Creating Automatic Advertisements: A Sentiment Analysis and Summarization approach

Colton Paul
Student of Computer Science
New York University
cjp419@nyu.edu

Yikai Feng
Student of Computer Science
New York University
yf730@nyu.edu

Ziming Sheng
Student of Computer Science
New York University
zs720@nyu.edu

Abstract

Online advertising is becoming a larger and larger percentage of companies' advertising expenditure, encouraging development of better ways to create online ads and better ways to engage with online consumers. In this research, we explore the creation of automatic advertisements as a potential tool for companies to compete in rapidly changing markets. Using a large corpus of Amazon reviews, this task is carried out through the synthesis of semantic orientation, summarization, and additional filtering, ultimately generating new advertisements formed from user reviews. Specifically, the goal of our work is for these advertisements to be the best combination of sentences that both demonstrate enthusiasm for the product and nicely summarize its positive features.

1 Introduction

Internet advertising is a multibillion dollar industry where profits are directly related to a user's likelihood of clicking on an ad. This has spurred a great amount of research from corporations such as Facebook and Google to better target users with advertisements on their websites. A less analyzed problem, however, is how a retailer can more easily design the advertisements to achieve this goal. Our project looks into the possibility of automatic advertisement creation based on positive reviews of the product to accomplish this.

First, automatic advertisements would enable sellers with a great many products to advertise specific products without personally designing hundreds of individual ads. Secondly, automatic advertisements could have the huge advantage of incorporating known information about the individual into the advertisement creation process, thus doing a better job of catering their ads to specific consumers.

Unfortunately, there are some inherent difficulties in this task preventing a single simple approach. For example, reviews have a lot of excess information (e.g. anecdotes and background information). Additionally, even 5-star reviews have some negative comments about the product that would be entirely unsuitable for an advertisement, so the positivity and negativity of every statement must be taken into account. Lastly, can one bring these two metrics together to generate a coherent advertisement?

We take a summarization extraction approach to the task, using the reviewer's opinions themselves to generate an advertisement. Our process for tackling the aforementioned challenges for a given product is as follows:

1. Collect all positive reviews for the product and process them into collections of sentences.

2. Extract all opinion phrases from the sentences using manually determined part-of-speech patterns that tend to indicate opinion. These patterns are gathered experimentally and build off the work of Turney (Turney, 2001).

3. Determine the semantic orientation of each phrase using Pointwise Mutual Information between the phrase and a number of stars out of 5 (i.e. how strongly the phrase correlates with positive or negative reviews). Sentences are then scored as sums of their phrases' scores.

4. The TextRank algorithm (Mihalcea and Tarau, 2004) is implemented to rank all of the

product's sentences in order of summarization efficiency, with a score being assigned to each sentence.

5. A formula is experimentally determined to best combine the semantic orientation scores and summarization scores to select the top candidate sentences for extraction. Lastly, though out of the scope of this project, these candidate sentences would be such that they could be inserted into a generic advertising template for use as an advertisement.

As an outline, in section 2 we briefly introduce the data set, and in section 3 we explain our data processing and opinion extraction technique using this data. Section 4 details the semantic orientation step, and Section 5 covers the summarization step and how to combine the two into a single score. Lastly, sections 6 and 7 evaluate the process and provide concluding remarks on its efficacy.

2 Data Set

We used the Amazon Product Data corpus to carry out our work (Mcauley et al., 2015). We noticed the corpus was heavily skewed towards positive reviews, so we took a subset of these reviews to make a balanced background corpus for semantic orientation purposes. This included 3000 positive (5-star reviews) and 3000 negative (1 or 2-star reviews) from every one of 24 product categories in the original corpus.

3 Opinion Phrases Extraction

The opinion phrase extraction consists of the rules for opinion phrases and sentence reduction:

1. The rules for opinion phrases:
   Some manually generated rules, which function to label each input phrase as opinion phrase or non-opinion phrase.

2. Reduce sentences to opinion phrase sets:
   For each given sentence, decompose it into a set of opinion phrases that match those rules.

3.1 Pre-processing

For a given review, pre-processing is needed before applying the opinion phrases extraction function.

1. For each review, split it by period, question mark, exclamation mark or semicolon. After this step each review is decomposed into several sub-sentences.

2. Apply Part of Speech (POS) tagging to each sub-sentence before checking them using opinion rules.

There exists a sentence tokenizer from the NLTK python package, which can split paragraphs into sentences. However, using NLTK's sentence tokenizer cannot obtain the ideal result since online reviews are mostly written in casual language and typos are relatively more frequent compared to formal documents. In this case, it is more reliable and efficient to split it up using certain characters. In this program, periods, semicolons, question marks and exclamation marks (”.”,”?”,”!”,”;”) are set as end point of a sub-sentence because they can cut paragraphs into sub-sentences of descent length. For example, dividing sub-sentences by commas would give sentences too short to contain opinion phrases.

