list.head(10)
```

Out[7]:

	0	text
0	RT @SaMor97691348: IF YOU ARE #ProChoice\nIF Y...	if you are \nif you want to \nif you agree ...
1	This is powerful. Please share!\n#prolifemyass...	this is powerful. please share!\n \n \nhtt...
2	RT @Springfield_Lab: @carolewilliams @sartain_...	@carolewilliams @sartain_stephen well done @h...
3	Follow @votelandsman #OH1\n\nIF YOU ARE #ProC...	follow @votelandsman 1\n\nif you are \nif y...

	0	text
4	RT @Nishaobgyn: A much belated &amp; very welc...	a much belated &amp; very welcome change to ...
5	RT @FreeIsMe3: Lyrics I put down due to the na...	lyrics i put down due to the nature of the wa...
7	IF YOU ARE #ProChoice\nIF YOU WANT TO #CodifyR...	if you are \nif you want to \nif you agree ...
9	RT @David_Leavitt: Guns have more rights than ...	guns have more rights than women\n\n htt...
10	Check out Stacey Smith's video! #TikTok https:...	check out stacey smith's video! https://t.co...
11	RT @lisasmithreed: There is an undeniable war ...	there is an undeniable war on women. this is...

In [8]:

```
#Calculating Negative, Positive, Neutral and Compound values

tweets_list[['polarity', 'subjectivity']] = tweets_list['text'].apply(lambda Text: pd.Series(TextBlob(Text).sentiment))
for index, row in tweets_list['text'].iteritems():
    score = SentimentIntensityAnalyzer().polarity_scores(row)
    negative = score['neg']
    neutral = score['neu']
    positive = score['pos']
    comp = score['compound']
    if negative > positive:
        tweets_list.loc[index, 'sentiment'] = "negative"
    elif positive > negative:
        tweets_list.loc[index, 'sentiment'] = "positive"
    else:
        tweets_list.loc[index, 'sentiment'] = "neutral"
    tweets_list.loc[index, 'neg'] = negative
    tweets_list.loc[index, 'neu'] = neutral
    tweets_list.loc[index, 'pos'] = positive
    tweets_list.loc[index, 'compound'] = comp

#Print the first 10 records in the tweets_list table
tweets_list.head(10)
```

Out[8]:

	0	text	polarity	subjectivity	sentiment	neg	neu	pos	compound
0	RT @SaMor97691348: IF YOU ARE #ProChoice\nIF Y...	if you are \nif you want to \nif you agree ...	0.000000	0.000000	positive	0.000	0.743	0.257	0.4215
1	This is powerful. Please share!\n#prolifemyass...	this is powerful. please share!\n \n \nhtt...	0.375000	1.000000	positive	0.000	0.283	0.717	0.7644

	0	text	polarity	subjectivity	sentiment	neg	neu	pos	compound
2	RT @Springfield_Lab: @carolewilliams @sartain_...	@carolewilliams @sartain_stephen well done @h...	0.000000	0.750000	positive	0.000	0.618	0.382	0.7003
3	Follow @votelandsman #OH1\n\nIF YOU ARE #ProC...	follow @votelandsman 1\n\nif you are \nif y...	0.000000	0.000000	positive	0.000	0.787	0.213	0.4215
4	RT @Nishaobgyn: A much belated &amp; very welc...	a much belated &amp; very welcome change to ...	0.600000	0.600000	positive	0.000	0.838	0.162	0.5095
5	RT @FreeIsMe3: Lyrics I put down due to the na...	lyrics i put down due to the nature of the wa...	0.163889	0.402778	positive	0.000	0.617	0.383	0.9081
7	IF YOU ARE #ProChoice\nIF YOU WANT TO #CodifyR...	if you are \nif you want to \nif you agree ...	0.000000	0.000000	positive	0.000	0.759	0.241	0.4215
9	RT @David_Leavitt: Guns have more rights than ...	guns have more rights than women\n\n htt...	0.500000	0.500000	neutral	0.000	1.000	0.000	0.0000
10	Check out Stacey Smith's video! #TikTok https:...	check out stacey smith's video! https://t.co...	0.000000	0.000000	neutral	0.000	1.000	0.000	0.0000
11	RT @lisasmithreed: There is an undeniable war ...	there is an undeniable war on women. this is...	0.500000	0.500000	negative	0.365	0.635	0.000	-0.7845

In [9]:

