ICE - 1: Introduction to NLP basic techniques

Deliverables:

##

Code provided below (Score = 40%)

(1) Run the tutorial (20%)

(2) Answer any question asked in between (20%)

###

Tutorial provided in PDF (Score = 60%)

(3) Run the other tutorial provided in word and transform that code in the same workbook below. (10%)

(4) Use the same concept to parse data from any live website that talk about SpaceX technologies (15%)

(5) Perform Frequency Distribution (10%)

(6) Visualize Tokens (10%)

(7) Answer the Questions at the end (15%)

###

Good Luck with the first NLP In class tasks ✌

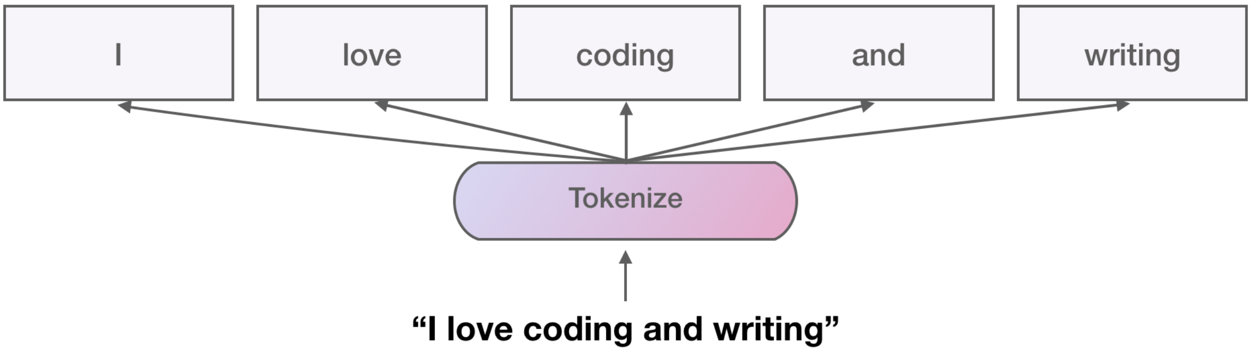
Fundamentals of NLP: Tokenization

Natural language processing (NLP) has made substantial advances in the past few years due to the success of [modern techniques (https://nlpoverview.com/)](https://nlpoverview.com/) that are based on [deep learning (https://en.wikipedia.org/wiki/Deep_learning)](https://en.wikipedia.org/wiki/Deep_learning). With the rise of the popularity of NLP and the availability of different forms of large-scale data, it is now even more imperative to understand the inner workings of NLP techniques and concepts, from first principles, as they find their way into real-world usage and applications that affect society at large. Building intuitions and having a solid grasp of concepts are both important for coming up with innovative techniques, improving research, and building safe, human-centered AI and NLP technologies.

In this first chapter, which is part of a series called **Fundamentals of NLP**, we will learn about some of the most important **basic concepts** that power NLP techniques used for research and building real-world applications. Some of these techniques include *lemmatization*, *stemming*, *tokenization*, and *sentence segmentation*. These are all important techniques to train efficient and effective NLP models. Along the way, we will also cover best practices and common mistakes to avoid when training and building NLP models. We also provide some exercises for you to keep practicing and exploring some ideas.

In every chapter, we will introduce the theoretical aspect and motivation of each concept covered. Then we will obtain hands-on experience by using bootstrap methods, industry-standard tools, and other open-source libraries to implement the different techniques. Along the way, we will also cover best practices, share important references, point out common mistakes to avoid when training and building NLP models, and discuss what lies ahead.

Tokenization



With any typical NLP task, one of the first steps is to tokenize your pieces of text into its individual words/tokens (process demonstrated in the figure above), the result of which is used to create so-called vocabularies that will be used in the langauge model you plan to build. This is actually one of the techniques that we will use the most throughout this series but here we stick to the basics.

Below I am showing you an example of a simple tokenizer without any following any standards. All it does is extract tokens based on a white space seperator.

Try to running the following code blocks.

Type *Markdown* and LaTeX: α^2

In [2]:

```
## required libraries that need to be installed
#%%capture
!pip install -U spacy
!pip install -U spacy-lookups-data
!python -m spacy download en_core_web_sm
```

```
Collecting spacy
  Downloading spacy-3.4.1-cp39-cp39-win_amd64.whl (11.8 MB)
Collecting wasabi<1.1.0,>=0.9.1
  Downloading wasabi-0.10.1-py3-none-any.whl (26 kB)
Collecting typer<0.5.0,>=0.3.0
  Downloading typer-0.4.2-py3-none-any.whl (27 kB)
Collecting spacy-loggers<2.0.0,>=1.0.0
  Downloading spacy_loggers-1.0.3-py3-none-any.whl (9.3 kB)
Requirement already satisfied: jinja2 in c:\users\rajesh nemani\anaconda3
\lib\site-packages (from spacy) (2.11.3)
Collecting langcodes<4.0.0,>=3.2.0
  Downloading langcodes-3.3.0-py3-none-any.whl (181 kB)
Collecting pathy>=0.3.5
  Downloading pathy-0.6.2-py3-none-any.whl (42 kB)
Collecting spacy-legacy<3.1.0,>=3.0.9
  Downloading spacy_legacy-3.0.10-py2.py3-none-any.whl (21 kB)
Requirement already satisfied: numpy>=1.15.0 in c:\users\rajesh nemani\ana
conda3\lib\site-packages (from spacy) (1.21.5)
Requirement already satisfied: packaging>=20.0 in c:\users\rajesh nemani\a
```

In [4]:

```
## tokenizing a piecen of text
doc = "I love coding and writing"
for i, w in enumerate(doc.split(" ")):
    print("Token " + str(i) + ": " + w)
```

```
Token 0: I
Token 1: love
Token 2: coding
Token 3: and
Token 4: writing
```

All the code does is separate the sentence into individual tokens. The above simple block of code works well on the text I have provided. But typically, text is a lot noisier and complex than the example I used. For instance, if I used the word "so-called" is that one word or two words? For such scenarios, you may need more advanced approaches for tokenization. You can consider stripping away the "-" and splitting into two tokens or just combining into one token but this all depends on the problem and domain you are working on.

Another problem with our simple algorithm is that it cannot deal with extra whitespaces in the text. In addition, how do we deal with cities like "New York" and "San Francisco"?

In [1]:

```
import spacy

nlp = spacy.load("en_core_web_sm")
doc = nlp("New York and San Francisco ")
for chunk in doc.noun_chunks:
    print(chunk.text)
```

```
New York
San Francisco
```

Exercise 1: Copy the code from above and add extra whitespaces to the string value assigned to the `doc` variable and identify the issue with the code. Then try to fix the issue. Hint: Use `text.strip()` to fix the problem.

