

In [1]:

```
# Importing the Python libraries
# Polarity classification - Opinion Mining - Sentiment Analysis
import nltk
import pandas as pd
import wordcloud
from matplotlib import pyplot as plt
from nltk.sentiment.vader import SentimentIntensityAnalyzer
from wordcloud import WordCloud
from datetime import date, datetime

%matplotlib inline
```

```
/opt/conda/lib/python3.7/site-packages/nltk/twitter/__init__.py:20: Us
erWarning: The twython library has not been installed. Some functional
ity from the twitter package will not be available.
  warnings.warn("The twython library has not been installed. "
```

In [2]:

```
df = pd.read_csv("../input/consumer-reviews-of-amazon-products/Datafiniti_Amazon_Consumer_Reviews_of_Amazon_Products.csv")
df.head()
```

Out[2]:

	id	dateAdded	dateUpdated	name	asins	brand	
0	AVqVGZNvQMIgsOJE6eUY	2017-03-03T16:56:05Z	2018-10-25T16:36:31Z	Amazon Kindle E-Reader 6" Wifi (8th Generation...)	B00ZV9PXP2	Amazon	(Fea
1	AVqVGZNvQMIgsOJE6eUY	2017-03-03T16:56:05Z	2018-10-25T16:36:31Z	Amazon Kindle E-Reader 6" Wifi (8th Generation...)	B00ZV9PXP2	Amazon	(Fea
2	AVqVGZNvQMIgsOJE6eUY	2017-03-03T16:56:05Z	2018-10-25T16:36:31Z	Amazon Kindle E-Reader 6" Wifi (8th Generation...)	B00ZV9PXP2	Amazon	(Fea
3	AVqVGZNvQMIgsOJE6eUY	2017-03-03T16:56:05Z	2018-10-25T16:36:31Z	Amazon Kindle E-Reader 6" Wifi (8th Generation...)	B00ZV9PXP2	Amazon	(Fea
4	AVqVGZNvQMIgsOJE6eUY	2017-03-03T16:56:05Z	2018-10-25T16:36:31Z	Amazon Kindle E-Reader 6" Wifi (8th Generation...)	B00ZV9PXP2	Amazon	(Fea

5 rows × 24 columns

In [3]:

```
mydf = df[['id', 'name', 'categories', 'reviews.text', 'reviews.title', 'reviews.use
mydf.head()
```

Out[3]:

	id	name	categories	reviews.text	reviews.title	revi
0	AVqVGZNvQMlgsOJE6eUY	Amazon Kindle E-Reader 6" Wifi (8th Generation...	Computers,Electronics Features,Tablets,Electro...	I thought it would be as big as small paper bu...	Too small	
1	AVqVGZNvQMlgsOJE6eUY	Amazon Kindle E-Reader 6" Wifi (8th Generation...	Computers,Electronics Features,Tablets,Electro...	This kindle is light and easy to use especiall...	Great light reader. Easy to use at the beach	
2	AVqVGZNvQMlgsOJE6eUY	Amazon Kindle E-Reader 6" Wifi (8th Generation...	Computers,Electronics Features,Tablets,Electro...	Didnt know how much i'd use a kindle so went f...	Great for the price	
3	AVqVGZNvQMlgsOJE6eUY	Amazon Kindle E-Reader 6" Wifi (8th Generation...	Computers,Electronics Features,Tablets,Electro...	I am 100 happy with my purchase. I caught it o...	A Great Buy	
4	AVqVGZNvQMlgsOJE6eUY	Amazon Kindle E-Reader 6" Wifi (8th Generation...	Computers,Electronics Features,Tablets,Electro...	Solid entry level Kindle. Great for kids. Gift...	Solid entry-level Kindle. Great for kids	

In [4]:

```
def opinion(output, threshold):
    if output['neg'] > threshold:
        return 'neg_count'
    elif output['pos'] > threshold:
        return 'pos_count'
    return 'neu_count'
```

In [5]:

```
def get_season(now):
    if isinstance(now, datetime):
        now = now.date()
    now = now.replace(year=Y)
    return next(season for season, (start, end) in seasons if start <= now <= end)
```

In [6]:

```

sia = SentimentIntensityAnalyzer()

mydict = {}
threshold = 0.2

for index, row in df.iterrows():
    ps = sia.polarity_scores(row['reviews.text'])
    if row['id'] not in mydict:
        mydict[row['id']] = ps
        mydict[row['id']]['count'] = 0
        mydict[row['id']]['neg_count'] = 0
        mydict[row['id']]['pos_count'] = 0
        mydict[row['id']]['neu_count'] = 0

        mydict[row['id']]['neg'] += ps['neg']
        mydict[row['id']]['neu'] += ps['neu']
        mydict[row['id']]['pos'] += ps['pos']
        mydict[row['id']]['compound'] += ps['compound']
        mydict[row['id']]['count'] += 1
        mydict[row['id']]['opinion(ps, threshold)] += 1

print(mydict)

{'AVqVGZNVQmLgsOJE6eUY': {'neg': 2.6570000000000005, 'neu': 70.25500000000001, 'pos': 24.091, 'compound': 58.8058, 'count': 96, 'neg_count': 1, 'pos_count': 56, 'neu_count': 39}, 'AWFUWc8THh53nbDRF6YO': {'neg': 10.24, 'neu': 459.1859999999999, 'pos': 181.56199999999995, 'compound': 459.7326, 'count': 650, 'neg_count': 6, 'pos_count': 437, 'neu_count': 207}, 'AWK8z0pOIwln0LfXlSxH': {'neg': 3.8190000000000004, 'neu': 139.94000000000008, 'pos': 52.233000000000002, 'compound': 133.53910000000002, 'count': 195, 'neg_count': 2, 'pos_count': 139, 'neu_count': 54}, 'AV-XeQLWuClrwyj_gbP5': {'neg': 0.536, 'neu': 3.0399999999999996, 'pos': 1.424, 'compound': 2.1677, 'count': 4, 'neg_count': 0, 'pos_count': 3, 'neu_count': 1}, 'AWMjT0WguClrwyj_rFh3': {'neg': 9.675, 'neu': 410.86100000000001, 'pos': 170.45999999999999, 'compound': 397.43310000000065, 'count': 590, 'neg_count': 7, 'pos_count': 410, 'neu_count': 173}, 'AVpfIfGAlcnluZ0-emyp': {'neg': 2.377, 'neu': 14.475, 'pos': 6.148, 'compound': 9.581100000000001, 'count': 22, 'neg_count': 4, 'pos_count': 15, 'neu_count': 3}, 'AVph0EeEilAPnD_x9myq': {'neg': 14.414000000000007, 'neu': 380.59399999999996, 'pos': 166.97499999999998, 'compound': 377.40810000000003, 'count': 561, 'neg_count': 7, 'pos_count': 405, 'neu_count': 149}, 'AVqVGZSEQmLgsOJE6eUc': {'neg': 3.9339999999999993, 'neu': 113.45200000000004, 'pos': 42.61499999999998, 'compound': 98.77089999999994, 'count': 159, 'neg_count': 2, 'pos_count': 99, 'neu_count': 58}, 'AVpgdkC8ilAPnD_xsvyi': {'neg': 12.849999999999998, 'neu': 257.01699999999999, 'pos': 102.13499999999992, 'compound': 219.22170000000023, 'count': 371, 'neg_count': 10, 'pos_count': 237, 'neu_count': 124}, 'AVqVGWLKnnclJgDc3jF1': {'neg': 5.813999999999998, 'neu': 146.92100000000005, 'pos': 65.25999999999999, 'compound': 142.49269999999996, 'count': 217, 'neg_count': 1, 'pos_count': 162, 'neu_count': 54}, 'AVqkIhkhv8e3D10-lebZ': {'neg': 1.984, 'neu': 32.444, 'pos': 17.574000000000005, 'compound': 35.2718, 'count': 51, 'neg_count': 1, 'pos_count': 36, 'neu_count': 14}, 'AVqkIh9HQmLgsOJE6fu_': {'neg': 2.7170000000000005, 'neu': 50.902, 'pos': 17.38, 'compound': 39.96709999999999, 'count': 70, 'neg_count': 0, 'pos_count': 40, 'neu_count': 30}, 'AVpfpK8KLJeJML43BCuD': {'neg': 6.1030000000000015, 'neu': 158.42199999999985, 'pos': 61.47900000000003, 'compound': 149.93450000000013, 'count': 225, 'neg_count': 1, 'pos_count': 147, 'neu_count': 77}, 'AVpidLjVilAPnD_xEVpI': {'neg': 4.468000000000001, 'neu': 74.814, 'pos': 27.724

