Term Frequency Analysis of Drug Review Dataset BDPP

Karl-Johan Djervbrant

May 2020

1 Introduction

As a part of the course *Big Data Parallel Programming* at *Halmstad University* the student is sought to either use data mining or machine learning techniques on a large data set with the use of *Apache Spark*. This project will focus on data mining of the *Drug Review Dataset* shared by Gräßer *et al.*[1]. The dataset consists of over 200000 reviews from patient prescribed a drug for a given medical condition, later on, the patient gave a review of the experience and along with a rating between zero and ten. The theory is to analyze which terms are most commonly associated with a drug prescribed for a medical condition, with this information one might draw conclusions on what side-effects or results are most frequent.

In section 2 the different development suites and libraries will be presented, along with information about the dataset. This is followed by a documentation about the approach and how the different libraries where used in section 3. Thereafter the results are presented in section 4 and are discussed in section 5, along with potential areas of further research and improvements. High resolution of the images in the Figures can be found in the Appendix.

2 Background

Before analyzing the data the environments used had do be setup. Python 3.8.2, openJDK-13 and Apache Spark 3.0.0 preview2 with Pyspark was used. The reason Spark 3.0.0 preview2 was chosen is because Spark 2.4.5 doesn't work with Python 3.8+. The development suit used was Jupyter lab since the notebook layout of Jupyter simplifies analysis and speeds up development.

The dataset is acquired from the *UCI Machine Learning Repository* and consists of 263041 reviews (215063 according to UCI but this is not correct) of different drugs prescribed to patients. When Gräßer *et al.* created this dataset they collected the data from two sites, 215063 reviews where from *Drugs.com* and

3551 where from *Druglib.com*. The data is stored in a tab separated file (`.tsv`) split into two files, 75% in the train file and 25% in the test file. These two files are joined since no machine learning techniques will be used.

3 Method

To begin, some basic research was made to understand the dataset, such as number of distinct drugs and conditions and missing data. As seen in Table 1 there's quite a lot of missing data and since personal reviews are hard (if not impossible) to interpolate, the rows where data is missing are dropped. However, the data in the columns `id, date` and `usefulCount` won't be analyzed, therefor missing data in these columns are allowed. After reduction there were a total of 189918 reviews left.

	id	drugName	condition	review	rating	date	usefulCount
Before	2	23896	25112	23922	72027	72034	72034
After	1	0	0	0	0	3	3

Table 1: Number of missing data in each column of the dataset, before and after reduction.

Since the goal of the project is to analyze which terms are most frequently used when reviewing a drug prescribed for a given condition, the dataset is grouped by `drugName` and `condition` where each review in the group is concatenated to one string and each rating are appended to an array, as seen in Table 2 and 3 where a subset of the dataset is presented. For each "drug per condition" the mean and standard deviation calculated, along with the number of reviews.

drugName	condition	count
Etonogestrel	Birth Control	3937
Ethinyl estradiol / norethindrone	Birth Control	2787
Nexplanon	Birth Control	2573
Levonorgestrel	Birth Control	2521
Ethinyl estradiol / levonorgestrel	Birth Control	1921
Ethinyl estradiol / norgestimate	Birth Control	1905
Levonorgestrel	Emergency Contraception	1499
Phentermine	Weight Loss	1488
Implanon	Birth Control	1351
Miconazole	Vaginal Yeast Infection	1201

Table 2: Subset of number of prescriptions of a drug to treat a given condition.

To begin, research had to be done on how to do pre-processing in Natural

drugName	condition	review	rating	num_reviews	avg	std
Absorica	Acne	I'm a 23..	[1.0, 4.0 ..	4	6.0	4.24

Table 3: An example of how an entry in the DataFrame looks like after the first reduction.