In [12]:

```
###  ENTER CODE HERE
## tokenizing a piecen of text with extra spaces
doc = "I            love     coding               and           writing"
#removing extra spaces
for i, w in enumerate(doc.strip().split()):
    print("Token " + str(i) + ": " + w)
###
```

```
Token 0: I
Token 1: love
Token 2: coding
Token 3: and
Token 4: writing
```

Tokenization can also come in different forms. For instance, more recently a lot of state-of-the-art NLP models such as [BERT (https://arxiv.org/pdf/1810.04805.pdf)](https://arxiv.org/pdf/1810.04805.pdf) make use of `subword` tokens in which frequent combinations of characters also form part of the vocabulary. This helps to deal with the so-called out of vocabulary (OOV) problem. We will discuss this in upcoming chapters, but if you are interested in reading more about this now, check this [paper (https://static.googleusercontent.com/media/research.google.com/en//pubs/archive/37842.pdf)](https://static.googleusercontent.com/media/research.google.com/en//pubs/archive/37842.pdf).

To demonstrate how you can achieve more reliable tokenization, we are going to use [spaCy (https://spacy.io/)](https://spacy.io/), which is an impressive and robust Python library for natural language processing. In particular, we are going to use the built-in tokenizer found [here (https://spacy.io/usage/linguistic-features#sbd-custom)](https://spacy.io/usage/linguistic-features#sbd-custom).

Run the code block below.

In [3]:

```python
## import the libraries
import spacy
## load the language model
nlp = spacy.load("en_core_web_sm")

## tokenization
doc = nlp("This is the so-called lemmatization")
for token in doc:
    print(token.text)
```

```
This
is
the
so
-
called
lemmatization
```

All the code does is tokenize the text based on a pre-built language model.

Try putting different running text into the `nlp()` part of the code above. The tokenizer is quiet robust and it includes a series of built-in rules that deal with exceptions and special cases such as those tokens that contain puctuations like "`" and ".", "-", etc. You can even add your own rules, find out how here (https://spacy.io/usage/linguistic-features#special-cases).

In a later chapter of the series, we will do a deep dive on tokenization and the different tools that exist out there that can simplify and speed up the process of tokenization to build vocabularies. Some of the tools we will explore are the Keras Tokenizer API (https://www.tensorflow.org/api_docs/python/tf/keras/preprocessing/text/Tokenizer) and Hugging Face Tokenizer (https://github.com/huggingface/tokenizers).

You can add the 2nd tutorial from word file provided in canvas below. The code is also provided. Please keep in mind that do not put all code in one block split in a fashionable way.

In [177]:

```python
from bs4 import BeautifulSoup
import urllib.request
import nltk
from nltk.corpus import stopwords
import pandas as pd
import seaborn as sns

```

In [56]:

```
pip install html5lib
```

```
Requirement already satisfied: html5lib in c:\users\rajesh nemani\anaconda3
\lib\site-packages (1.1)
Requirement already satisfied: webencodings in c:\users\rajesh nemani\anacon
da3\lib\site-packages (from html5lib) (0.5.1)
Requirement already satisfied: six>=1.9 in c:\users\rajesh nemani\anaconda3
\lib\site-packages (from html5lib) (1.16.0)
Note: you may need to restart the kernel to use updated packages.
```

In [176]:

```
pip install seaborn
```

```
Requirement already satisfied: seaborn in c:\users\rajesh nemani\anaconda3\l
ib\site-packages (0.11.2)
Requirement already satisfied: scipy>=1.0 in c:\users\rajesh nemani\anaconda
3\lib\site-packages (from seaborn) (1.7.3)
Requirement already satisfied: numpy>=1.15 in c:\users\rajesh nemani\anacond
a3\lib\site-packages (from seaborn) (1.21.5)
Requirement already satisfied: pandas>=0.23 in c:\users\rajesh nemani\anacon
da3\lib\site-packages (from seaborn) (1.4.2)
Requirement already satisfied: matplotlib>=2.2 in c:\users\rajesh nemani\ana
conda3\lib\site-packages (from seaborn) (3.5.1)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\rajesh nemani\a
naconda3\lib\site-packages (from matplotlib>=2.2->seaborn) (1.3.2)
Requirement already satisfied: packaging>=20.0 in c:\users\rajesh nemani\ana
conda3\lib\site-packages (from matplotlib>=2.2->seaborn) (21.3)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\rajesh neman
i\anaconda3\lib\site-packages (from matplotlib>=2.2->seaborn) (2.8.2)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\rajesh nemani\a
naconda3\lib\site-packages (from matplotlib>=2.2->seaborn) (4.25.0)
Requirement already satisfied: pyparsing>=2.2.1 in c:\users\rajesh nemani\an
aconda3\lib\site-packages (from matplotlib>=2.2->seaborn) (3.0.4)
Note: you may need to restart the kernel to use updated packages.
Requirement already satisfied: cycler>=0.10 in c:\users\rajesh nemani\anacon
da3\lib\site-packages (from matplotlib>=2.2->seaborn) (0.11.0)
Requirement already satisfied: pillow>=6.2.0 in c:\users\rajesh nemani\anaco
nda3\lib\site-packages (from matplotlib>=2.2->seaborn) (9.0.1)
Requirement already satisfied: pytz>=2020.1 in c:\users\rajesh nemani\anacon
da3\lib\site-packages (from pandas>=0.23->seaborn) (2021.3)
Requirement already satisfied: six>=1.5 in c:\users\rajesh nemani\anaconda3
\lib\site-packages (from python-dateutil>=2.7->matplotlib>=2.2->seaborn) (1.
16.0)
```

In [58]:

```
url='https://www.cnbc.com/2021/08/09/spacex-acquiring-satellite-data-start-up-swarm-tec
response = urllib.request.urlopen(url)
```