```
#Creating new data frames for all sentiments (positive, negative and neutral)

negative_tweets = tweets_list[tweets_list["sentiment"]=="negative"]
positive_tweets = tweets_list[tweets_list["sentiment"]=="positive"]
neutral_tweets = tweets_list[tweets_list["sentiment"]=="neutral"]
```

In [10]:

```
#Function to output count and percentage of tweets corresponding to each sentiment

def count_sentiment(data,feature):
    total=data.loc[:,feature].value_counts(dropna=False)
    percentage=round(data.loc[:,feature].value_counts(dropna=False,normalize=True)*100,2)
    return pd.concat([total,percentage],axis=1,keys=['Total','Percentage'])
```

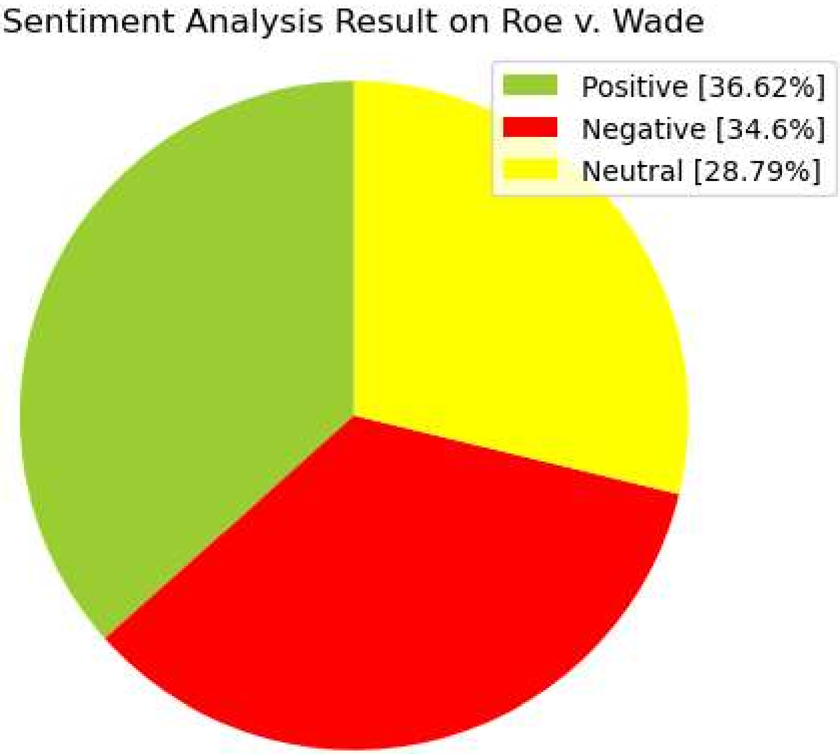
In [17]:

```
#Print the count of tweets for each sentiment
#and a pie chart representing the distribution of tweets with different sentiments

print(count_sentiment(tweets_list,"sentiment"))
```

```python
sentiment_split = count_sentiment(tweets_list,"sentiment")
pie_data = pd.DataFrame(sentiment_split)
sizes = [pie_data.iloc[0][1], pie_data.iloc[1][1], pie_data.iloc[2][1]]
colors = ['yellowgreen', 'red', 'yellow']
patches, texts = plt.pie(sizes,colors=colors, startangle=90)
labels = ['Positive ['+str(pie_data.iloc[0][1])+'%]' , 'Negative ['+str(pie_data.iloc[1][1])+'%]','Neutral ['+str(pie_dat
plt.style.use('default')
plt.legend(labels)
plt.title("Sentiment Analysis Result on Roe v. Wade" )
plt.axis('equal')
plt.show()
```