```

```
000000000007, 'compound': 64.8389, 'count': 106, 'neg_count': 4, 'pos_
count': 69, 'neu_count': 33}, 'AVqVGZ03nnclJgDc3jGK': {'neg': 0.493,
'neu': 18.445999999999998, 'pos': 6.059999999999999, 'compound': 20.59
6299999999996, 'count': 24, 'neg_count': 0, 'pos_count': 14, 'neu_coun
t': 10}, 'AVphPmHuilaPnD_x3E5h': {'neg': 2.4589999999999996, 'neu': 5
9.395, 'pos': 21.151999999999997, 'compound': 58.715499999999956, 'cou
nt': 82, 'neg_count': 1, 'pos_count': 54, 'neu_count': 27}, 'AVqkIhwDv
8e3D10-lebb': {'neg': 25.598999999999993, 'neu': 550.35199999999997, 'p
os': 222.06300000000005, 'compound': 504.800100000000027, 'count': 797,
'neg_count': 22, 'pos_count': 527, 'neu_count': 248}, 'AVpjEN4jLJeJML4
3rpUe': {'neg': 15.889000000000005, 'neu': 329.43499999999995, 'pos':
122.67699999999995, 'compound': 265.41099999999999, 'count': 467, 'neg_
count': 16, 'pos_count': 290, 'neu_count': 161}, 'AVpftoiijlcnluz0-p5n
2': {'neg': 0.9830000000000002, 'neu': 16.420000000000005, 'pos': 5.59
5, 'compound': 14.7948, 'count': 22, 'neg_count': 2, 'pos_count': 13,
'neu_count': 7}, 'AVqkIj9snnclJgDc3khU': {'neg': 2.026, 'neu': 39.5829
99999999984, 'pos': 12.394, 'compound': 31.0861, 'count': 53, 'neg_cou
nt': 0, 'pos_count': 28, 'neu_count': 25}, 'AVqVGZN9QMIgsOJE6eUZ': {'n
eg': 0.56, 'neu': 29.359999999999996, 'pos': 10.08, 'compound': 31.544
900000000005, 'count': 39, 'neg_count': 0, 'pos_count': 27, 'neu_coun
t': 12}, 'AVqkIh8WQMIgsOJE6fu-': {'neg': 2.1330000000000005, 'neu': 4
2.993999999999998, 'pos': 13.874999999999998, 'compound': 35.466400000
0001, 'count': 58, 'neg_count': 0, 'pos_count': 33, 'neu_count': 25},
'AVqkIiKWnnclJgDc3khH': {'neg': 2.03, 'neu': 29.819999999999997, 'po
s': 9.151000000000002, 'compound': 22.764099999999996, 'count': 40, 'n
eg_count': 4, 'pos_count': 20, 'neu_count': 16}, 'AVqkIdZiv8e3D10-lea
J': {'neg': 3.9899999999999998, 'neu': 73.52199999999998, 'pos': 24.48
8000000000007, 'compound': 51.46799999999999, 'count': 101, 'neg_coun
t': 5, 'pos_count': 60, 'neu_count': 36}}
```

In [7]:

```
mydf2 = pd.DataFrame(mydict.values())
mydf2["id"] = mydict.keys()
mydf2 = mydf2.merge(mydf[["id", "name", "categories"]].drop_duplicates(), how = "left")
mydf2
```

Out[7]:

	neg	neu	pos	compound	count	neg_count	pos_count	neu_count	id
0	2.657	70.255	24.091	58.8058	96	1	56	39	AVqVGZNvQMIgsOJE6eUY
1	10.240	459.186	181.562	459.7326	650	6	437	207	AWFUWc8THh53nbDRF6YO
2	3.819	139.940	52.233	133.5391	195	2	139	54	AWK8z0pOlwn0LfXISxH