In [59]:

```
html = response.read()
```

In [129]:

```
print(html)
```

```
b'<!DOCTYPE html><html lang="en" prefix="og=https://ogp.me/ns#" itemscope
="" itemType="https://schema.org/WebPage"><head><link rel="preload" as="fo
nt" href="https://static-redesign.cnbcfm.com/dist/icomoon.ttf" type="font/
ttf" crossorigin="anonymous"/><link rel="preload" as="font" href="https://
static-redesign.cnbcfm.com/dist/351C86_0_0.woff2" type="font/woff2" crosso
rigin="anonymous"/><link rel="preload" as="font" href="https://static-rede
sign.cnbcfm.com/dist/351C86_1_0.woff2" type="font/woff2" crossorigin="anon
ymous"/><link rel="preload" as="font" href="https://static-redesign.cnbcf
m.com/dist/351C86_2_0.woff2" type="font/woff2" crossorigin="anonymous"/><l
ink rel="preload" as="font" href="https://static-redesign.cnbcfm.com/dist/
351C86_3_0.woff2" type="font/woff2" crossorigin="anonymous"/><link rel="pr
eload" as="font" href="https://static-redesign.cnbcfm.com/dist/351C86_4_0.
woff2" type="font/woff2" crossorigin="anonymous"/><link rel="preload" as
="font" href="https://static-redesign.cnbcfm.com/dist/LyonText-Bold-Web.wo
ff2" type="font/woff2" crossorigin="anonymous"/><link rel="preload" as="fo
nt" href="https://static-redesign.cnbcfm.com/dist/LyonText-Regular-Web.wof
f2" type="font/woff2" crossorigin="anonymous"/><meta name="format-detectio
n" content="telephone=no"/><style type="text/css">@charset "UTF-8";.Recapt
chaAcknowledgement-acknowledgement{-webkit-box-flex:1;color:#747474;-ms-fl
```

In [130]:

```
soup = BeautifulSoup(html,"html.parser")
```

In [62]:

```
print(soup)
```

```
<!DOCTYPE html>
<html itemscope="" itemtype="https://schema.org/WebPage" lang="en" prefix
="og=https://ogp.me/ns#"><head><link as="font" crossorigin="anonymous" hre
f="https://static-redesign.cnbcfm.com/dist/icomoon.ttf" rel="preload" type
="font/ttf"/><link as="font" crossorigin="anonymous" href="https://static-
redesign.cnbcfm.com/dist/351C86_0_0.woff2" rel="preload" type="font/woff
2"/><link as="font" crossorigin="anonymous" href="https://static-redesign.
cnbcfm.com/dist/351C86_1_0.woff2" rel="preload" type="font/woff2"/><link a
s="font" crossorigin="anonymous" href="https://static-redesign.cnbcfm.com/
dist/351C86_2_0.woff2" rel="preload" type="font/woff2"/><link as="font" cr
ossorigin="anonymous" href="https://static-redesign.cnbcfm.com/dist/351C86
_3_0.woff2" rel="preload" type="font/woff2"/><link as="font" crossorigin
="anonymous" href="https://static-redesign.cnbcfm.com/dist/351C86_4_0.woff
2" rel="preload" type="font/woff2"/><link as="font" crossorigin="anonymou
s" href="https://static-redesign.cnbcfm.com/dist/LyonText-Bold-Web.woff2"
rel="preload" type="font/woff2"/><link as="font" crossorigin="anonymous" h
ref="https://static-redesign.cnbcfm.com/dist/LyonText-Regular-Web.woff2" r
el="preload" type="font/woff2"/><meta content="telephone=no" name="format-
detection"/><style type="text/css">@charset "UTF-8";.RecaptchaAcknowledgem
```

In [134]:

```
text = soup.get_text(strip=True)
text=(((((text.replace("\r", " ")).replace("\n", " ")).replace("    ", " ")).replace(''
```

In [135]:

```
print(text)
```

```
SpaceX acquiring satellite data startup Swarm TechnologiesSkip NavigationMar
ketsPreMarketsU.S. MarketsEurope MarketsChina MarketsAsia MarketsWorld Marke
tsCurrenciesCryptocurrencyFutures and CommoditiesBondsFunds and ETFsBusiness
EconomyFinanceHealth and ScienceMediaReal EstateEnergyClimateTransportationI
ndustrialsRetailWealthLifeSmall BusinessInvestingPersonal FinanceFintechFina
ncial AdvisorsOptions ActionETF StreetBuffett ArchiveEarningsTrader TalkTech
CybersecurityEnterpriseInternetMediaMobileSocial MediaCNBC Disruptor 50Tech
GuidePoliticsWhite HousePolicyDefenseCongressEquity and OpportunityEurope Po
liticsChina PoliticsAsia PoliticsWorld PoliticsCNBC TVLive AudioLatest Video
Top VideoCEO InterviewsEurope TVAsia TVCNBC PodcastsDigital OriginalsWatchli
stInvesting ClubTrade AlertsJim's Morning ThoughtsAnalysisMorning MeetingTru
st PortfolioPROPro NewsPro LiveSubscribeSign InMenuMake ItUSAINTLSearch quot
es, news and videosWatchlistSIGN INCreate free accountMarketsBusinessInvesti
ngTechPoliticsCNBC TVWatchlistInvesting ClubPROMenuInvesting in SpaceSpaceX
is buying satellite data startup Swarm, in a rare acquisition by Elon Musk's
space companyPublished Mon, Aug 9 202110:28 AM EDTUpdated Mon, Aug 9 202111:
49 AM EDTMichael Sheetz@thesheetztweetzWATCH LIVEKey PointsSwarm will become
a direct whollyowned subsidiary of SpaceX upon consummation of the Proposed
Transaction, the startup wrote in a Federal Communications Commission filing
Aug. 6.While Swarm's satellites provide a different service from SpaceX's St
arlink broadband network, the acquisition benefits Musk's company by bringin
g access to the intellectual property and expertise of the startup.A Falcon
9 rocket launches the Transporter1 mission in January 2021.SpaceXSpaceX is a
cquiring satellite data startup Swarm Technologies, in a rare deal byElon Mu
sk'sspace company that expands the team — and possibly the technological cap
abilities — of its growing Starlink internet service.Swarm, which has 120 of
its tiny SpaceBEE satellites in orbit, reached an agreement with SpaceX on J
uly 16 to merge, according to an Aug. 6 filing with the Federal Communicatio
ns Commission.The company will become a direct whollyowned subsidiary of Spa
ceX upon consummation of the Proposed Transaction, Swarm wrote in the filin
g.Terms and financial details about the deal were not disclosed. SpaceX and
Swarm did not respond to CNBC's requests for comment. Swarm last completed a
fundraising round in January 2019 at a $85 million valuation, according to P
itchbook.The deal marks an uncommon acquisition for SpaceX, which tends to d
esign and build systems inhouse. But FCC licenses can be difficult and timec
onsuming to get approved, and Swarm will transfer control of of its satellit
e and ground station licenses to SpaceX as part of the deal, according to th
e filing.Swarm's services will benefit from the better capitalization and ac
cess to resources available to SpaceX, as well as the synergies associated w
ith acquisition by a provider of satellite design, manufacture, and launch s
ervices, the filing said.The company noted that the acquisition benefits Spa
ceX by bringing access to the intellectual property and expertise developed
by the Swarm team.Starlink isSpaceX's capitalintensive projectto build an in
terconnected internet network with thousands of satellites — known in the sp
ace industry as a constellation — designed to deliver highspeed internet to
consumers anywhere on the planet. How SpaceX might utilize the Swarm technol
ogy is unclear, as Starlink satellites operate in a different manner compare
d with the SpaceBEEs.Swarm's IoT techFounded in 2016 and headquartered in Mo
untain View, California, Swarm has been building a constellation of 150 sate
llites. Its SpaceBEEs are the smallest commercially operational satellites i
n space, the company says – at 11 centimeters by 11 centimeters by 2.8 centi
meters, the satellites are about the size of a small notebook.Swarm's satell
ites communicate with its antennas on the ground, with a Swarm Tile that can
be embedded into a circuit board, to connect Internet of Things, or IoT, dev
ices to a global communications network. A Swarm Tile is priced at $119, and
its larger standalone Eval Kit is $499, with the company charges a $5 a mont
```

```
h subscription fee to use the network.Eval KitSwarm TechnologiesThe company
offers services for a wide variety of IoT uses, including agriculture, marit
ime, energy, environmental, and transportation sectors.Swarm came under FCC
scrutiny in 2018 afterthe unauthorized launch of its first four SpaceBEE sat
ellites. The FCC ended its investigation with Swarm agreeing to pay a $900,0
00 penalty and implement a fiveyear regulatory compliance plan.Become a smar
ter investor withCNBC Pro.Get stock picks, analyst calls, exclusive intervie
ws and access to CNBC TV.Sign up to start afree trial today.TVWATCH LIVEWATC
H IN THE APPUP NEXT |ETListenTVWATCH LIVEWATCH IN THE APPUP NEXT |ETListenSu
bscribe to CNBC PROLicensing and ReprintsCNBC CouncilsSupply Chain ValuesCNB
C on PeacockJoin the CNBC PanelDigital ProductsNews ReleasesClosed Captionin
gCorrectionsAbout CNBCInternshipsSite MapAd ChoicesCareersHelpContactNews Ti
psGot a confidential news tip? We want to hear from you.Get In TouchAdvertis
e With UsPlease Contact UsCNBC NewslettersSign up for free newsletters and g
et more CNBC delivered to your inboxSign Up NowGet this delivered to your in
box, and more info about our products and services.Privacy Policy|Do Not Sel
l My Personal Information|CA Notice|Terms of Service©2022CNBC LLC. All Right
s Reserved.A Division of NBCUniversalData is a realtime snapshot *Data is de
layed at least 15 minutes. Global Business and Financial News, Stock Quotes,
and Market Data and Analysis.Market Data Terms of Use and DisclaimersData al
so provided by
```