Language Processing, where the book Introduction to Information Retrieval written by Christoffer D. Manning *et al.* [2] was a useful read. The following steps are based on information in this book. The first step before analysis of the reviews can begin were to remove unwanted characters. As seen in Figure 1 there's a lot of unwanted characters which would disturb the final result. Thereafter, each review is tokenized [3], which creates unigrams of the review. The `RegexTokenize()` came in handy at this step since the user can select what to split on with a Regex function. Another feature is that it's possible to select if the tokenized words should be kept in their original form or transformed to lowercase. After every review has been tokenized, every stop word is removed since these doesn't give any valuable information about how the patient reacted to the drug. This was easily done with the pyspark function `StopWordsRemover()` from `pyspark.ml.feature`. Next step were to use lemmatization [4] which transform a word to its base form, e.g.

words → *word*
boats, boat's, boats' → *boat*
months, things, happened → *month, thing, happen*

"""I’m a 23 year old female who has had acne issues since about 12 years old. Topicals kept me okay through my teen years until I got put on an antibiotic called Solodyn when I began getting adult acne at 22. It worked great but is not a permanent fix unless you want to take that pill every day for the rest of your life, so I tried Absorica. 5 months was all it took to clear up my severe nodular acne and dry out the blackheads I've had for 10 years. I experienced no serious side effects, just dry skin/lips and nosebleeds. With Aquaphor and Vaseline on hand, it was easily manageable and just a minor annoyance. Completely worth the results. I highly recommend Absorica, so long as you've done your research and monitor your health on it."""

Figure 1: An example of a review before it's been processed to remove unwanted characters.

To lemmatize each word the `WordNetLemmatizer()` function form the `NLTK`-library [5] was used, together with the `pos_tag()` function from the same library. The POS-tagger (Part-of-Speech) tags each word with the corresponding word class, i.e. Noun, Verb, Adverb or Adjective.

took Tenormin years ago → *('took', 'verb'), ('Tenormin', 'noun'), ('years', 'noun'), ('ago', 'adverb')*

Next step after lemmatization were to create the Bi-grams, using the `NGram()` function in `pyspark.ml.feature`. The reason bi-grams was used instead of unigrams were that two terms can give more context, especially when describing a side effect of a drug as *('lose weight')* and *('bad acne')* is much more informative than *('lose'), ('weight'), ('bad'), ('acne')*. These bi-grams are now counted and the term frequency for a drug-condition pair is presented as a list where the first element is the bi-gram and the second is the frequency of how many times this term appeared in the reviews, e.g. `[['bad pain', 18], ['side effect', 12], ['work great', 6]]`.

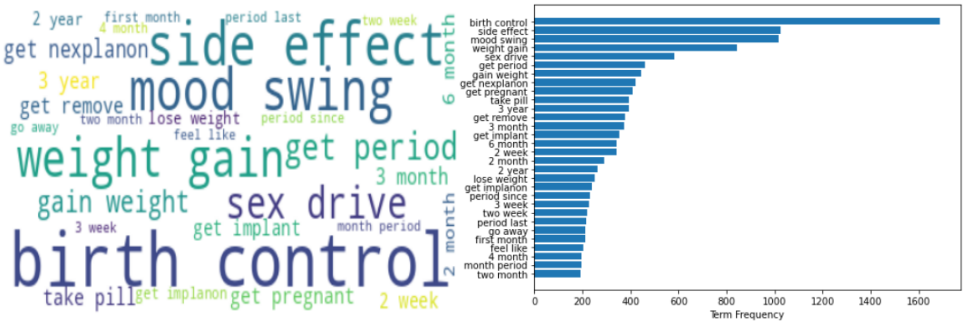
To extract further information about the reviews and bi-grams, sentiment analysis is used to get how positive or negative a term or sentence is.
The `nltk.sentiment.vader` [6] has a pre-trained model, `SentimentIntensityAnalyzer`, which is used to extract the compound sentiment score, were a negative term i.e. *bad pain* scores close to -1 which is the lower bound, and *love this* is a positive term and scores close to 1 which is the upper bound.