```
          Total  Percentage
negative    290       36.62
positive    274       34.60
neutral     228       28.79
```



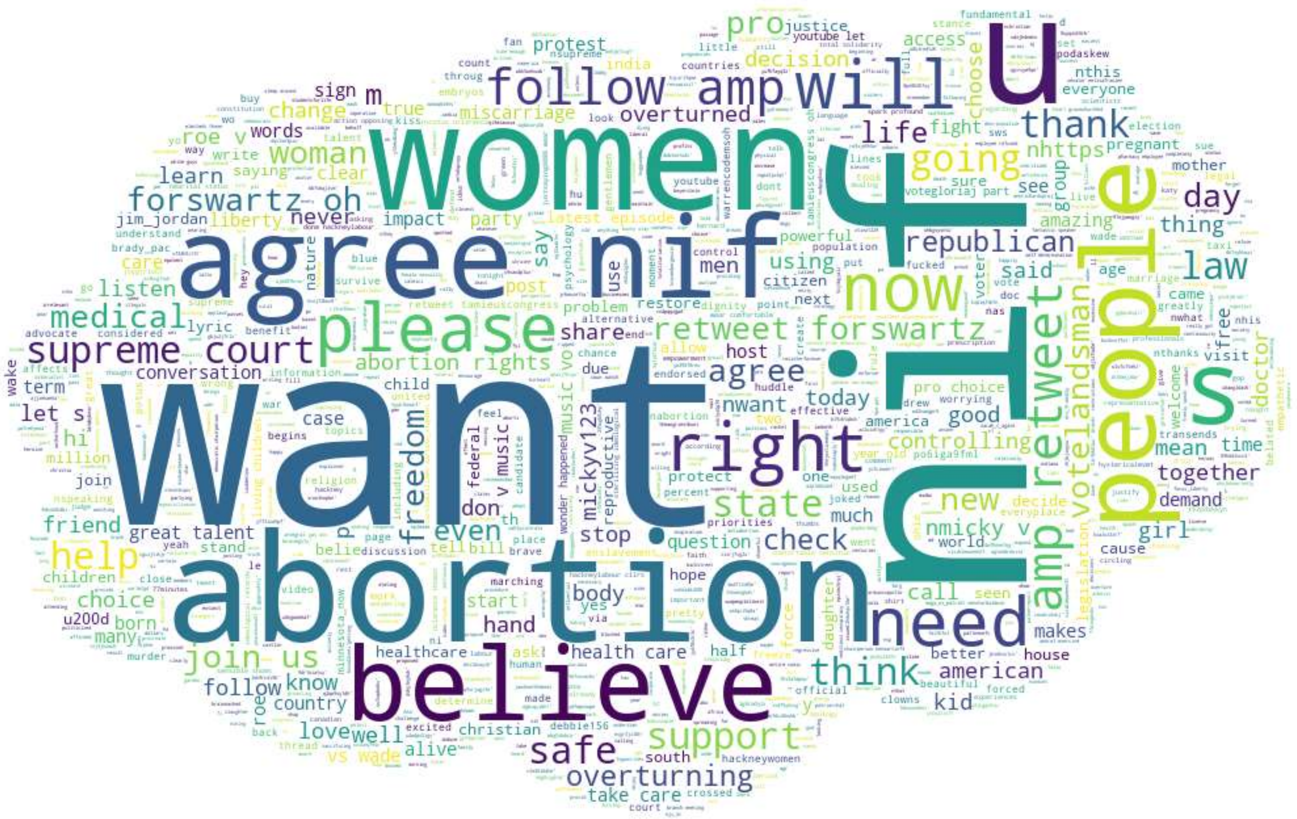
In [18]:

```python
#Function Definition for creating and printing Wordcloud
```

```
def print_wordcloud(text):
    mask = np.array(Image.open("wordcloud.png"))
    STOPWORDS.update(["https", "co", "n", "t"])
    stopwords = set(STOPWORDS)
    wc = WordCloud(background_color="white",
                   mask = mask,
                   max_words=2000,
                   stopwords=stopwords,
                   repeat=True)
    wc.generate(str(text))
    wc.to_file("wc.png")
    path="wc.png"
    display(Image.open(path))
```

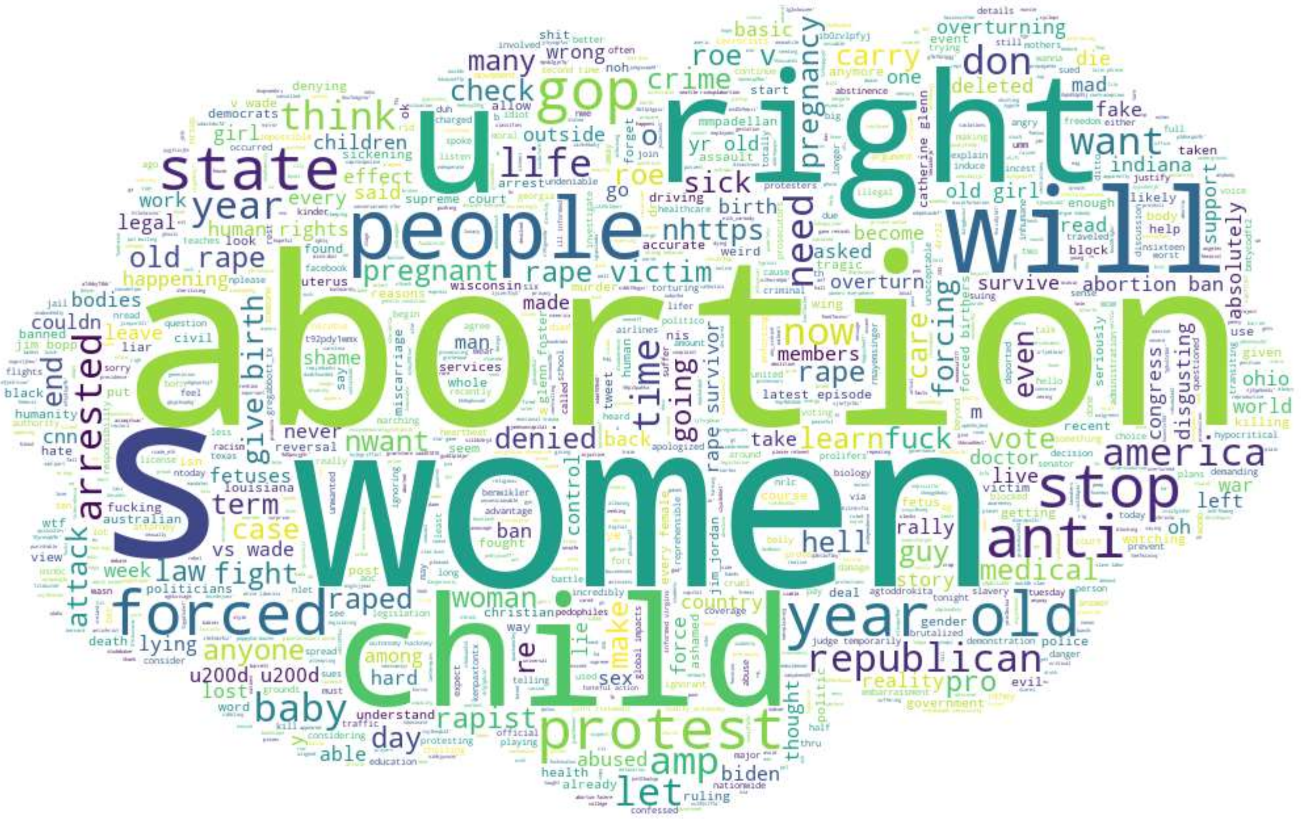
In [19]:

```
#Function call to print wordcloud for positive sentiment
print_wordcloud(positive_tweets["text"].values)
```

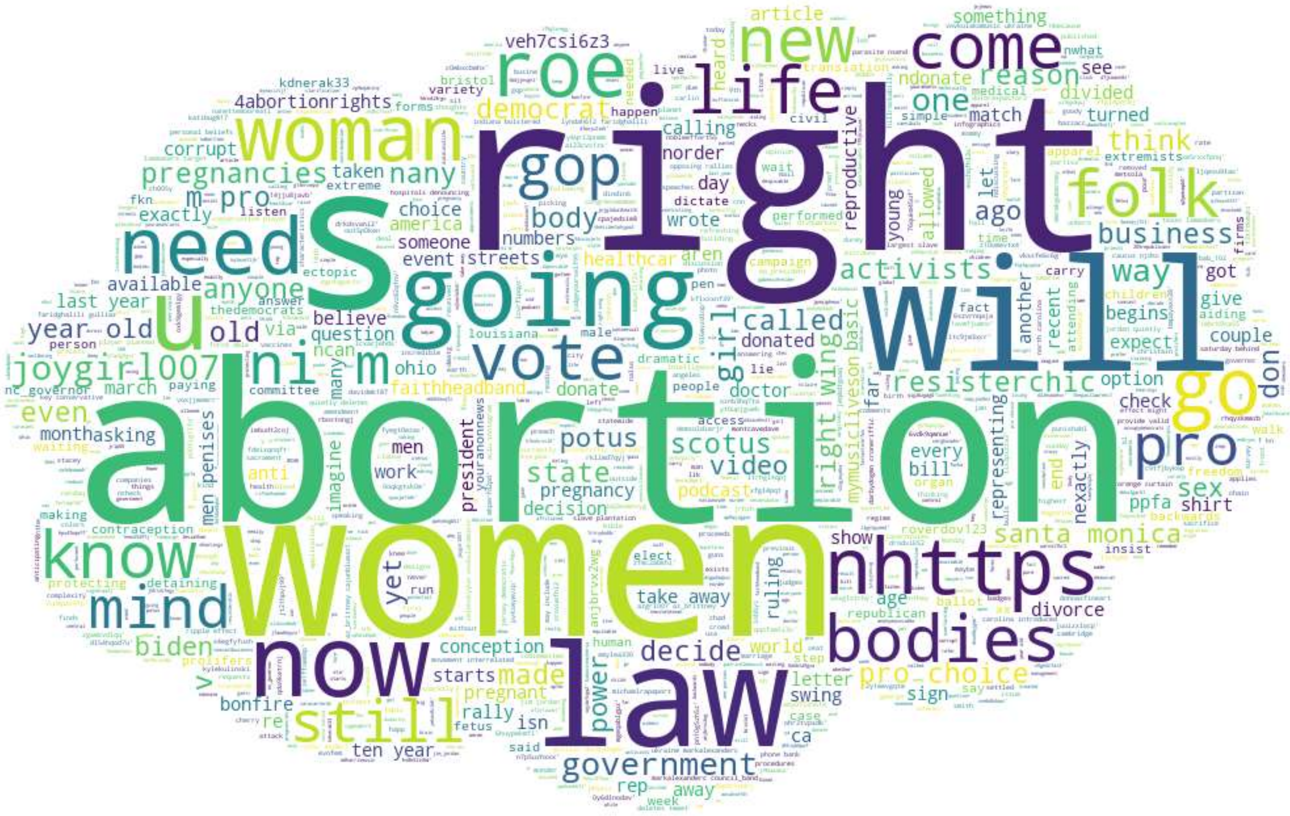


In [20]:

```
#Function call to print wordcloud for negative sentiment
print_wordcloud(negative_tweets["text"].values)
```



```
In [21]:  #Function call to print wordcloud for neutral sentiment
           print_wordcloud(neutral_tweets["text"].values)
```


right
abortion
women
s
will
law
going
woman
roe
new
come
life
gop
folk
need
u
vote
go
pro
now
nhttps
bodies
know
mind
still
ni m
joygirl1007
government
scotus
state
potus
video
body
activists
way
business
think
reason
one
democrat
pregnancies
nany
anyone
old
year
last year
called
girl
right wing
resisterchic
santa monica
sex
ppfa
pro choice
decide
made
biden
conception
starts
bonfire
rally
pregnant
isn
power
rep
away
swing
sign
letter
case
ca
world
take away
ruling
republican
divorce
age
show
roverdov123
podcast
decision
pregnancy
imagine
men
anti
work
president
believe
ohio
faithheadband
donate
doctor
donated
reproductive
mymusicliveson
basic
far
every
bill
representing
nexactly
end
check
option
expect
couple
give
got
begins
recent
attending
children
another
let
ago
young
allowed
wrote
norder
calling
heard
day
dictate
healthcare
event
streets
numbers
someone
choice
america
exactly
m pro
corrupt
taken
extreme
4abortionrights
kdnerak33
veh7csi6z3
bristol
variety
forms
happen
live
civil
ndonate
match
turned
divided
medical
nwhat
something
article
translation
available
question
via
ncan
many
penises
asking
month
even
march
paying
committee
men
contraception
detaining
protecting
prolifers
making
waiting
nc_governor
person
ectopic
louisiana
listen
thedemocrats
fkn
republican
ballot
organ
performed
campaign
fact
carry
see
insist
shirt
backwards
freedom
walk
aiding
apparel
extremists
firms
dramatic
people
access
lie
pen
elect
human
step
week
said
fetus
ten year
re
v
don
answer
run
yet
youranonnews
anjbrvx2wg
sex