In [136]:

```python
tokens = [t for t in text.split()]
```

In [137]:

```
print(tokens)
```

```
['SpaceX', 'acquiring', 'satellite', 'data', 'startup', 'Swarm', 'Technologi
esSkip', 'NavigationMarketsPreMarketsU.S.', 'MarketsEurope', 'MarketsChina',
'MarketsAsia', 'MarketsWorld', 'MarketsCurrenciesCryptocurrencyFutures', 'an
d', 'CommoditiesBondsFunds', 'and', 'ETFsBusinessEconomyFinanceHealth', 'an
d', 'ScienceMediaReal', 'EstateEnergyClimateTransportationIndustrialsRetailW
ealthLifeSmall', 'BusinessInvestingPersonal', 'FinanceFintechFinancial', 'Ad
visorsOptions', 'ActionETF', 'StreetBuffett', 'ArchiveEarningsTrader', 'Talk
TechCybersecurityEnterpriseInternetMediaMobileSocial', 'MediaCNBC', 'Disrupt
or', '50Tech', 'GuidePoliticsWhite', 'HousePolicyDefenseCongressEquity', 'an
d', 'OpportunityEurope', 'PoliticsChina', 'PoliticsAsia', 'PoliticsWorld',
'PoliticsCNBC', 'TVLive', 'AudioLatest', 'VideoTop', 'VideoCEO', 'Interviews
Europe', 'TVAsia', 'TVCNBC', 'PodcastsDigital', 'OriginalsWatchlistInvestin
g', 'ClubTrade', "AlertsJim's", 'Morning', 'ThoughtsAnalysisMorning', 'Meeti
ngTrust', 'PortfolioPROPro', 'NewsPro', 'LiveSubscribeSign', 'InMenuMake',
'ItUSAINTLSearch', 'quotes,', 'news', 'and', 'videosWatchlistSIGN', 'INCreat
e', 'free', 'accountMarketsBusinessInvestingTechPoliticsCNBC', 'TVWatchlistI
nvesting', 'ClubPROMenuInvesting', 'in', 'SpaceSpaceX', 'is', 'buying', 'sat
ellite', 'data', 'startup', 'Swarm,', 'in', 'a', 'rare', 'acquisition', 'b
y', 'Elon', "Musk's", 'space', 'companyPublished', 'Mon,', 'Aug', '9', '2021
10:28', 'AM', 'EDTUpdated', 'Mon,', 'Aug', '9', '202111:49', 'AM', 'EDTMicha
el', 'Sheetz@thesheetztweetzWATCH', 'LIVEKey', 'PointsSwarm', 'will', 'becom
e', 'a', 'direct', 'whollyowned', 'subsidiary', 'of', 'SpaceX', 'upon', 'con
summation', 'of', 'the', 'Proposed', 'Transaction,', 'the', 'startup', 'wrot
e', 'in', 'a', 'Federal', 'Communications', 'Commission', 'filing', 'Aug.',
'6.While', "Swarm's", 'satellites', 'provide', 'a', 'different', 'service',
'from', "SpaceX's", 'Starlink', 'broadband', 'network,', 'the', 'acquisitio
n', 'benefits', "Musk's", 'company', 'by', 'bringing', 'access', 'to', 'th
e', 'intellectual', 'property', 'and', 'expertise', 'of', 'the', 'startup.
A', 'Falcon', '9', 'rocket', 'launches', 'the', 'Transporter1', 'mission',
'in', 'January', '2021.SpaceXSpaceX', 'is', 'acquiring', 'satellite', 'dat
a', 'startup', 'Swarm', 'Technologies,', 'in', 'a', 'rare', 'deal', 'byElo
n', "Musk'sspace", 'company', 'that', 'expands', 'the', 'team', '—', 'and',
'possibly', 'the', 'technological', 'capabilities', '—', 'of', 'its', 'growi
ng', 'Starlink', 'internet', 'service.Swarm,', 'which', 'has', '120', 'of',
'its', 'tiny', 'SpaceBEE', 'satellites', 'in', 'orbit,', 'reached', 'an', 'a
greement', 'with', 'SpaceX', 'on', 'July', '16', 'to', 'merge,', 'accordin
g', 'to', 'an', 'Aug.', '6', 'filing', 'with', 'the', 'Federal', 'Communicat
ions', 'Commission.The', 'company', 'will', 'become', 'a', 'direct', 'wholly
owned', 'subsidiary', 'of', 'SpaceX', 'upon', 'consummation', 'of', 'the',
'Proposed', 'Transaction,', 'Swarm', 'wrote', 'in', 'the', 'filing.Terms',
'and', 'financial', 'details', 'about', 'the', 'deal', 'were', 'not', 'discl
osed.', 'SpaceX', 'and', 'Swarm', 'did', 'not', 'respond', 'to', "CNBC's",
'requests', 'for', 'comment.', 'Swarm', 'last', 'completed', 'a', 'fundraisi
ng', 'round', 'in', 'January', '2019', 'at', 'a', '$85', 'million', 'valuati
on,', 'according', 'to', 'Pitchbook.The', 'deal', 'marks', 'an', 'uncommon',
'acquisition', 'for', 'SpaceX,', 'which', 'tends', 'to', 'design', 'and', 'b
uild', 'systems', 'inhouse.', 'But', 'FCC', 'licenses', 'can', 'be', 'diffic
ult', 'and', 'timeconsuming', 'to', 'get', 'approved,', 'and', 'Swarm', 'wil
l', 'transfer', 'control', 'of', 'of', 'its', 'satellite', 'and', 'ground',
'station', 'licenses', 'to', 'SpaceX', 'as', 'part', 'of', 'the', 'deal,',
'according', 'to', 'the', "filing.Swarm's", 'services', 'will', 'benefit',
'from', 'the', 'better', 'capitalization', 'and', 'access', 'to', 'resource
s', 'available', 'to', 'SpaceX,', 'as', 'well', 'as', 'the', 'synergies', 'a
