TEXT CLASSIFICATION ON PUBLIC UNSTRUCTURED HEALTHCARE DATA

Francisco Salas

Springboard 2018

TABLE OF CONTENTS

INTRODUCTION

Health informatics is a field of health data management that focuses on the collection, storage, distribution, and use of health data. The use of health data falls into two categories, primary and secondary use. Primary is when the health data collected is used to deliver health care to the individual from whom it was collected and secondary when the health data is used for clinical research, quality assurance, research & development and among other fields.

Health data is described as the epidemiology information related to health conditions, reproductive outcomes, causes of death or any data related to the quality of life, they are classified as either structured or unstructured.

Structured health data is standardized and easily transferable between health information systems. It includes quantitative data like patient demographics, medication list, patient vitals, family health history, lab results.

Unstructured health data unlike structured it does not follow a particular format and therefore not standardized. It includes physician notes about a patient, emails, patient surveys among others.

Understanding and reconciling the two major types of health data is a challenging problem in the health informatics field. The way structure data is stored (rows & tables) makes it easier to analyze and store because of its straightforward boundaries, while unstructured data with its many formats is a little more difficult. Some benefits of unstructured health data are that it has a chance to optimize personal patient care if experts can find a way to decode it.

IBM is one company with its Watson for Patient Record Analytics (aka Watson EMRA) that is at the forefront using Natural Language Processing (NLP) and machine learning to provide intelligent insights from the patient record for patient care.

Natural Language Processing or (NLP) is an area of AI that deals with the interactions between computers and natural human languages.

Because of the regulations of HIPPA, most health data is private, but there is a vast amount of unstructured public health data in the form of online healthcare reviews.

Crowd-sourcing specialized sites like RateMds, ZocDoc, Vitals or third party sited like Facebook or Yelp have a vast amount of data in the form of online reviews.

A question arises with this kind of data, can public unregulated healthcare data improve patient care?

I believe that health care providers can improve patient care by analyzing unstructured public healthcare data like online reviews by understanding patients' needs in their own words using text classification to extract meaning.

DATA ACQUISITION

Powerful crowd-sourced sites like Yelp provide a partition of their data for students freely to conduct research or analysis, and that is what it will be used.

The Yelp Dataset Challenge[1] it is a subset of businesses reviews, it contains about 188,593 business with around 5.9 million reviews.

YELP DATASET

Two files were used

- `yelp_academic_dataset_business.json`
  - Contains business data including location data, attributes, and categories.
- `yelp_academic_dataset_review.json`
  - Contains full review text data including the `user_id` that wrote the review and the `business_id` the review is written for.

[1] https://www.yelp.com/dataset/

DATA CLEANING

Yelp business reviews are divided into 22 top categories[2] with multiple subcategories, since this project will be focusing on just healthcare reviews, we will drop any non-healthcare and medical reviews from our dataset.

DATASETS

yelp_academic_dataset_business.json		
Variable Name	Description	Used
business_id	character unique string business id	Y
name	string, the business's name	Y
neighborhood	string, the neighborhood's name	N
address	string, the full address of the business	N
city	string, the city	Y
state	string, two character state code, if applicable	Y
postal_code	string, the postal code	N
latitude	float, latitude	N
longitude	float, longitude	N
stars	float, star rating, rounded to half-stars	N
review_count	integer, number of reviews	Y
is_open	integer, 0 or 1 for closed or open, respectively	N
attributes	object, business attributes to values	N
categories	an array of strings of business categories	Y
hours	an object of a key day to value hours, hours are using a 24hr clock	N

yelp_academic_dataset_review.json		
Variable name	Description	Used
review_id	string, 22 character unique review id	Y
user_id	string, 22 character unique user id, maps to the user in user.json	Y
business_id	string, 22 character business id, maps to business in business.json	Y
stars	integer, star rating	Y
date	string, date formatted YYYY-MM-DD	Y
text	string, the review itself	Y
useful	integer, number of useful votes received	Y
funny	integer, number of funny votes received	Y
cool	integer, number of cool votes received	Y

CLEANING STEPS

Starting first with the business dataset loaded in padas and selecting seven useful columns and dropping the rest.