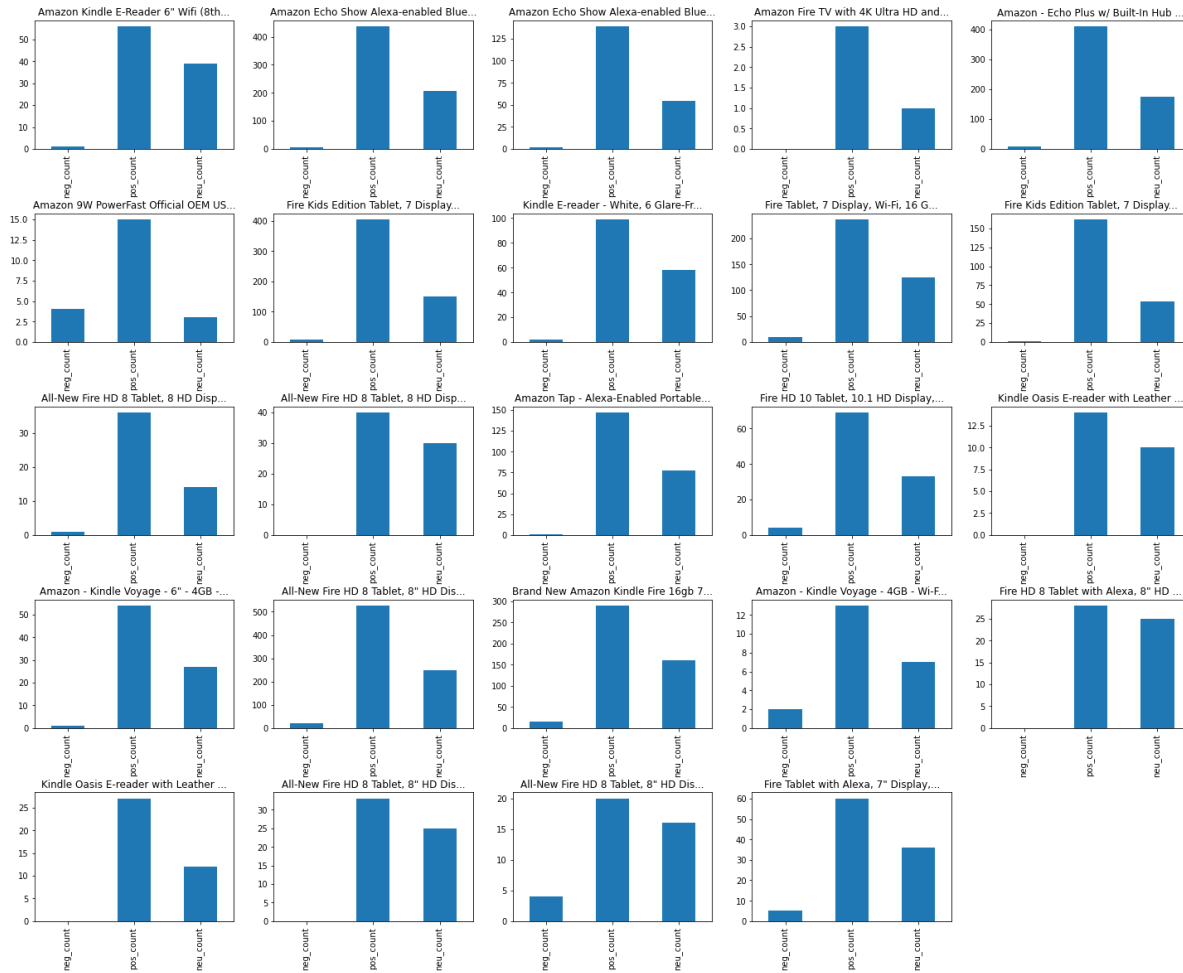
In [8]:

```
plt.figure(figsize=(25, 20))
plt.subplots_adjust(hspace=0.5)
plt.suptitle("Sentiment Category Count by Product", fontsize=18, y=0.95)

for n, row in mydf2.iterrows():
    ax = plt.subplot(5, 5, n + 1)
    row[["neg_count", "pos_count", "neu_count"]].plot.bar(ax=ax)

# chart formatting
ax.set_title(row[9][:35] + "...")
```

Sentiment Category Count by Product



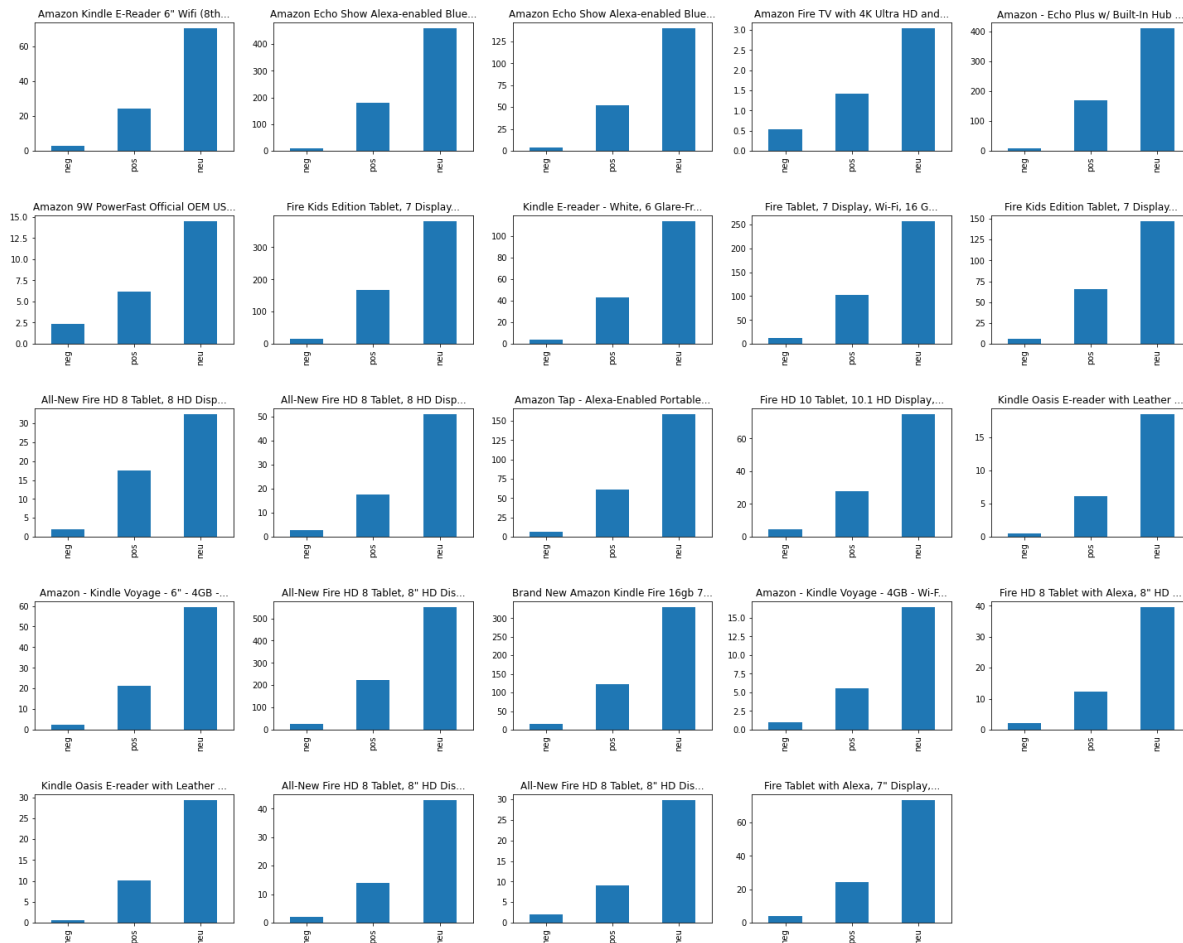
In [9]:

```
plt.figure(figsize=(25, 20))
plt.subplots_adjust(hspace=0.5)
plt.suptitle("Sentiment Sum by Product", fontsize=18, y=0.95)

for n, row in mydf2.iterrows():
    ax = plt.subplot(5, 5, n + 1)
    row[["neg", "pos", "neu"]].plot.bar(ax=ax)

    # chart formatting
    ax.set_title(row[9][:35] + "...")
```