ssociated', 'with', 'acquisition', 'by', 'a', 'provider', 'of', 'satellite',
'design,', 'manufacture,', 'and', 'launch', 'services,', 'the', 'filing', 's
aid.The', 'company', 'noted', 'that', 'the', 'acquisition', 'benefits', 'Spa
ceX', 'by', 'bringing', 'access', 'to', 'the', 'intellectual', 'property',
'and', 'expertise', 'developed', 'by', 'the', 'Swarm', 'team.Starlink', "isS
paceX's", 'capitalintensive', 'projectto', 'build', 'an', 'interconnected',
'internet', 'network', 'with', 'thousands', 'of', 'satellites', '–', 'know
n', 'in', 'the', 'space', 'industry', 'as', 'a', 'constellation', '–', 'desi
gned', 'to', 'deliver', 'highspeed', 'internet', 'to', 'consumers', 'anywher
e', 'on', 'the', 'planet.', 'How', 'SpaceX', 'might', 'utilize', 'the', 'Swa
rm', 'technology', 'is', 'unclear,', 'as', 'Starlink', 'satellites', 'operat
e', 'in', 'a', 'different', 'manner', 'compared', 'with', 'the', "SpaceBEEs.
Swarm's", 'IoT', 'techFounded', 'in', '2016', 'and', 'headquartered', 'in',
'Mountain', 'View,', 'California,', 'Swarm', 'has', 'been', 'building', 'a',
'constellation', 'of', '150', 'satellites.', 'Its', 'SpaceBEEs', 'are', 'th
e', 'smallest', 'commercially', 'operational', 'satellites', 'in', 'space,',
'the', 'company', 'says', '-', 'at', '11', 'centimeters', 'by', '11', 'centi
meters', 'by', '2.8', 'centimeters,', 'the', 'satellites', 'are', 'about',
'the', 'size', 'of', 'a', 'small', "notebook.Swarm's", 'satellites', 'commun
icate', 'with', 'its', 'antennas', 'on', 'the', 'ground,', 'with', 'a', 'Swa
rm', 'Tile', 'that', 'can', 'be', 'embedded', 'into', 'a', 'circuit', 'boar
d,', 'to', 'connect', 'Internet', 'of', 'Things,', 'or', 'IoT,', 'devices',
'to', 'a', 'global', 'communications', 'network.', 'A', 'Swarm', 'Tile', 'i
s', 'priced', 'at', '$119,', 'and', 'its', 'larger', 'standalone', 'Eval',
'Kit', 'is', '$499,', 'with', 'the', 'company', 'charges', 'a', '$5', 'a',
'month', 'subscription', 'fee', 'to', 'use', 'the', 'network.Eval', 'KitSwar
m', 'TechnologiesThe', 'company', 'offers', 'services', 'for', 'a', 'wide',
'variety', 'of', 'IoT', 'uses,', 'including', 'agriculture,', 'maritime,',
'energy,', 'environmental,', 'and', 'transportation', 'sectors.Swarm', 'cam
e', 'under', 'FCC', 'scrutiny', 'in', '2018', 'afterthe', 'unauthorized', 'l
aunch', 'of', 'its', 'first', 'four', 'SpaceBEE', 'satellites.', 'The', 'FC
C', 'ended', 'its', 'investigation', 'with', 'Swarm', 'agreeing', 'to', 'pa
y', 'a', '$900,000', 'penalty', 'and', 'implement', 'a', 'fiveyear', 'regula
tory', 'compliance', 'plan.Become', 'a', 'smarter', 'investor', 'withCNBC',
'Pro.Get', 'stock', 'picks,', 'analyst', 'calls,', 'exclusive', 'interview
s', 'and', 'access', 'to', 'CNBC', 'TV.Sign', 'up', 'to', 'start', 'afree',
'trial', 'today.TVWATCH', 'LIVEWATCH', 'IN', 'THE', 'APPUP', 'NEXT', '|ETLis
tenTVWATCH', 'LIVEWATCH', 'IN', 'THE', 'APPUP', 'NEXT', '|ETListenSubscrib
e', 'to', 'CNBC', 'PROLicensing', 'and', 'ReprintsCNBC', 'CouncilsSupply',
'Chain', 'ValuesCNBC', 'on', 'PeacockJoin', 'the', 'CNBC', 'PanelDigital',
'ProductsNews', 'ReleasesClosed', 'CaptioningCorrectionsAbout', 'CNBCInterns
hipsSite', 'MapAd', 'ChoicesCareersHelpContactNews', 'TipsGot', 'a', 'confid
ential', 'news', 'tip?', 'We', 'want', 'to', 'hear', 'from', 'you.Get', 'I
n', 'TouchAdvertise', 'With', 'UsPlease', 'Contact', 'UsCNBC', 'NewslettersS
ign', 'up', 'for', 'free', 'newsletters', 'and', 'get', 'more', 'CNBC', 'del
ivered', 'to', 'your', 'inboxSign', 'Up', 'NowGet', 'this', 'delivered', 't
o', 'your', 'inbox,', 'and', 'more', 'info', 'about', 'our', 'products', 'an
d', 'services.Privacy', 'Policy|Do', 'Not', 'Sell', 'My', 'Personal', 'Infor
mation|CA', 'Notice|Terms', 'of', 'Service©2022CNBC', 'LLC.', 'All', 'Right
s', 'Reserved.A', 'Division', 'of', 'NBCUniversalData', 'is', 'a', 'realtim
e', 'snapshot', '*Data', 'is', 'delayed', 'at', 'least', '15', 'minutes.',
'Global', 'Business', 'and', 'Financial', 'News,', 'Stock', 'Quotes,', 'an
d', 'Market', 'Data', 'and', 'Analysis.Market', 'Data', 'Terms', 'of', 'Us
e', 'and', 'DisclaimersData', 'also', 'provided', 'by']
```