- business_id
- categories
- city
- name
- review_count
- star_avg
- state

Since this project will be focusing on only US Healthcare reviews, any business outside of the US was dropped by filtering on city and state columns and dropping any null values with 2 or more null rows.

[2] https://www.yelpblog.com/2018/01/yelp_category_list

To select only healthcare related businesses, the categories column was expanded and filter by only choosing categories that fell under the 'Health and Medical[3] and dropping any business that did not fall under that filter.

With this new healthcare business dataset groping by categories and counting total reviews, a new categorical column was created by selecting the top healthcare subcategories and labeling each business according to its top category creating nine categories. Most businesses fell under these top categories

- chiropractors
- hospitals
- family practices
- obstetrician
- diagnostic service
- urgent care
- physical therapy
- mental health

Finally, the newly clean business dataset was merged with the review dataset by using pandas `merge` on `business_id`, a column that both datasets.

```
health = pd.merge(df_business,df_review, on='business_id')
```

Data Cleaning Results

- Original dataset
  - Total business: 188,593
  - Total Reviews: 5,996,996
- Healthcare only dataset
  - Total Business: 3,062
  - Total Reviews: 44,918

[3] https://www.yelpblog.com/2018/01/yelp_category_list#section9

TEXT PREPROCESSING

Before any analysis is made, some text processing needs to be done. In our new dataset, the column `text` contains a full review by a given user to a specific business; it can contain a maximum of 5,000 characters according to yelp user agreement.

CUSTOM FUNCTIONS

If a user review is describing a medical professional under a specific name like NP for Nurse Practitioner or PA for Physician Assistant, all 44 thousand reviews would need to expand medical title acronyms for the healthcare professional. A custom python dictionary with healthcare professional acronym title as key and the expanded term as the value was created and looped over every review and replace the given text; the same treatment was made for contractions.

Example

```
MEDICAL_MAP = {
'GYN':'Gynecologist',
'RN': 'Registered Nurse',
# more ...
CONTRACTION_MAP = {
"ain't": "is not",
"aren't": "are not",
"can't": "cannot",
"'cause": "because",
"could've": "could have",
# more ...
```

```
def replace_contraction_medical(text):
    '''replaces medical titles and
    contractions and expands them'''
    for word in text.split():
        if word  in MEDICAL_MAP:
            text = text.replace(word, MEDICAL_MAP[word])
        if word  in CONTRACTION_MAP:
            text = text.replace(word,  CONTRACTION_MAP[word])
    return text
```

With the help of textacy a python library based on Spacy with text was lowercase, removed any URLs, odd punctuations, and number references were removed.

```
df['processed'] = df['text'].map(lambda x: textacy.preprocess.preprocess_text(x,
                                                                             lowercase=True,
                                                                             no_urls=True,
                                                                             no_punct=True,
                                                                             no_numbers=True))
```

Results

Sample original text

```
"Memorial Day Weekend..  I can't Thank Dr, Shucmacher, his head nurse and staff for saving my Life...  I had an
 allergic reaction and they immediately went into action when I arrived and I can't Thank them enough... It jus
t so happened (with follow up) that I have a growth on my tongue that swells when I have an allergic reaction c
ausing me to not be able  to swallow and difficulty speaking and breathing...  The whole staff was amazing, car
ing and truly interested in what they could do to help me...  They were an AMAZING STAFF and I can't thank them
 enough....THANK YOU From THE Bottom of my HEART!  Holly Hernandez/Pahrump NV"
```

Cleanup version

```
"memorial day weekend i cannot thank doctor shucmacher his head nurse and staff for saving my life i had an
allergic reaction and they immediately went into action when i arrived and i cannot thank them enough it just
so happened with follow up that i have a growth on my tongue that swells when i have an allergic reaction
causing me to not be able to swallow and difficulty speaking and breathing the whole staff was amazing caring
and truly interested in what they could do to help me they were an amazing staff and i cannot thank them enough
thank you from the bottom of my heart holly hernandez pahrump nv"
```

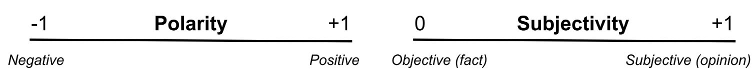
SENTIMENT ANALYSIS

Sentiment analysis refers to the use of natural language processing to extract and analyze subjective information. It is a process of determining the opinion or feelings of a piece of text like a tweet or a movie review. Companies find it useful for gaining insights into customers opinions and how they feel and identify of things that they like and dislike.