Sentiment Sum by Product

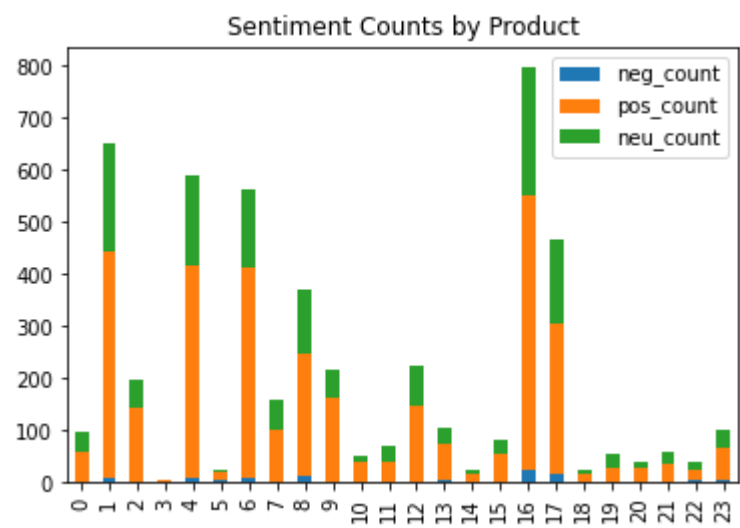


In [10]:

```
mydf2[["neg_count", "pos_count", "neu_count"]].plot.bar(stacked=True, title= "Sentin
```

Out[10]:

<AxesSubplot:title={'center': 'Sentiment Counts by Product'}>

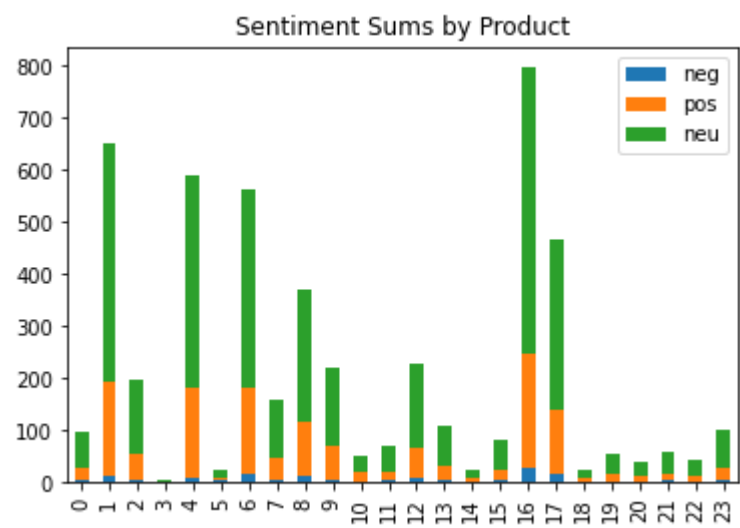


In [11]:

```
mydf2[["neg", "pos", "neu"]].plot.bar(stacked=True, title= "Sentiment Sums by Product")
```

Out[11]:

<AxesSubplot:title={'center': 'Sentiment Sums by Product'}>





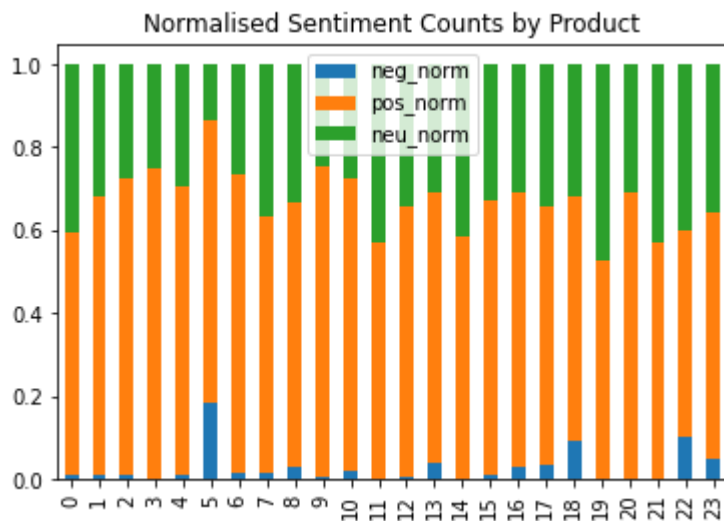
In [12]:

```
mydf2["neg_norm"] = mydf2["neg_count"] / mydf2["count"]
mydf2["pos_norm"] = mydf2["pos_count"] / mydf2["count"]
mydf2["neu_norm"] = mydf2["neu_count"] / mydf2["count"]

mydf2[["neg_norm", "pos_norm", "neu_norm"]].plot.bar(stacked=True, title= "Normalise
```

Out[12]:

```
<AxesSubplot:title={'center':'Normalised Sentiment Counts by Product'}>
```



In [13]:

```
# Make the same grid but with wordclouds
plt.figure(figsize=(25, 20))
plt.subplots_adjust(hspace=0.5)
plt.suptitle("Wordcloud by Product", fontsize=18, y=0.95)

for n, row in mydf2.iterrows():
    ax = plt.subplot(5, 5, n + 1)
    text = " ".join(list(mydf.loc[df['id'] == row.id]["reviews.text"]))

    wordcloud = WordCloud(max_font_size=100, max_words=100, background_color="white")

    # Display the generated image:
    plt.imshow(wordcloud, interpolation='bilinear')
    plt.title(row[9][:35] + "...")
    plt.axis("off")
```