In [138]:

```
clean_tokens = tokens[:]
```

In [139]:

```
clean_tokens
```

Out[139]:

```
['SpaceX',
 'acquiring',
 'satellite',
 'data',
 'startup',
 'Swarm',
 'TechnologiesSkip',
 'NavigationMarketsPreMarketsU.S.',
 'MarketsEurope',
 'MarketsChina',
 'MarketsAsia',
 'MarketsWorld',
 'MarketsCurrenciesCryptocurrencyFutures',
 'and',
 'CommoditiesBondsFunds',
 'and',
 'ETFsBusinessEconomyFinanceHealth',
 'and'.
```

In [140]:

```
sr = stopwords.words('english')
```

In [141]:

```
for token in tokens:
    if token in stopwords.words('english'):
        clean_tokens.remove(token)
```

In [142]:

```
print(clean_tokens)
```

```
['SpaceX', 'acquiring', 'satellite', 'data', 'startup', 'Swarm', 'Technologi
esSkip', 'NavigationMarketsPreMarketsU.S.', 'MarketsEurope', 'MarketsChina',
'MarketsAsia', 'MarketsWorld', 'MarketsCurrenciesCryptocurrencyFutures', 'Co
mmoditiesBondsFunds', 'ETFsBusinessEconomyFinanceHealth', 'ScienceMediaRea
l', 'EstateEnergyClimateTransportationIndustrialsRetailWealthLifeSmall', 'Bu
sinessInvestingPersonal', 'FinanceFintechFinancial', 'AdvisorsOptions', 'Act
ionETF', 'StreetBuffett', 'ArchiveEarningsTrader', 'TalkTechCybersecurityEnt
erpriseInternetMediaMobileSocial', 'MediaCNBC', 'Disruptor', '50Tech', 'Guid
ePoliticsWhite', 'HousePolicyDefenseCongressEquity', 'OpportunityEurope', 'P
oliticsChina', 'PoliticsAsia', 'PoliticsWorld', 'PoliticsCNBC', 'TVLive', 'A
udioLatest', 'VideoTop', 'VideoCEO', 'InterviewsEurope', 'TVAsia', 'TVCNBC',
'PodcastsDigital', 'OriginalsWatchlistInvesting', 'ClubTrade', "AlertsJi
m's", 'Morning', 'ThoughtsAnalysisMorning', 'MeetingTrust', 'PortfolioPROPr
o', 'NewsPro', 'LiveSubscribeSign', 'InMenuMake', 'ItUSAINTLSearch', 'quote
s,', 'news', 'videosWatchlistSIGN', 'INCreate', 'free', 'accountMarketsBusin
essInvestingTechPoliticsCNBC', 'TVWatchlistInvesting', 'ClubPROMenuInvestin
g', 'SpaceSpaceX', 'buying', 'satellite', 'data', 'startup', 'Swarm,', 'rar
e', 'acquisition', 'Elon', "Musk's", 'space', 'companyPublished', 'Mon,', 'A
ug', '9', '202110:28', 'AM', 'EDTUpdated', 'Mon,', 'Aug', '9', '202111:49',
'AM', 'EDTMichael', 'Sheetz@thesheetztweetzWATCH', 'LIVEKey', 'PointsSwarm',
'become', 'direct', 'whollyowned', 'subsidiary', 'SpaceX', 'upon', 'consumma
tion', 'Proposed', 'Transaction,', 'startup', 'wrote', 'Federal', 'Communica
tions', 'Commission', 'filing', 'Aug.', '6.While', "Swarm's", 'satellites',
'provide', 'different', 'service', "SpaceX's", 'Starlink', 'broadband', 'net
work,', 'acquisition', 'benefits', "Musk's", 'company', 'bringing', 'acces
s', 'intellectual', 'property', 'expertise', 'startup.A', 'Falcon', '9', 'ro
cket', 'launches', 'Transporter1', 'mission', 'January', '2021.SpaceXSpace
X', 'acquiring', 'satellite', 'data', 'startup', 'Swarm', 'Technologies,',
'rare', 'deal', 'byElon', "Musk'sspace", 'company', 'expands', 'team', '—',
'possibly', 'technological', 'capabilities', '—', 'growing', 'Starlink', 'in
ternet', 'service.Swarm,', '120', 'tiny', 'SpaceBEE', 'satellites', 'orbi
t,', 'reached', 'agreement', 'SpaceX', 'July', '16', 'merge,', 'according',
'Aug.', '6', 'filing', 'Federal', 'Communications', 'Commission.The', 'compa
ny', 'become', 'direct', 'whollyowned', 'subsidiary', 'SpaceX', 'upon', 'con
summation', 'Proposed', 'Transaction,', 'Swarm', 'wrote', 'filing.Terms', 'f
inancial', 'details', 'deal', 'disclosed.', 'SpaceX', 'Swarm', 'respond', "C
NBC's", 'requests', 'comment.', 'Swarm', 'last', 'completed', 'fundraising',
'round', 'January', '2019', '$85', 'million', 'valuation,', 'according', 'Pi
tchbook.The', 'deal', 'marks', 'uncommon', 'acquisition', 'SpaceX,', 'tend
s', 'design', 'build', 'systems', 'inhouse.', 'But', 'FCC', 'licenses', 'dif
ficult', 'timeconsuming', 'get', 'approved,', 'Swarm', 'transfer', 'contro
l', 'satellite', 'ground', 'station', 'licenses', 'SpaceX', 'part', 'deal,',
'according', "filing.Swarm's", 'services', 'benefit', 'better', 'capitalizat
ion', 'access', 'resources', 'available', 'SpaceX,', 'well', 'synergies', 'a