TextBlob is a python library for processing textual data it is built on top of nltk; it provides a rules base sentiment score along with additional functionality.

The sentiment property of textblob returns a tuple with two values, polarity & subjectivity. Polarity is a float value from negative one to positive one and describes the sentiment of the text with values closer to negative are consider "bad" while values closer to positive are consider "good."

Subjectivity is a float value with a range of zero to a positive one and describes how opinionated the text is with values closer to zero are objective (fact) while values closer to one are subjective (opinion).



The way textblob sentiment calculations work is by using a large lexicon of English adjectives with polarity and subjectivity values created by Tom De Smedt and Walter Daelmans. [4]

It is a rules-based approach, it attains the value from the lexicon and multiplies it bases on the previous word and returns the average value.

Example if we use the word `"great"` textblob polarity value will be `0.8`, wich is a positive statement but if we use the phrase `"not great,"` the polarity value will be `-0.4`, given that 0.8 is multiplied by `-.05.`

Polarity and subjectivity values were taken for all 44,918 reviews creating two new columns in the dataset with its respective names.

```
polarity = lambda x: TextBlob(x).sentiment.polarity
subjectivity = lambda x: TextBlob(x).sentiment.subjectivity
# create new cols
df['polarity'] = df['processed'].apply(polarity)
df['subjectivity'] = df['processed'].apply(subjectivity)
```

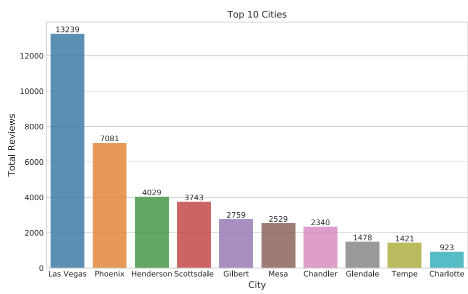
4 https://github.com/sloria/TextBlob/blob/eb08c120d364e908646731d60b4e4c6c1712ff63/textblob/en/en-sentiment.xml

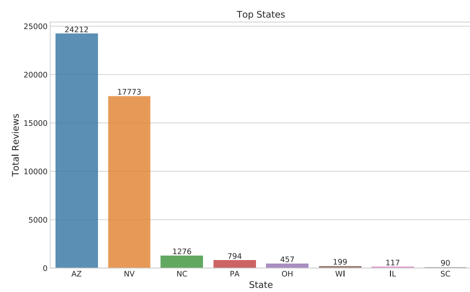
EDA

City and State

There are about 190 unique cities with Las Vegas coming in at number one with 13,239 reviews followed by Phoenix with almost half the values in at 7,081.

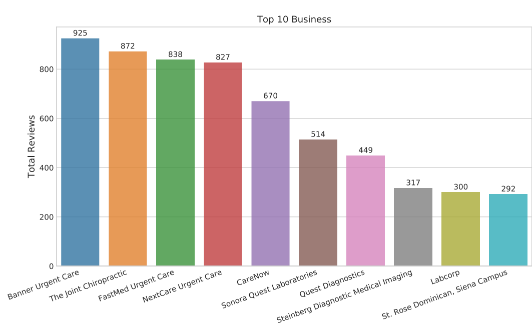
For states, Arizona comes in at number 1 with 24,212 reviews and Nevada following close with 17,773, and it drops off quickly with North Carolina with only 1,276.

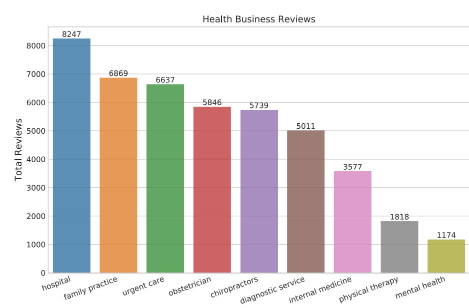




Business

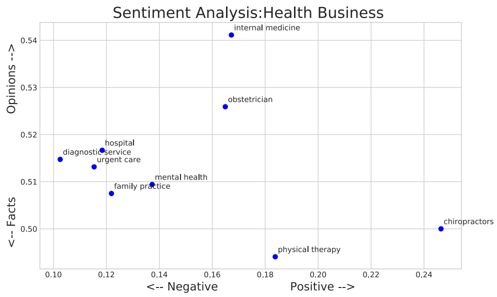
For the top ten businesses, we have four out of the top five total reviews are for urgent care facilities. For Business type, we have hospital with 8,2487 reviews followed by family practice with 6,869.