Wordcloud by Product





In [14]:

```
# Make the same grid but with wordclouds
plt.figure(figsize=(25, 20))
plt.subplots_adjust(hspace=0.5)
plt.suptitle("Positive Review Wordcloud by Product", fontsize=18, y=0.95)

for n, row in mydf2.iterrows():
    ax = plt.subplot(5, 5, n + 1)
    reviews = list(mydf.loc[df['id'] == row.id]["reviews.text"])
    text = ""
    for rev in reviews:
        ps = sia.polarity_scores(rev)
        if opinion(ps, threshold) == "pos_count":
            text += rev

    wordcloud = WordCloud(max_font_size=100, max_words=100, background_color="white")

    # Display the generated image:
    plt.imshow(wordcloud, interpolation='bilinear')
    plt.title(row[9][:35] + "...")
    plt.axis("off")
```

Positive Review Wordcloud by Product





In [15]:

```
# Make the same grid but with wordclouds
plt.figure(figsize=(25, 20))
plt.subplots_adjust(hspace=0.5)
plt.suptitle("Negative Review Wordcloud by Product", fontsize=18, y=0.95)

for n, row in mydf2.iterrows():
    ax = plt.subplot(5, 5, n + 1)
    reviews = list(mydf.loc[df['id'] == row.id]['reviews.text'])
    text = ""
    for rev in reviews:
        ps = sia.polarity_scores(rev)
        if opinion(ps, threshold) == "neg_count":
            text += rev
    if not text:
        text = "NA"

    wordcloud = WordCloud(max_font_size=100, max_words=100, background_color="white")

    # Display the generated image:
    plt.imshow(wordcloud, interpolation='bilinear')
    plt.title(row[9][:35] + "...")
    plt.axis("off")
```

Negative Review Wordcloud by Product



In [16]:

```
Y = 2000 # dummy leap year to allow input X-02-29 (leap day)
seasons = [('winter', (date(Y, 1, 1), date(Y, 3, 20))),
            ('spring', (date(Y, 3, 21), date(Y, 6, 20))),
            ('summer', (date(Y, 6, 21), date(Y, 9, 22))),
            ('autumn', (date(Y, 9, 23), date(Y, 12, 20))),
            ('winter', (date(Y, 12, 21), date(Y, 12, 31)))]
```



In [17]:

```

season_df = pd.DataFrame({})
season_dict = {}

for n, row in df.iterrows():
    if row['id'] not in season_dict:
        season_dict[row['id']] = {}
        season_dict[row['id']]['name'] = None
        season_dict[row['id']]['spring'] = 0
        season_dict[row['id']]['summer'] = 0
        season_dict[row['id']]['autumn'] = 0
        season_dict[row['id']]['winter'] = 0
        season_dict[row['id']]['lastestdate'] = None

    season_datetime = datetime.strptime(row['reviews.date'][:10], '%Y-%m-%d')
    season_string = get_season(season_datetime)
    season_dict[row['id']]['name'] = row['name']
    season_dict[row['id']]['season_string'] += 1
    if season_dict[row['id']]['lastestdate'] == None or season_datetime > season_dict[row['id']]['lastestdate']:
        season_dict[row['id']]['lastestdate'] = season_datetime

print(season_dict)

```

```

{'AVqVGZNVQmlgsOJE6eUY': {'name': 'Amazon Kindle E-Reader 6" Wifi (8th
Generation, 2016)', 'spring': 16, 'summer': 12, 'autumn': 25, 'winte
r': 43, 'lastestdate': datetime.datetime(2018, 5, 23, 0, 0)}, 'AWFUWc8
THh53nbDRF6YO': {'name': 'Amazon Echo Show Alexa-enabled Bluetooth Spe
aker with 7" Screen', 'spring': 37, 'summer': 75, 'autumn': 208, 'wint
er': 330, 'lastestdate': datetime.datetime(2018, 9, 15, 0, 0)}, 'AWK8z
OpOIwln0LfXlSxH': {'name': 'Amazon Echo Show Alexa-enabled Bluetooth S
peaker with 7" Screen', 'spring': 27, 'summer': 11, 'autumn': 51, 'win
ter': 106, 'lastestdate': datetime.datetime(2018, 5, 24, 0, 0)}, 'AV-X
eQLWuClrwyj_gbp5': {'name': 'Amazon Fire TV with 4K Ultra HD and Alexa
Voice Remote (Pendant Design) | Streaming Media Player', 'spring': 3,
'summer': 0, 'autumn': 1, 'winter': 0, 'lastestdate': datetime.datetim
e(2018, 5, 25, 0, 0)}, 'AWMjT0WguClrwyj_rFh3': {'name': 'Amazon - Echo
Plus w/ Built-In Hub - Silver', 'spring': 28, 'summer': 0, 'autumn': 2
16, 'winter': 346, 'lastestdate': datetime.datetime(2018, 4, 21, 0,
0)}, 'AVpfIfGAlcnluZ0-emy': {'name': 'Amazon 9W PowerFast Official OE
M USB Charger and Power Adapter for Fire Tablets and Kindle eReaders',
'spring': 4, 'summer': 10, 'autumn': 6, 'winter': 2, 'lastestdate': da
tetime.datetime(2017, 7, 7, 0, 0)}, 'AVph0EeEilAPnD_x9myq': {'name':
'Fire Kids Edition Tablet, 7 Display, Wi-Fi, 16 GB, Blue Kid-Proof Cas
e', 'spring': 105, 'summer': 30, 'autumn': 115, 'winter': 311, 'lastes
tdate': datetime.datetime(2017, 6, 2, 0, 0)}, 'AVqVGZSEQmlgsOJE6eUc':
{'name': 'Kindle E-reader - White, 6 Glare-Free Touchscreen Display, W
i-Fi - Includes Special Offers', 'spring': 22, 'summer': 26, 'autumn':
49, 'winter': 62, 'lastestdate': datetime.datetime(2018, 5, 17, 0,
0)}, 'AVpgdkC8ilAPnD_xsvyi': {'name': 'Fire Tablet, 7 Display, Wi-Fi,
16 GB - Includes Special Offers, Black', 'spring': 80, 'summer': 136,
'autumn': 53, 'winter': 102, 'lastestdate': datetime.datetime(2017, 8,
30, 0, 0)}, 'AVqVGWLKnnclJgDc3jF1': {'name': 'Fire Kids Edition Table
t, 7 Display, Wi-Fi, 16 GB, Green Kid-Proof Case', 'spring': 47, 'summ
er': 8, 'autumn': 41, 'winter': 121, 'lastestdate': datetime.datetime
(2017, 6, 2, 0, 0)}, 'AVqkIhkhv8e3D10-lebZ': {'name': 'All-New Fire HD
8 Tablet, 8 HD Display, Wi-Fi, 16 GB - Includes Special Offers, Blue',
'spring': 6, 'summer': 0, 'autumn': 14, 'winter': 31, 'lastestdate': d
atetime.datetime(2017, 4, 3, 0, 0)}, 'AVqkIh9HQmlgsOJE6fu_': {'name':
'All-New Fire HD 8 Tablet, 8 HD Display, Wi-Fi, 32 GB - Includes Speci
al Offers, Blue', 'spring': 16, 'summer': 0, 'autumn': 8, 'winter': 4