ssociated', 'acquisition', 'provider', 'satellite', 'design,', 'manufactur
e,', 'launch', 'services,', 'filing', 'said.The', 'company', 'noted', 'acqui
sition', 'benefits', 'SpaceX', 'bringing', 'access', 'intellectual', 'proper
ty', 'expertise', 'developed', 'Swarm', 'team.Starlink', "isSpaceX's", 'capi
talintensive', 'projectto', 'build', 'interconnected', 'internet', 'networ
k', 'thousands', 'satellites', '—', 'known', 'space', 'industry', 'constella
tion', '—', 'designed', 'deliver', 'highspeed', 'internet', 'consumers', 'an
ywhere', 'planet.', 'How', 'SpaceX', 'might', 'utilize', 'Swarm', 'technolog
y', 'unclear,', 'Starlink', 'satellites', 'operate', 'different', 'manner',
'compared', "SpaceBEEs.Swarm's", 'IoT', 'techFounded', '2016', 'headquartere
d', 'Mountain', 'View,', 'California,', 'Swarm', 'building', 'constellatio
n', '150', 'satellites.', 'Its', 'SpaceBEEs', 'smallest', 'commercially', 'o
perational', 'satellites', 'space,', 'company', 'says', '-', '11', 'centimet
ers', '11', 'centimeters', '2.8', 'centimeters,', 'satellites', 'size', 'sma
ll', "notebook.Swarm's", 'satellites', 'communicate', 'antennas', 'ground,',
'Swarm', 'Tile', 'embedded', 'circuit', 'board,', 'connect', 'Internet', 'Th
ings,', 'IoT,', 'devices', 'global', 'communications', 'network.', 'A', 'Swa
rm', 'Tile', 'priced', '$119,', 'larger', 'standalone', 'Eval', 'Kit', '$49
9,', 'company', 'charges', '$5', 'month', 'subscription', 'fee', 'use', 'net
work.Eval', 'KitSwarm', 'TechnologiesThe', 'company', 'offers', 'services',
'wide', 'variety', 'IoT', 'uses,', 'including', 'agriculture,', 'maritime,',
'energy,', 'environmental,', 'transportation', 'sectors.Swarm', 'came', 'FC
C', 'scrutiny', '2018', 'afterthe', 'unauthorized', 'launch', 'first', 'fou
r', 'SpaceBEE', 'satellites.', 'The', 'FCC', 'ended', 'investigation', 'Swar
m', 'agreeing', 'pay', '$900,000', 'penalty', 'implement', 'fiveyear', 'regu
latory', 'compliance', 'plan.Become', 'smarter', 'investor', 'withCNBC', 'Pr
o.Get', 'stock', 'picks,', 'analyst', 'calls,', 'exclusive', 'interviews',
'access', 'CNBC', 'TV.Sign', 'start', 'afree', 'trial', 'today.TVWATCH', 'LI
VEWATCH', 'IN', 'THE', 'APPUP', 'NEXT', '|ETListenTVWATCH', 'LIVEWATCH', 'I
N', 'THE', 'APPUP', 'NEXT', '|ETListenSubscribe', 'CNBC', 'PROLicensing', 'R
eprintsCNBC', 'CouncilsSupply', 'Chain', 'ValuesCNBC', 'PeacockJoin', 'CNB
C', 'PanelDigital', 'ProductsNews', 'ReleasesClosed', 'CaptioningCorrections
About', 'CNBCInternshipsSite', 'MapAd', 'ChoicesCareersHelpContactNews', 'Ti
psGot', 'confidential', 'news', 'tip?', 'We', 'want', 'hear', 'you.Get', 'I
n', 'TouchAdvertise', 'With', 'UsPlease', 'Contact', 'UsCNBC', 'NewslettersS
ign', 'free', 'newsletters', 'get', 'CNBC', 'delivered', 'inboxSign', 'Up',
'NowGet', 'delivered', 'inbox,', 'info', 'products', 'services.Privacy', 'Po
licy|Do', 'Not', 'Sell', 'My', 'Personal', 'Information|CA', 'Notice|Terms',
'Service©2022CNBC', 'LLC.', 'All', 'Rights', 'Reserved.A', 'Division', 'NBCU
niversalData', 'realtime', 'snapshot', '*Data', 'delayed', 'least', '15', 'm
inutes.', 'Global', 'Business', 'Financial', 'News,', 'Stock', 'Quotes,', 'M
arket', 'Data', 'Analysis.Market', 'Data', 'Terms', 'Use', 'DisclaimersDat
a', 'also', 'provided']
```

In [143]:

```
freq = nltk.FreqDist(clean_tokens)
```

In [144]:

```
print(freq)
```

```
<FreqDist with 407 samples and 521 outcomes>
```

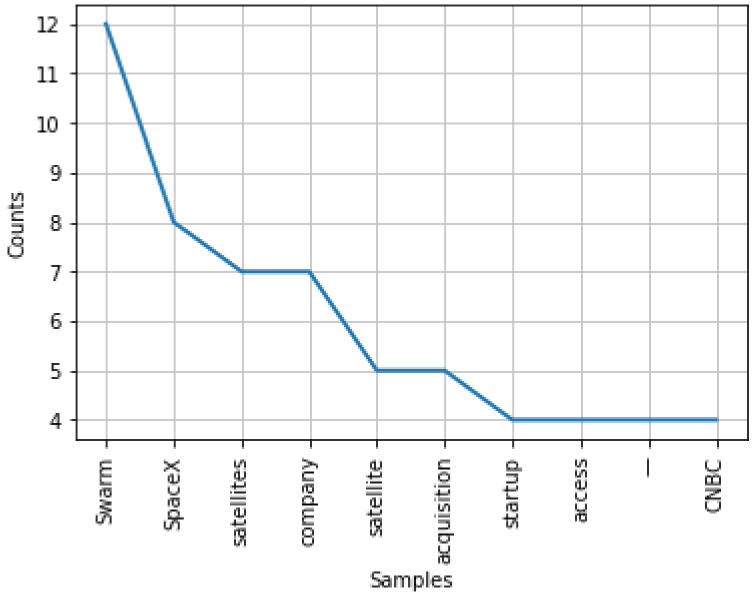
In [145]:

```
for key,val in freq.items():
    print (str(key) + ':' + str(val))
```

```
SpaceX:8
acquiring:2
satellite:5
data:3
startup:4
Swarm:12
TechnologiesSkip:1
NavigationMarketsPreMarketsU.S.:1
MarketsEurope:1
MarketsChina:1
MarketsAsia:1
MarketsWorld:1
MarketsCurrenciesCryptocurrencyFutures:1
CommoditiesBondsFunds:1
ETFsBusinessEconomyFinanceHealth:1
ScienceMediaReal:1
EstateEnergyClimateTransportationIndustrialsRetailWealthLifeSmall:1
BusinessInvestingPersonal:1
FinanceFintechFinancial:1
```