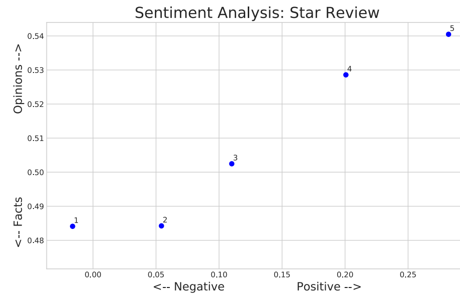




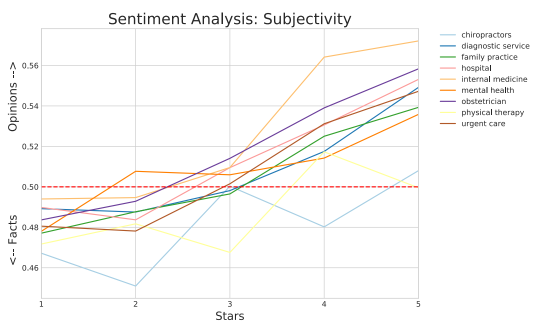
Sentiment Analysis

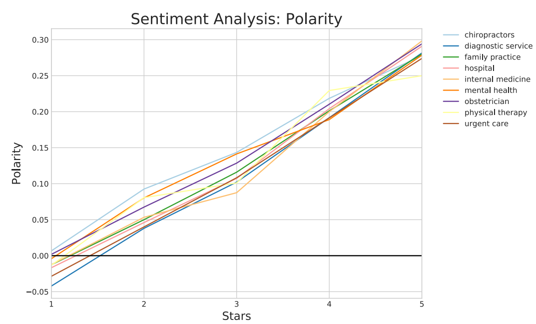
Sentiment values were plotted with polarity on the x-axis and subjectivity on the y-axis. The sentiment values for the type of health business are pretty interesting. Chiropractor reviews are the most positive while diagnostic reviews are mostly negative. Internal medicine reviews are the most subjective while physical therapy is the most factual. For sentiment value given star review seems pretty linear with 1 to 2-star reviews having a negative polarity with a low subjectivity while 3-5-star reviews have a positive polarity with a high subjectivity.





Going in deeper with star review by health business we can see that 1-2 star reviews are more objective by the values falling under 0.50 versus 3 to 5-star reviews except for chiropractors and physical therapy who mostly stay below 0.50. For polarity values the fall into a nice trend with only 1-star reviews falling under negative polarity values.



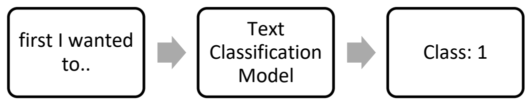


TEXT CLASSIFICATION

The goal of text classification is to automatically classify the text document into one or more defined categories. In our data we will try to predict star review value and what the type of healthcare business. For this project, Naïve Bayes and Support Vector Machines will be used.

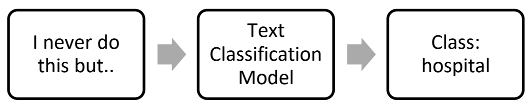
For star review value, a new column was created by combining one and two-star reviews called `bad_review` and 5 star reviews called `good_review to balance each class.`

```
data = df.ix[np.where((df.stars<=2)|(df.stars==5))]
data['review'] = np.where((data.stars<=2),'bad_review','good_review')
```



For healthcare business we will try to classify the text document from 9 different categories:

- `hospital`
- `family practice`
- `urgent care`
- `obstetrician`
- `chiropractors`
- `diagnostic service`
- `internal medicine`
- `physical therapy`
- `mental health`



NAÏVE BAYES

Naïve Bayes classifier is a good way to start since its very simple to implement; it lets you identify form a text source whether this label is more likely than that label, It's called naïve because it ignores word order and looks at the word frequency.