To present the results, a library named `wordcloud` is used which as the name suggests, creates a word cloud where N-ammounts of words can be shown in a quite unique and interesting way. The term frequency of the bi-grams are proportional to the font size, which results in an image where the most frequent bi-grams dominates the image. The other plots where created with `matplotlib` and are discussed more in Section 4.

4 Results

The dataset consists of 263041 reviews which is first reduced to 189918 reviews after the missing data is removed. After data is grouped by `drugName` and `condition`, there exists 9110 drugName-condition pairs.

The results from the analysis is presented in 4 figures per drug-condition pair. The word cloud and the bar-plot shows the 30 most frequent bi-grams and the two other figures lists the top 30 results sorted on sentiment score. As seen in Figure 2a where a word cloud of the reviews for the drug *Etonogestrel* is prescribed for the condition *Birth Control*, one can see a tendency that there might be some side effects. Along with the bar-plot in Figure 2b one can draw a conclusion that there's many which complains about side effects, these side effects might be mood swing and/or weight gain.



(a) 2a (b) 2b

Figure 2: Term frequency of 3937 reviews for Etonogestrel prescribed for Birth Control, mean score: 5.8, std: 3.33, mean sentiment: -0.01, std: 0.72. Higher resolution available in appendix.

To continue the analysis, the following tables in Figure 3 shows the term frequency and sentiment score for the bi-grams. Figure 3a is sorted on sentiment score in ascending order, this returns a list with the bi-grams which are the most negative. Figure 3b is also sorted on sentiment score, but in descending order which results in a list with the most positive bi-grams. From this one can study both the negative and positive bi-grams to widen the perspective about how the drug is performing.

Bi-gram	Term Frequency	Sentiment score
panic attack	36	-0.7506
depression anxiety	68	-0.6597
anxiety depression	48	-0.6597
bad cramp	55	-0.6486
cramp bad	21	-0.6486
absolutely hate	21	-0.6115
really bad	90	-0.5849
swing depression	49	-0.5719
depression mood	27	-0.5719
negative side	43	-0.5719
negative review	36	-0.5719
period bad	28	-0.5423
bad side	63	-0.5423
bad acne	41	-0.5423
swing horrible	22	-0.5423
bad get	30	-0.5423
bad review	41	-0.5423
make bad	22	-0.5423
bad birth	37	-0.5423
bad experience	62	-0.5423
bad part	47	-0.5423
bad thing	49	-0.5423
gotten bad	27	-0.5423
bad period	25	-0.5423
bad mood	57	-0.5423
horrible mood	36	-0.5423
much bad	24	-0.5423
bad headache	32	-0.5423
get bad	85	-0.5423
insertion hurt	24	-0.5267

(a) 3a

Bi-gram	Term Frequency	Sentiment score
good luck	22	0.7096
absolutely love	75	0.6697
best decision	23	0.6369
love period	31	0.6369
definitely recommend	63	0.6369
love first	33	0.6369
love birth	73	0.6369
love nexplanon	60	0.6369
love implanon	59	0.6369
best birth	50	0.6369
best thing	24	0.6369
love get	35	0.6369
far love	24	0.6369
year love	44	0.6369
period love	22	0.6369
month love	39	0.6369
period great	21	0.6249
great get	28	0.6249
great experience	29	0.6249
great birth	34	0.6249
great period	34	0.6249
work great	62	0.6249
year great	24	0.6249
great first	24	0.6249
think great	35	0.6249
month great	30	0.6249
super heavy	21	0.5994
super light	22	0.5994
month gain	42	0.5267
since gain	23	0.5267