```



```

6, 'lastestdate': datetime.datetime(2017, 5, 26, 0, 0)}, 'AVpfpK8KLJeJ
ML43BCuD': {'name': 'Amazon Tap - Alexa-Enabled Portable Bluetooth Spe
aker', 'spring': 27, 'summer': 88, 'autumn': 48, 'winter': 62, 'lastes
tdate': datetime.datetime(2018, 4, 11, 0, 0)}, 'AVpidLjVilAPnD_xEVpI':
{'name': 'Fire HD 10 Tablet, 10.1 HD Display, Wi-Fi, 16 GB - Includes
Special Offers, Silver Aluminum', 'spring': 31, 'summer': 37, 'autum
n': 4, 'winter': 34, 'lastestdate': datetime.datetime(2017, 8, 26, 0,
0)}, 'AVqVGZO3nnclJgDc3jGK': {'name': 'Kindle Oasis E-reader with Leat
her Charging Cover - Merlot, 6 High-Resolution Display (300 ppi), Wi-Fi
- Includes Special Offers', 'spring': 12, 'summer': 3, 'autumn': 5,
'winter': 4, 'lastestdate': datetime.datetime(2017, 6, 30, 0, 0)}, 'AV
phPmHuilaPnD_x3E5h': {'name': 'Amazon - Kindle Voyage - 6" - 4GB - Bla
ck', 'spring': 16, 'summer': 26, 'autumn': 18, 'winter': 22, 'lastestd
ate': datetime.datetime(2018, 3, 21, 0, 0)}, 'AVqkIhwDv8e3D10-lebb':
{'name': 'All-New Fire HD 8 Tablet, 8" HD Display, Wi-Fi, 16 GB - Incl
udes Special Offers, Magenta', 'spring': 127, 'summer': 0, 'autumn': 1
19, 'winter': 551, 'lastestdate': datetime.datetime(2017, 6, 4, 0,
0)}, 'AVpjEN4jLJeJML43rpUe': {'name': 'Brand New Amazon Kindle Fire 16
gb 7" Ips Display Tablet Wifi 16 Gb Blue', 'spring': 103, 'summer': 14
1, 'autumn': 81, 'winter': 142, 'lastestdate': datetime.datetime(2017,
8, 4, 0, 0)}, 'AVpftoiJlcnuZ0-p5n2': {'name': 'Amazon - Kindle Voyage
- 4GB - Wi-Fi + 3G - Black', 'spring': 5, 'summer': 9, 'autumn': 1, 'w
inter': 7, 'lastestdate': datetime.datetime(2016, 5, 7, 0, 0)}, 'AVqkI
j9snnclJgDc3khU': {'name': 'Fire HD 8 Tablet with Alexa, 8" HD Displa
y, 32 GB, Tangerine - with Special Offers', 'spring': 9, 'summer': 0,
'autumn': 7, 'winter': 37, 'lastestdate': datetime.datetime(2017, 6,
3, 0, 0)}, 'AVqVGZN9QMLgsOJE6eUZ': {'name': 'Kindle Oasis E-reader wit
h Leather Charging Cover - Black, 6" High-Resolution Display (300 pp
i), Wi-Fi - Includes Special Offers', 'spring': 13, 'summer': 11, 'aut
umn': 5, 'winter': 10, 'lastestdate': datetime.datetime(2017, 7, 22,
0, 0)}, 'AVqkIh8WQMLgsOJE6fu-': {'name': 'All-New Fire HD 8 Tablet, 8"
HD Display, Wi-Fi, 32 GB - Includes Special Offers, Black', 'spring':
9, 'summer': 0, 'autumn': 6, 'winter': 43, 'lastestdate': datetime.dat
etime(2017, 6, 2, 0, 0)}, 'AVqkIiKWnnclJgDc3khH': {'name': 'All-New Fi
re HD 8 Tablet, 8" HD Display, Wi-Fi, 32 GB - Includes Special Offers,
Magenta', 'spring': 5, 'summer': 0, 'autumn': 5, 'winter': 30, 'lastes
tdate': datetime.datetime(2017, 5, 7, 0, 0)}, 'AVqkIdZiv8e3D10-leaJ':
{'name': 'Fire Tablet with Alexa, 7" Display, 16 GB, Magenta - with Sp
ecial Offers', 'spring': 19, 'summer': 29, 'autumn': 18, 'winter': 35,
'lastestdate': datetime.datetime(2017, 5, 22, 0, 0)}}

```

In [18]:

```
season_df = pd.DataFrame(season_dict.values())
season_df["id"] = season_dict.keys()
season_df
```

Out[18]:

	name	spring	summer	autumn	winter	lastestdate	id
0	Amazon Kindle E-Reader 6" Wifi (8th Generation...	16	12	25	43	2018-05-23	AVqVGZNvQMlgsOJE6eUY
1	Amazon Echo Show Alexa-enabled Bluetooth Speak...	37	75	208	330	2018-09-15	AWFUWc8THh53nbDRF6YO
2	Amazon Echo Show Alexa-enabled Bluetooth Speak...	27	11	51	106	2018-05-24	AWK8z0pOlwln0LfXISxH
3	Amazon Fire TV with 4K Ultra HD and Alexa Voic...	3	0	1	0	2018-05-25	AV-XeQLWuC1rwyj_gbP5
4	Amazon - Echo Plus w/ Built-In Hub - Silver	28	0	216	346	2018-04-21	AWMjT0WguC1rwyj_rFh3
5	Amazon 9W PowerFast Official OEM USB Charger a...	4	10	6	2	2017-07-07	AVpflfGA1cnluZ0-emyp
6	Fire Kids Edition Tablet, 7 Display, Wi-Fi, 16...	105	30	115	311	2017-06-02	AVph0EeEilAPnD_x9myq
7	Kindle E-reader - White, 6 Glare-Free Touchscr...	22	26	49	62	2018-05-17	AVqVGZSEQMlgsOJE6eUc
8	Fire Tablet, 7 Display, Wi-Fi, 16 GB - Include...	80	136	53	102	2017-08-30	AVpgdkC8ilAPnD_xsvyi
9	Fire Kids Edition Tablet, 7 Display, Wi-Fi, 16...	47	8	41	121	2017-06-02	AVqVGWLKnnnc1JgDc3jF1
10	All-New Fire HD 8 Tablet, 8 HD Display, Wi-Fi,...	6	0	14	31	2017-04-03	AVqklhkhv8e3D1O-lebZ
11	All-New Fire HD 8 Tablet, 8 HD Display, Wi-Fi,...	16	0	8	46	2017-05-26	AVqklh9HQMlgsOJE6fu_
12	Amazon Tap - Alexa-Enabled Portable Bluetooth ...	27	88	48	62	2018-04-11	AVpfpK8KLJeJML43BCuD
13	Fire HD 10 Tablet, 10.1 HD Display, Wi-Fi, 16 ...	31	37	4	34	2017-08-26	AVpidLjVilAPnD_xEVpl
14	Kindle Oasis E-reader with Leather Charging Co...	12	3	5	4	2017-06-30	AVqVGZO3nnnc1JgDc3jGK