SUPPORT VECTOR MACHINE

In SVM each data item as a point in *n-feature* space with the value of each feature being the value of a particular coordinate. We perform classification by finding the hyper-plane that differentiates the two classes. What we are trying to do is maximize our margin, that is the distance between hyper-plane and the nearest points.

VECTORIZATION

Vectorization is the process of converting a collection of documents into a numerical feature vector aka (bag of words). For each algorithm to work, text data will be vectorized.

Results

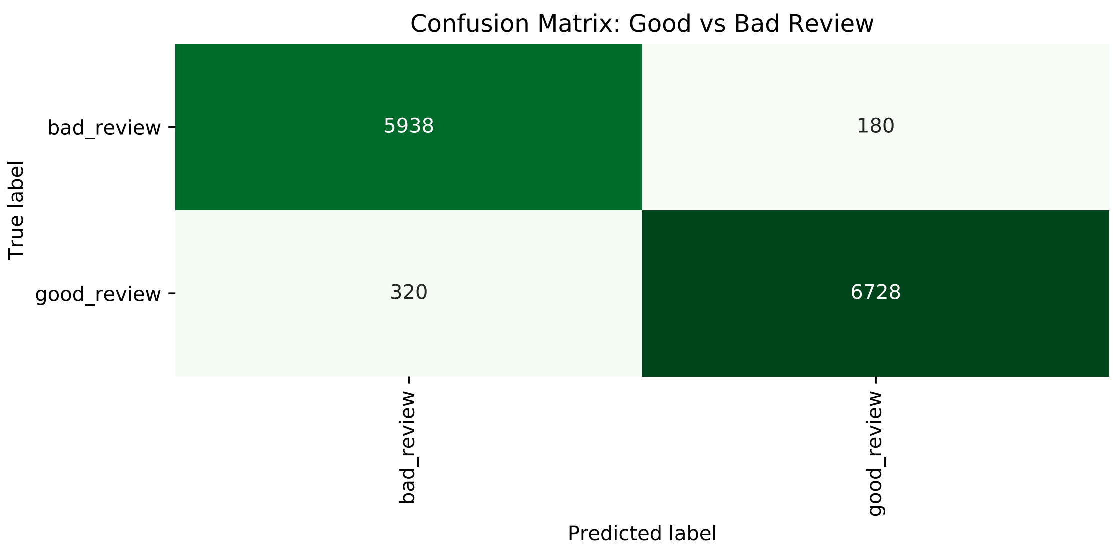
Multinomial Naïve Bayes: Star review

- Accuracy: 0.962
- Parameters
  - ngram_range=(1,2)
  - max_features = 500000
  - max_df = 0.5
  - alpha = 0.5

Classification report

	precision	recall	f1-score	support
bad_review	0.95	0.97	0.96	6118
good_review	0.97	0.95	0.96	7048
micro avg	0.96	0.96	0.96	13166
macro avg	0.96	0.96	0.96	13166

Confusion Matrix



Multinomial Naïve Bayes: Healthcare Business

- Accuracy: 0.597
- F1 score: 0.60
- Parameters
  - ngram_range=(1,2)
  - stopwords='english'
  - max_features=10000
  - max_df=0.6
  - alpha=1