In [146]:

```
freq.plot(10, cumulative= False)
```



Out[146]:

```
<AxesSubplot:xlabel='Samples', ylabel='Counts'>
```

Different visualization for frequency distribution

In [182]:

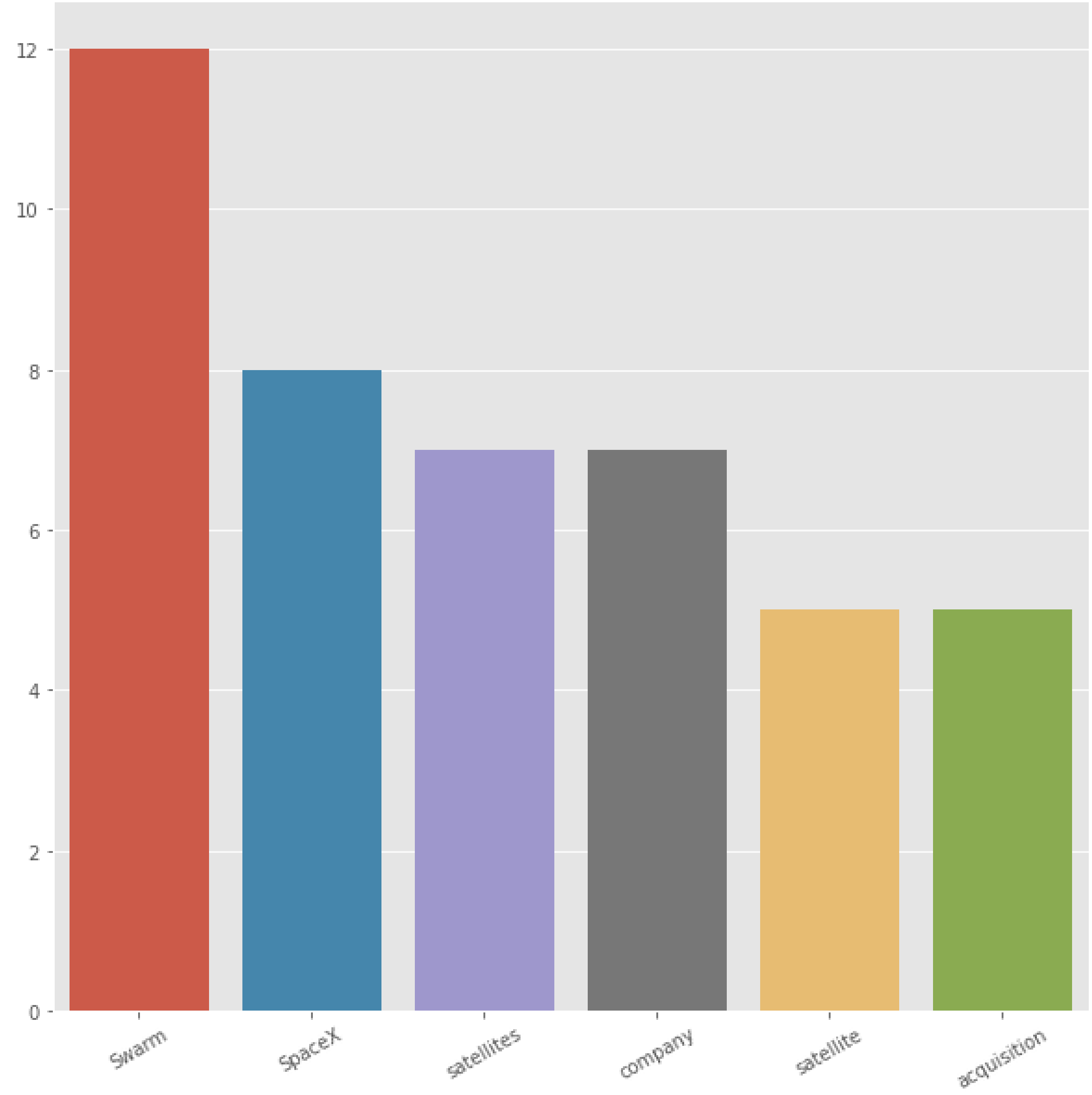
```

## Creating FreqDist for whole BoW, keeping the 10 most common tokens
all_fdist = freq.most_common(6)

## Conversion to Pandas series via Python Dictionary for easier plotting
all_fdist = pd.Series(dict(all_fdist))

## Setting figure, ax into variables
fig, ax = plt.subplots(figsize=(10,10))

## Seaborn plotting using Pandas attributes + xtick rotation for ease of viewing
all_plot = sns.barplot(x=all_fdist.index, y=all_fdist.values, ax=ax)
plt.xticks(rotation=30);
```



In [159]:

```
#print(list(freq.keys()))
#for i in range(0,10):
print([k for k,v in freq.items() if v >= 5])
```

```
['SpaceX', 'satellite', 'Swarm', 'acquisition', 'satellites', 'company']
```

Questions

1)Why we use stopwords? Why stopwords are not necessary for NLP frequency distribution

Answer) Stop words are unnecessary words or can we said as set of very common words which has no vaild information but without them sentences have no proper form and meaning.For freqency distribution in NLP we just count the number of word repeated where those words are not important they are just noise.So, we remove those stopwords.

2)Based on high frequency words what information you can extract from the graph?

Answer) High frequency words are those words which are repeated several times in a passage, which means they could carry high importance in that passage.We could know that the passage is about.

3) Can you provide different visualization for frequency distribution? If yes, please perform. If no, why?

Answer) Please find the visualization chart above

In [ ]:

```

```