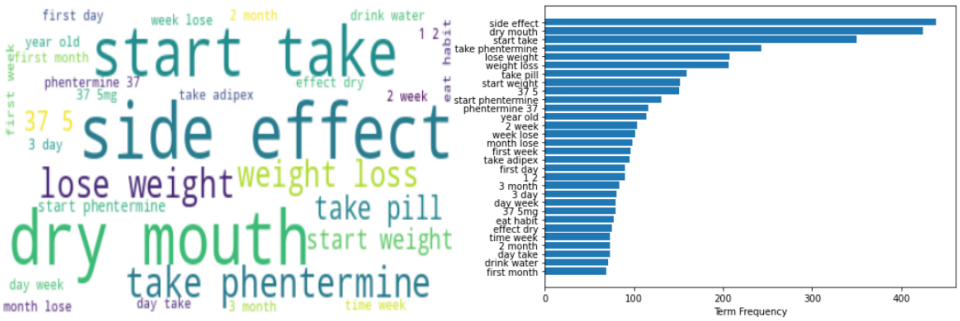
(b) 3b

Bi-gram	Term Frequency	Sentiment score
birth control	1689	0.0000
side effect	1026	0.0000
mood swing	1016	0.0000
weight gain	845	0.5267
sex drive	584	0.0000
get period	459	0.0000
gain weight	443	0.5267
get nexplanon	423	0.0000
get pregnant	410	0.0000
take pill	395	0.0000
3 year	393	0.0000
get remove	378	0.0000
3 month	374	0.0000
get implant	352	0.0000
6 month	342	0.0000
2 week	342	0.0000
2 month	289	0.0000
2 year	263	0.0000
lose weight	251	-0.4019
get implanon	240	0.0000
period since	232	0.0000
3 week	227	0.0000
two week	218	0.0000
period last	214	0.0000
go away	213	0.0000
first month	212	0.0000
feel like	204	0.3612
4 month	197	0.0000
month period	196	0.0000
two month	191	0.0000

(c) 3c

Figure 3: Tables showing Term Frequency and Sentiment score for 30 Bi-grams generated from 3937 reviews for Etonogestrel prescribed for Birth Control. 2a sorted on Sentiment score in ascending and 2b in descending, 2c is sorted on Term Frequency descending. Higher resolution available in appendix.

From Figure 3b and 3c we can see that there's users who likes this medication, *absolutely love* occurs 75 times and *definitely recommend* occurs 63 times. But these are out numbered by *weight gain*, *side effect* and *mood swing* which occurs 1689, 1026 and 1016 times. If you also take into account that the mean score is 5.8 which is fairly mediocre, one can say that this medication might do it's job but it for sure has drawbacks.



(a) 4a (b) 4b

Figure 4: Term frequency of 1488 reviews for Phentermine prescribed for Weight Loss, mean score: 8.77, std: 2.0, mean sentiment: 0,26, std: 0.62. Higher resolution available in appendix.

Figure 4 shows the results for another drug, this time with a higher mean score and mean sentiment but with less reviews. It has a high term frequency for *side effect*, *dry mouth* and *weight loss* which is also presented in Figure 5.

Bi-gram	Term Frequency	Sentiment score
lose 5	35	-0.4019
lose 7	25	-0.4019
need lose	21	-0.4019
goal lose	21	-0.4019
trouble sleep	24	-0.4019
lose 6	25	-0.4019
lose 10	55	-0.4019
far lose	22	-0.4019
lose 4	26	-0.4019
lose total	51	-0.4019
lose another	23	-0.4019
already lose	48	-0.4019
weight lose	24	-0.4019
day lose	44	-0.4019
lose 9	23	-0.4019
lose 10lbs	23	-0.4019
lose 8	38	-0.4019
lose 30	37	-0.4019
lose 12	21	-0.4019
ago lose	40	-0.4019
lose 20	33	-0.4019
lb lose	26	-0.4019
lose 15	30	-0.4019
week lose	101	-0.4019
month lose	98	-0.4019
lose weight	207	-0.4019
want lose	35	-0.3400
loose weight	21	-0.3182
weight loss	206	-0.3182
blood pressure	46	-0.2960