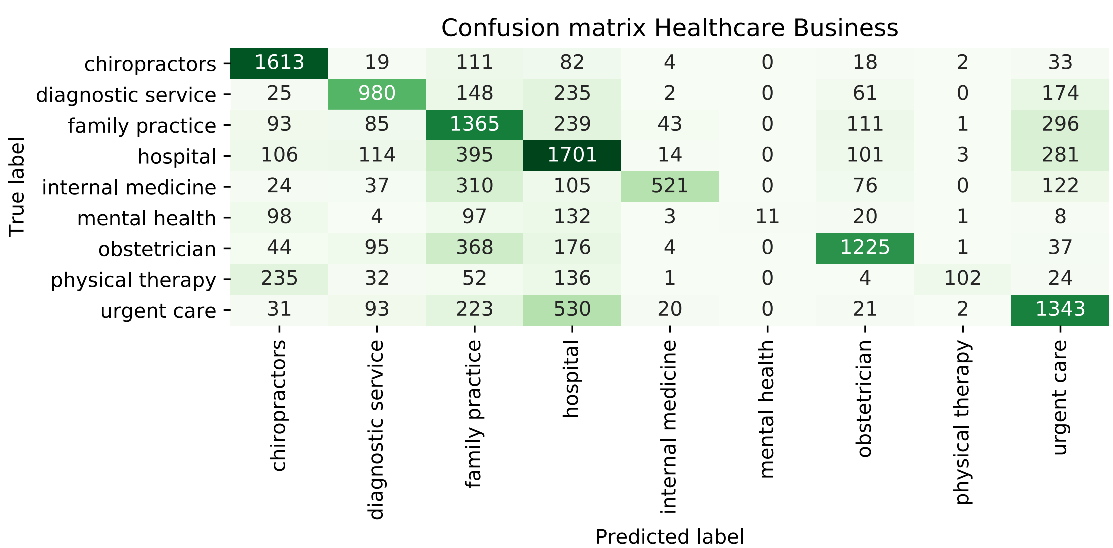
Classification Report

```
                     precision    recall  f1-score   support

       chiropractors       0.71      0.86      0.77      1882
 diagnostic service       0.67      0.61      0.64      1625
    family practice       0.44      0.61      0.51      2233
           hospital       0.51      0.63      0.56      2715
  internal medicine       0.85      0.43      0.57      1195
      mental health       1.00      0.03      0.06       374
       obstetrician       0.75      0.63      0.69      1950
   physical therapy       0.91      0.17      0.29       586
        urgent care       0.58      0.59      0.58      2263

          micro avg       0.60      0.60      0.60     14823
          macro avg       0.71      0.51      0.52     14823
       weighted avg       0.64      0.60      0.59     14823
```

Confusion matrix



Linear Support Vector Classification: Star Review

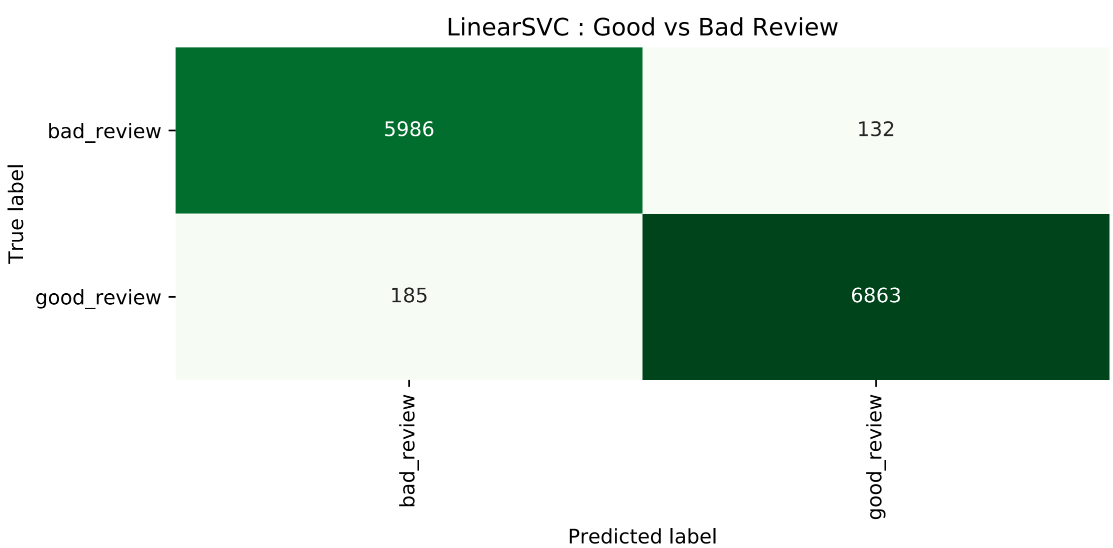
- Accuracy: 0.976
- Parameters
  - ngram_range=(1,2)
  - max_df = 0.5