	name	spring	summer	autumn	winter	lastestdate	id
15	Amazon - Kindle Voyage - 6" - 4GB - Black	16	26	18	22	2018-03-21	AVphPmHuiAPnD_x3E5h
16	All-New Fire HD 8 Tablet, 8" HD Display, Wi-Fi...	127	0	119	551	2017-06-04	AVqklhwDv8e3D1O-lebb
17	Brand New Amazon Kindle Fire 16gb 7" Ips Displ...	103	141	81	142	2017-08-04	AVpjEN4jLJeJML43rpUe
18	Amazon - Kindle Voyage - 4GB - Wi-Fi + 3G - Black	5	9	1	7	2016-05-07	AVpftoj1cnluZ0-p5n2
19	Fire HD 8 Tablet with Alexa, 8" HD Display, 32...	9	0	7	37	2017-06-03	AVqklj9snnc1JgDc3khU
20	Kindle Oasis E-reader with Leather Charging Co...	13	11	5	10	2017-07-22	AVqVGZN9QMlgsOJE6eUZ
21	All-New Fire HD 8 Tablet, 8" HD Display, Wi-Fi...	9	0	6	43	2017-06-02	AVqklh8WQMlgsOJE6fu-
22	All-New Fire HD 8 Tablet, 8" HD Display, Wi-Fi...	5	0	5	30	2017-05-07	AVqkliKWnnc1JgDc3khH
23	Fire Tablet with Alexa, 7" Display, 16 GB, Mag...	19	29	18	35	2017-05-22	AVqkldZiv8e3D1O-leaJ

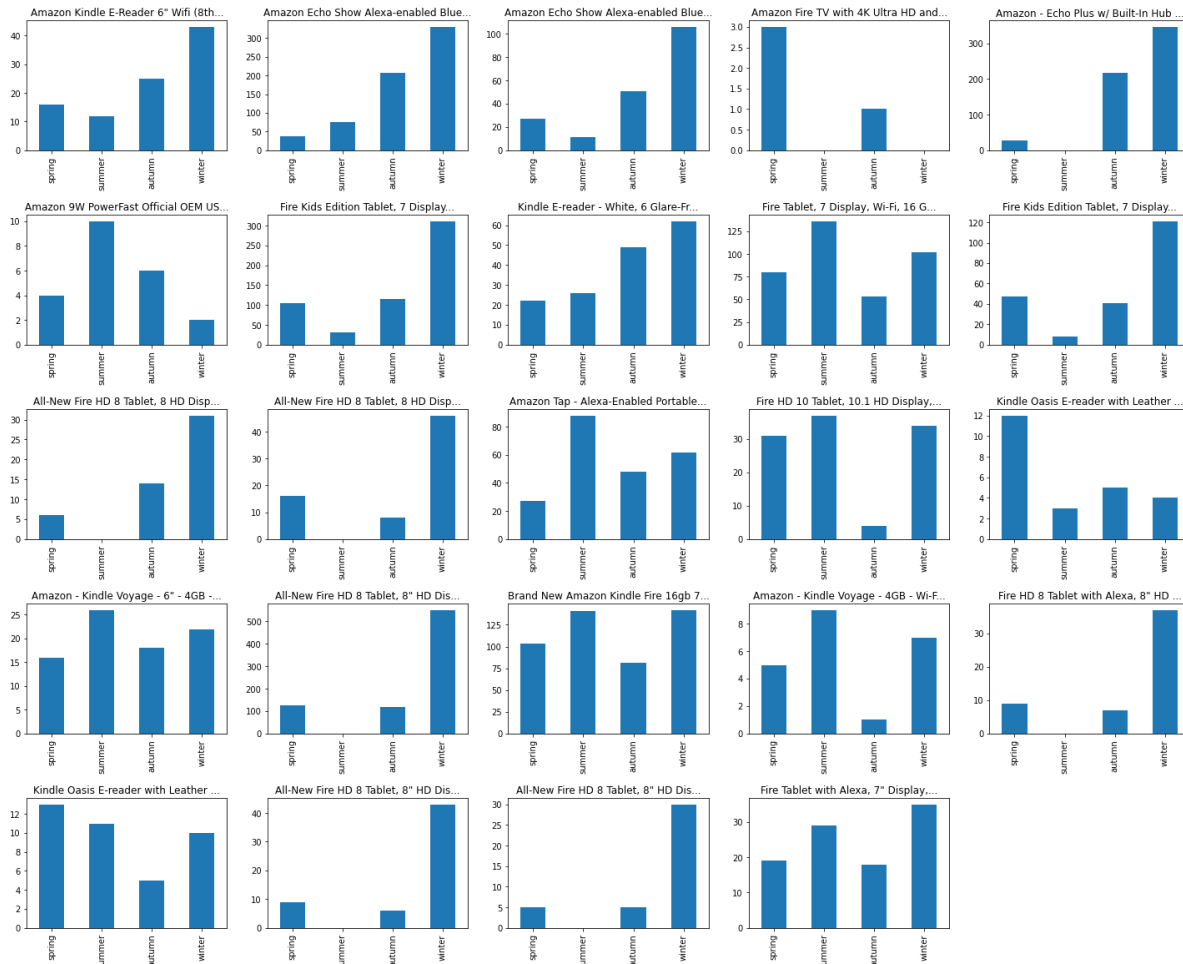
In [19]:

```
plt.figure(figsize=(25, 20))
plt.subplots_adjust(hspace=0.5)
plt.suptitle("Seasons by Product", fontsize=18, y=0.95)

for n, row in season_df.iterrows():
    ax = plt.subplot(5, 5, n + 1)
    row[["spring", "summer", "autumn", "winter"]].plot.bar(ax=ax)

# chart formatting
ax.set_title(row[0][:35] + "...")
```