(a) 5a

Bi-gram	Term Frequency	Sentiment score
good luck	59	0.7096
wish luck	22	0.6908
feel great	53	0.6249
work great	29	0.6249
weight gain	22	0.5267
gain back	43	0.5267
gain weight	68	0.5267
luck everyone	23	0.4588
highly recommend	25	0.4201
eat healthy	63	0.4019
recommend anyone	26	0.3612
felt like	29	0.3612
feel like	62	0.3612
make sure	34	0.3182
work well	25	0.2732
much energy	28	0.2732
give energy	37	0.2732
lot energy	42	0.2732
want get	27	0.0772
reach goal	30	0.0258
prescribe phentermine	24	0.0000
weigh today	25	0.0000
start work	25	0.0000
every morning	24	0.0000
lifestyle change	25	0.0000
4 pound	25	0.0000
pound start	25	0.0000
4 week	24	0.0000
2 year	25	0.0000
weight come	25	0.0000

(b) 5b

Bi-gram	Term Frequency	Sentiment score
side effect	439	0.0000
dry mouth	424	0.0000
start take	350	0.0000
take phentermine	243	0.0000
lose weight	207	-0.4019
weight loss	206	-0.3182
take pill	159	0.0000
start weight	152	0.0000
37 5	151	0.0000
start phentermine	131	0.0000
phentermine 37	116	0.0000
year old	114	0.0000
2 week	103	0.0000
week lose	101	-0.4019
month lose	98	-0.4019
first week	96	0.0000
take adipex	95	0.0000
first day	90	0.0000
1 2	90	0.0000
3 month	84	0.0000
3 day	80	0.0000
day week	79	0.0000
37 5mg	79	0.0000
eat habit	77	0.0000
effect dry	75	0.0000
2 month	73	0.0000
day take	73	0.0000
time week	73	0.0000
drink water	71	0.0000
first month	69	0.0000

(c) 5c

Figure 5: Tables showing Term Frequency and Sentiment score for 30 Bi-grams generated from 1488 reviews for Phentermine prescribed for Weight Loss. 2a sorted on Sentiment score in ascending and 2b in descending, 2c is sorted on Term Frequency descending. Higher resolution available in appendix.

5 Conclusion

The word cloud and bar-plot gives a great overview about the information in the reviews, one can quick and easy get a sens about the information even though you have no prior knowledge. The tables in Figure 3 can be used in a deeper analysis to answer more underlying questions, such as *which are the most common bi-grams with a negative and positive meaning?* and *Are there many bi-grams with negative meaning, if not, how does it compare to the mean score of the reviews?*.

But there's still much more room for improvements on this implementation. If we take a closer look at Figure 3 we can see that many bi-grams occur two ore more times. E.g. *weight gain* and *gain weight* is essentially the same words but are shown as two separate. One other thing which one might consider evaluating is changing every number to the same word, it would result in that *3 month, 2 month* and *4 month* would be reduced to *NUMBER month*, since these entries might not give much information either way about the review. This would have an big impact on the results in Figure 5a since there's many entries with *lose* and a number.

Other areas of improvement is the analysis of review rating. According to Hu *et al.*[7] the mean metric of reviews is not meaningful when working with reviews which are asymmetric bimodal distributed, in other words, when the curve is J-Shaped. Since people tend to comment/review a product when they are either satisfied or unsatisfied with it, we get an under-reporting bias in the middle range.

When it comes to improvement with the code, there's might be room for improvements. It takes long time to do the last reduction when computing the term frequency for every bi-gram when it's done with `groupBy`, there might be other ways of performing this which are much faster. One other thing might be consider using `Spark-NLP` developed by John Snow Labs [8] instead of regular Spark. It has every algorithm which has been imported from `NLTK` already implemented and many pre-trained models.
Overall, this works well and gives an overview about how the patients react to the medication without the need of reading every review, which saves a lot of time from the researcher.