- c=5

Classification Report

	precision	recall	f1-score	support
bad_review	0.97	0.98	0.97	6118
good_review	0.98	0.97	0.98	7048
micro avg	0.98	0.98	0.98	13166
macro avg	0.98	0.98	0.98	13166
weighted avg	0.98	0.98	0.98	13166

confusion matrix



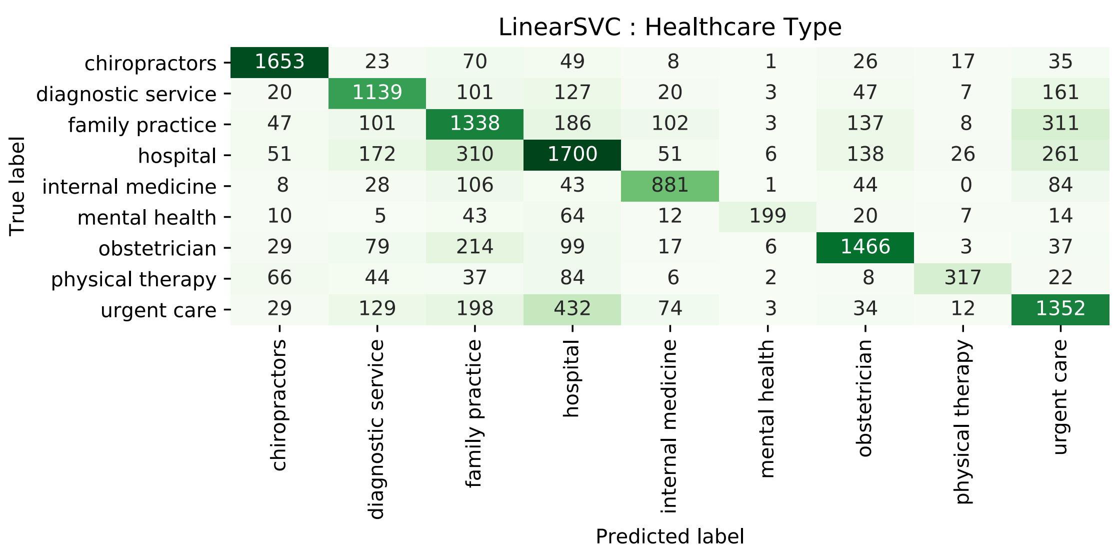
LINEAR SUPPORT VECTOR CLASSIFICATION: HEALTHCARE BUSINESS

- Accuracy: 0.678
- F1 Score : 0.68
- Parameters
  - ngram_range=(1,2)
  - max_df=0.55
  - stop_words_'english'
  - C=2

Classification Report

	precision	recall	f1-score	support
chiropractors	0.86	0.88	0.87	1882
diagnostic service	0.66	0.70	0.68	1625
family practice	0.55	0.60	0.58	2233
hospital	0.61	0.63	0.62	2715
internal medicine	0.75	0.74	0.74	1195
mental health	0.89	0.53	0.67	374
obstetrician	0.76	0.75	0.76	1950
physical therapy	0.80	0.54	0.64	586
urgent care	0.59	0.60	0.60	2263
micro avg	0.68	0.68	0.68	14823
macro avg	0.72	0.66	0.68	14823

Confusion Matrix



CONCLUSION

TEXT CLASSIFICATION: STAR RATING

Multinomial Bayes text classification condition on star rating generated an accuracy of 96% returning only 180 false positives out of 6118 negative reviews and 320 false negatives out of 7048 positive reviews.

Comparing it to Linear Support Vector Classification having an accuracy of 97% returning only 132 false positives out of 6118 for negative reviews and 185 false negatives out of 7048 positive reviews

TEXT CLASSIFICATION: HEALTH BUSINESS

Because of how unbalanced the data is, the F1 score was taken into consideration where the F1 score is the “harmonic mean” of recall and precision. Multinomial Bayes text classification condition on healthcare business generated a total overall F1 score of 0.60 with chiropractors having an F1 value of 0.77 while mental health returns an F1 value of 0.06, the lowest out of all the classes. That means we Incorrectly rejected 363 out of 374 times.

Comparing it to Linear Support Vector Classification having an overall F1 score of 0.68 with chiropractors having the best score with an F1 value of 0.87 with all classes returning F1 values higher than 0.50 with family practice returning the lowest F1 value of 0.58.

Given the results for both text classifications, it clearly shows that SVC outperforms Bayes.

Future improvements can be made by implementing other supervised classification algorithms like logistic regression or applying unsupervised classification like topic modeling to the data