Seasons by Product

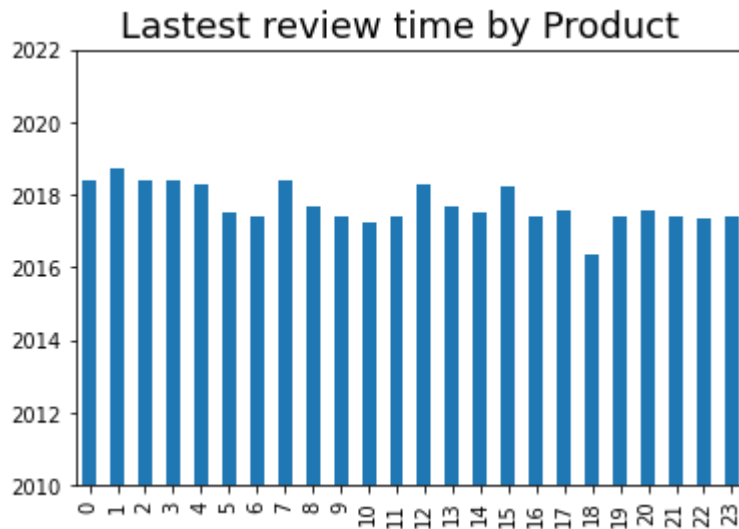


In [20]:

```
plt.figure()
plt.subplots_adjust(hspace=0.5)
plt.suptitle("Lastest review time by Product", fontsize=18, y=0.95)
plt.ylim((date(2010, 1, 1), date(2022, 1, 1)))
season_df['lastestdate'].plot.bar()
```

Out[20]:

<AxesSubplot:>



In [21]:

```
# Calculate the positive products in terms of normalized value.
myfilter = (mydf2['pos_norm'] > mydf2['neg_norm']) & (mydf2['pos_norm'] > mydf2['neu
c = len(mydf2[myfilter])

print("How many postive products:", c)
```

How many postive products: 24

In [22]:

# Show products name

```
for n, row in mydf2.iterrows():
    print("{0}: {1}".format(n, row['name']))
```

```
0: Amazon Kindle E-Reader 6" Wifi (8th Generation, 2016)
1: Amazon Echo Show Alexa-enabled Bluetooth Speaker with 7" Screen
2: Amazon Echo Show Alexa-enabled Bluetooth Speaker with 7" Screen
3: Amazon Fire TV with 4K Ultra HD and Alexa Voice Remote (Pendant Design) | Streaming Media Player
4: Amazon - Echo Plus w/ Built-In Hub - Silver
5: Amazon 9W PowerFast Official OEM USB Charger and Power Adapter for Fire Tablets and Kindle eReaders
6: Fire Kids Edition Tablet, 7 Display, Wi-Fi, 16 GB, Blue Kid-Proof Case
7: Kindle E-reader - White, 6 Glare-Free Touchscreen Display, Wi-Fi - Includes Special Offers
8: Fire Tablet, 7 Display, Wi-Fi, 16 GB - Includes Special Offers, Black
9: Fire Kids Edition Tablet, 7 Display, Wi-Fi, 16 GB, Green Kid-Proof Case
10: All-New Fire HD 8 Tablet, 8 HD Display, Wi-Fi, 16 GB - Includes Special Offers, Blue
11: All-New Fire HD 8 Tablet, 8 HD Display, Wi-Fi, 32 GB - Includes Special Offers, Blue
12: Amazon Tap - Alexa-Enabled Portable Bluetooth Speaker
13: Fire HD 10 Tablet, 10.1 HD Display, Wi-Fi, 16 GB - Includes Special Offers, Silver Aluminum
14: Kindle Oasis E-reader with Leather Charging Cover - Merlot, 6 High-Resolution Display (300 ppi), Wi-Fi - Includes Special Offers
15: Amazon - Kindle Voyage - 6" - 4GB - Black
16: All-New Fire HD 8 Tablet, 8" HD Display, Wi-Fi, 16 GB - Includes Special Offers, Magenta
17: Brand New Amazon Kindle Fire 16gb 7" Ips Display Tablet Wifi 16 Gb Blue
18: Amazon - Kindle Voyage - 4GB - Wi-Fi + 3G - Black
19: Fire HD 8 Tablet with Alexa, 8" HD Display, 32 GB, Tangerine - with Special Offers
20: Kindle Oasis E-reader with Leather Charging Cover - Black, 6" High-Resolution Display (300 ppi), Wi-Fi - Includes Special Offers
21: All-New Fire HD 8 Tablet, 8" HD Display, Wi-Fi, 32 GB - Includes Special Offers, Black
22: All-New Fire HD 8 Tablet, 8" HD Display, Wi-Fi, 32 GB - Includes Special Offers, Magenta
23: Fire Tablet with Alexa, 7" Display, 16 GB, Magenta - with Special Offers
```

## Delivery 4: Natural Language Processing - Sentiment Analysis (Marcell Veiner, Balin Lin)

Amazon Product Review : Business problem Perform a sentiment Analysis for Amazon Reviews using NLTK and other required Python packages and solve the following business problem for Amazon:

Which products should be kept?

- Products with high review counts and not terrible scores. For example, they are positive or neutral in term of NLP sentiment analysis.
  - Product 0, 1, 2, 4, 6, 7, 8, 9, 12, 13, 16, 17, 23

Which products should be dropped?

- The ones we have not kept. Furthermore, it's been a long time nobody buy the products.
- Products with a lot of bad reviews.
- Products with few reviews.
  - Product 3, 5, 14, 18, 22

Which products are junk?

- Products with low review counts and crap scores. For example, they are negative in term of NLP sentiment analysis.
  - Product 5, 18, 22

Which product should be recommended to customer?

- Products with high review scores and have at least some review counts.
  - Product 1, 4, 6, 8, 16, 17

Which consumer products are the best products?

- The ones with good reviews and a bunch of review counts.
  - Product 16

Which products should be planned for inventory for coming winter?

- The ones that sold a lot during autumn and winter.
  - Product 0, 1, 2, 4, 6, 7, 9, 10, 11, 16, 19, 21, 22

Which products require advertismment?

- Products with a few reviews but positive ones.
- Brand new products.
  - Product 3, 10, 11, 14, 15, 19, 20, 21

In list of opinion O how many quintuples have positive sentiment s?

- It is 24. We count the positive products with respect to the normalized value. The normalized positive value should be greater than the normalized negative and neutral values.