References

[1] Gräßer F, Kallumadi S, Malberg H, Zaunseder S. Aspect-Based Sentiment Analysis of Drug Reviews Applying Cross-Domain and Cross-Data Learning. In: Proceedings of the 2018 International Conference on Digital Health. DH '18. New York, NY, USA: Association for Computing Machinery; 2018. p. 121–125. Available from: https://doi.org/10.1145/3194658.3194677.

[2] Christopher D Manning PR, Schütze H. An Introduction to Information Retrieval. Cambridge University Press; 2009. Available from: https://www.informationretrieval.org/.

[3] Christopher D Manning PR, Schütze H. An Introduction to Information Retrieval. Cambridge University Press; 2009. p. 22–27. Available from: https://www.informationretrieval.org/.

[4] Christopher D Manning PR, Schütze H. An Introduction to Information Retrieval. Cambridge University Press; 2009. p. 32–34. Available from: https://www.informationretrieval.org/.

[5] Bird EL Steven, Klein E. Natural Language Processing with Python. O'Reilly Media Inc.; 2009. Available from: https://www.nltk.org/.

[6] Bonaccorso G. Machine Learning Algorithms: Popular algorithms for data science and machine learning. Packt Publishing Ltd; 2018.

[7] Hu N, Pavlou P, Zhang J. Overcoming the J-Shaped Distribution of Product Reviews. Communications of the ACM. 2009 10;52:144–147. Available from: https://doi.org/10.1145/1562764.1562800.

[8] Labs JS. Spark-NLP;. Accessed on: 14-05-2020. Available from: https://nlp.johnsnowlabs.com/.

A Appendix

Figure 6: Etonogestrel prescribed for Birth Control

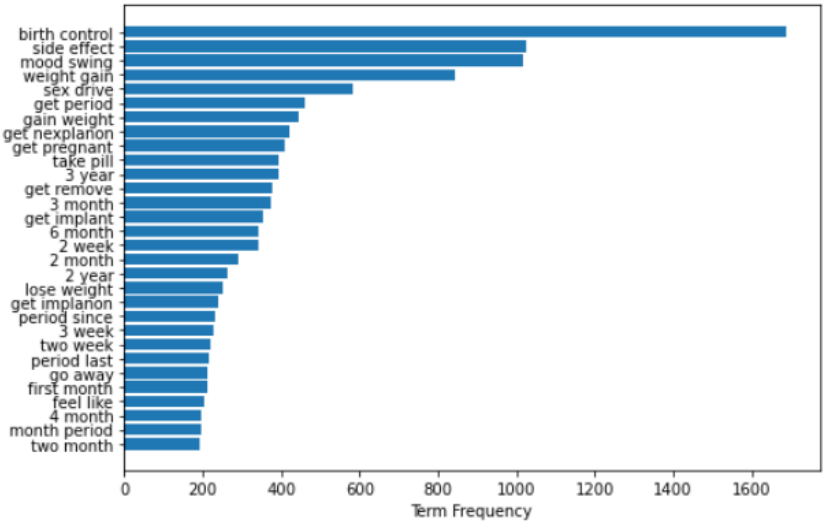


Figure 7: Etonogestrel prescribed for Birth Control

Bi-gram	Term Frequency	Sentiment score
birth control	1689	0.0000
side effect	1026	0.0000
mood swing	1016	0.0000
weight gain	845	0.5267
sex drive	584	0.0000
get period	459	0.0000
gain weight	443	0.5267
get nexplanon	423	0.0000
get pregnant	410	0.0000
take pill	395	0.0000
3 year	393	0.0000
get remove	378	0.0000
3 month	374	0.0000
get implant	352	0.0000
6 month	342	0.0000
2 week	342	0.0000
2 month	289	0.0000
2 year	263	0.0000
lose weight	251	-0.4019
get implanon	240	0.0000
period since	232	0.0000
3 week	227	0.0000
two week	218	0.0000
period last	214	0.0000
go away	213	0.0000
first month	212	0.0000
feel like	204	0.3612
4 month	197	0.0000
month period	196	0.0000
two month	191	0.0000

Figure 8: Etonogestrel prescribed for Birth Control

Bi-gram	Term Frequency	Sentiment score
good luck	22	0.7096
absolutely love	75	0.6697
best decision	23	0.6369
love period	31	0.6369
definitely recommend	63	0.6369
love first	33	0.6369
love birth	73	0.6369
love nexplanon	60	0.6369
love implanon	59	0.6369
best birth	50	0.6369
best thing	24	0.6369
love get	35	0.6369
far love	24	0.6369
year love	44	0.6369
period love	22	0.6369
month love	39	0.6369
period great	21	0.6249
great get	28	0.6249
great experience	29	0.6249
great birth	34	0.6249
great period	34	0.6249
work great	62	0.6249
year great	24	0.6249
great first	24	0.6249
think great	35	0.6249
month great	30	0.6249
super heavy	21	0.5994
super light	22	0.5994
month gain	42	0.5267
since gain	23	0.5267

Figure 9: Etonogestrel prescribed for Birth Control

Bi-gram	Term Frequency	Sentiment score
panic attack	36	-0.7506
depression anxiety	68	-0.6597
anxiety depression	48	-0.6597
bad cramp	55	-0.6486
cramp bad	21	-0.6486
absolutely hate	21	-0.6115
really bad	90	-0.5849
swing depression	49	-0.5719
depression mood	27	-0.5719
negative side	43	-0.5719
negative review	36	-0.5719
period bad	28	-0.5423
bad side	63	-0.5423
bad acne	41	-0.5423
swing horrible	22	-0.5423
bad get	30	-0.5423
bad review	41	-0.5423
make bad	22	-0.5423
bad birth	37	-0.5423
bad experience	62	-0.5423
bad part	47	-0.5423
bad thing	49	-0.5423
gotten bad	27	-0.5423
bad period	25	-0.5423
bad mood	57	-0.5423
horrible mood	36	-0.5423
much bad	24	-0.5423
bad headache	32	-0.5423
get bad	85	-0.5423
insertion hurt	24	-0.5267

Figure 10: Etonogestrel prescribed for Birth Control

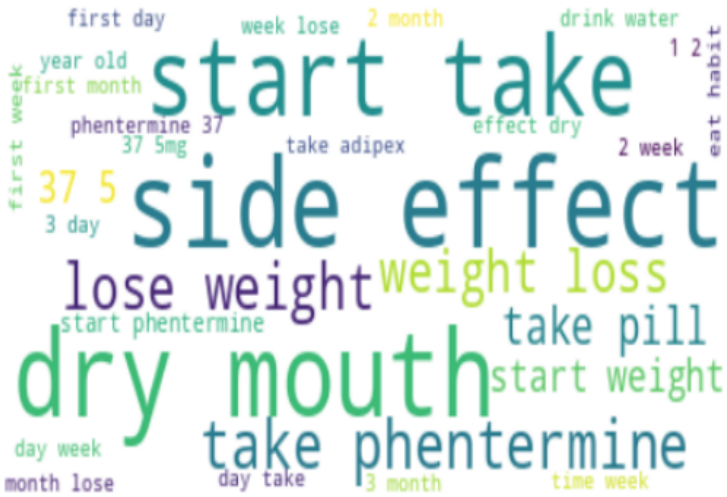


Figure 11: Phentermine prescribed for Weight Loss

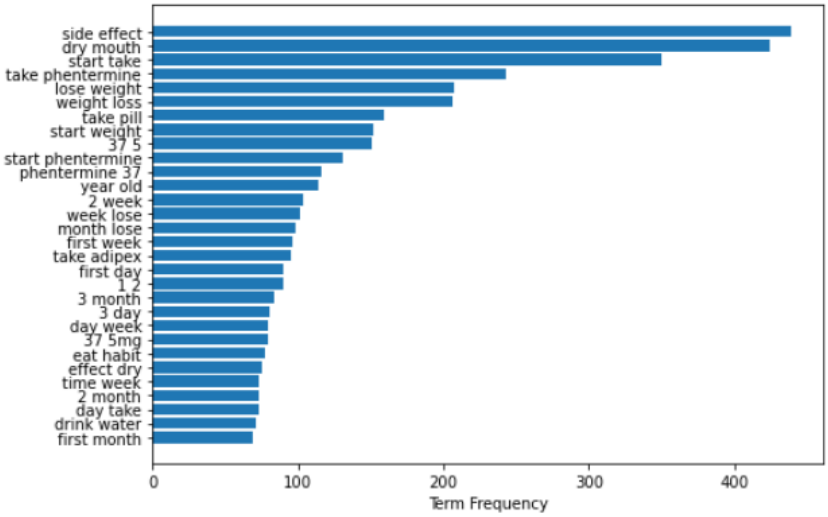


Figure 12: Phentermine prescribed for Weight Loss

Bi-gram	Term Frequency	Sentiment score
side effect	439	0.0000
dry mouth	424	0.0000
start take	350	0.0000
take phentermine	243	0.0000
lose weight	207	-0.4019
weight loss	206	-0.3182
take pill	159	0.0000
start weight	152	0.0000
37 5	151	0.0000
start phentermine	131	0.0000
phentermine 37	116	0.0000
year old	114	0.0000
2 week	103	0.0000
week lose	101	-0.4019
month lose	98	-0.4019
first week	96	0.0000
take adipex	95	0.0000
first day	90	0.0000
1 2	90	0.0000
3 month	84	0.0000
3 day	80	0.0000
day week	79	0.0000
37 5mg	79	0.0000
eat habit	77	0.0000
effect dry	75	0.0000
2 month	73	0.0000
day take	73	0.0000
time week	73	0.0000
drink water	71	0.0000
first month	69	0.0000

Figure 13: Phentermine prescribed for Weight Loss

Bi-gram	Term Frequency	Sentiment score
good luck	59	0.7096
wish luck	22	0.6908
feel great	53	0.6249
work great	29	0.6249
weight gain	22	0.5267
gain back	43	0.5267
gain weight	68	0.5267
luck everyone	23	0.4588
highly recommend	25	0.4201
eat healthy	63	0.4019
recommend anyone	26	0.3612
felt like	29	0.3612
feel like	62	0.3612
make sure	34	0.3182
work well	25	0.2732
much energy	28	0.2732
give energy	37	0.2732
lot energy	42	0.2732
want get	27	0.0772
reach goal	30	0.0258
prescribe phentermine	24	0.0000
weigh today	25	0.0000
start work	25	0.0000
every morning	24	0.0000
lifestyle change	25	0.0000
4 pound	25	0.0000
pound start	25	0.0000
4 week	24	0.0000
2 year	25	0.0000
weight come	25	0.0000

Figure 14: Phentermine prescribed for Weight Loss

Bi-gram	Term Frequency	Sentiment score
lose 5	35	-0.4019
lose 7	25	-0.4019
need lose	21	-0.4019
goal lose	21	-0.4019
trouble sleep	24	-0.4019
lose 6	25	-0.4019
lose 10	55	-0.4019
far lose	22	-0.4019
lose 4	26	-0.4019
lose total	51	-0.4019
lose another	23	-0.4019
already lose	48	-0.4019
weight lose	24	-0.4019
day lose	44	-0.4019
lose 9	23	-0.4019
lose 10lbs	23	-0.4019
lose 8	38	-0.4019
lose 30	37	-0.4019
lose 12	21	-0.4019
ago lose	40	-0.4019
lose 20	33	-0.4019
lb lose	26	-0.4019
lose 15	30	-0.4019
week lose	101	-0.4019
month lose	98	-0.4019
lose weight	207	-0.4019
want lose	35	-0.3400
loose weight	21	-0.3182
weight loss	206	-0.3182
blood pressure	46	-0.2960

Figure 15: Phentermine prescribed for Weight Loss