- However, this product, bought yesterday, is indeed a good one.
  [However, this product, bought yesterday, is indeed a good one]
  We can see in this example only one out of the four total sub-sentences contains an opinion phrase.

The next step is to apply POS tagging for each sub-sentence. NLTK's POS tagger is used to do the work.

3.2 Opinion Extraction Rules

The whole set of rules is primarily based on adjectives because by definition, adjectives are members of the class of words that modify nouns and pronouns. In most cases, the product is a specific object, which is also a noun. Therefore we can use adjectives in reviews as the customer's opinion for the product. Past work has verified this hypothesis. In particular, Turneys paper (Turney, 2001) introduced a set of rules based on POS tagged words to detect opinion phrases. In this project, we are also doing unsupervised learning so we brought in his idea and built our own rules based on his existing rules.

The first modification we did is to remove the rule: ”RB/RBR/RBS + VB/VBD/VBN/VBG + anything” from the rule set.

1. Back in 1995, Enya still had her creative spark, her own voice.
   Opinion extracted:

First Word	Second Word	Third Word(Not Extracted)
1. JJ	NN or NNS	anything
2. RB, RBR or RBS	JJ	not NN nor NNS
3. JJ	JJ	not NN nor NNS
4. NN or NNS	JJ	not NN nor NNS
5. RB, RBR or RBS	VB, VBD, VBN or VBG	anything

Table 1: Opinion Extraction Rules by Peter D. Turney.

   (a) ”still had” → remove
   (b) ”creative work”
   (c) ”own voice”

2. I’ve always had a soft spot for this song.
   Opinion extracted:
   (a) ”always had” → remove
   (b) ”soft spot”

The reason is that though we might gain opinion indicator verb phrases like still had and always had, they are actually meaningless without the following noun phrase or adjective phrase. However, we already have rules to deal with cases like noun phrases. Therefore we remove this rule in order to obtain a result that can better represent opinions of reviews.

The second modification we did is to add a new rule: ”(RB/RBR/RBS) + DT + NN/NNS”(adverbs are optional)

After reading through many reviews in our corpus, we noticed phrases containing positive words like masterpiece show up quite frequently. Therefore we need new rule to cover it. However, this is not enough. Introducing a new rule has the risk of overgenerating phrases in addition to the types of opinion phrases we wished to collect. So we further narrow the matched results down by using a positive word bank. (Hu and Liu, 2004) That is, for the new rule, we also check whether the noun in the phrase belongs to the positive vocabulary bank.

1. This is truly a masterpiece.
   [”truly a masterpiece”] → keep, since ”masterpiece” is in the positive word bank.

2. The companys second released product.
   [”the company”] → discard, ”company” not in positive word bank

Therefore the rules we set up for capturing opinion phrases are:

First Word	Second Word	Third Word(Not Extracted)
1. JJ	NN or NNS	anything
2. RB, RBR or RBS	JJ	not NN nor NNS
3. JJ	JJ	not NN nor NNS
4. NN or NNS	JJ	not NN nor NNS
5. DT	NN or NNS (in positive work bank)	anything
First Word	**Second Word**	**Third Word (Extracted)**
6. RB, RBR or RBS	DT	NN or NNS (in positive work bank)

Table 2: Modified Opinion Extraction Rules

Examples of opinion phrases extracted using the rules above from the corpus of digital music reviews are shown in Table 3.

3.3 Formatting Outputs

Given the ultimate goal of generating advertisements using summarization methods, the program filters out negative reviews to ensure the general positivity of the extracted opinion phrases. Using the above rules, we check through each sentence in every positive review for a given product and group the opinion phrases of a given sentence together.

4 Semantic Orientation

Our next step is to determine the semantic orientation of all the collected opinion phrases. Once again, we base our work off of Turney, with some modifications to benefit from our reviews being labeled with ratings (Turney, 2001).

First Word	Second Word	Third Word	Rules Satisfied
1. great	album	N/A	JJ NN
2. creative	spark	N/A	JJ NN
3. popularized	works	N/A	JJ NNS
4. truly	a	masterpiece	RB DT NN
5. very	good	N/A	RB JJ
6. more	enjoyable	N/A	RBR JJ
7. also	a	bit	RB DT NN
8. not	an	artist	RB DT NN
9. very	subtle	N/A	RB JJ
10. most	beautiful	N/A	RBS JJ
11. upbeat	tempo	N/A	JJ NN
12. ethereal	vocals	N/A	JJ NNS
13. angelic	voices	N/A	JJ NNS
14. wildly	popular	N/A	RB JJ
15. however	this	record	RB DT NN
16. albums	remastered	N/A	NNS JJ
17. years	old	N/A	NNS JJ

Table 3: Sample opinion phrases outputs using the rules above.

4.1 Scoring

First we similarly use Pointwise Mutual Information (PMI) as a measure of association defined as follows:

$$PMI(x, y) = log_2(\frac{p(x,y)}{p(x) \times p(y)})$$

(Church and Ullman, 1989). However, instead of calculating the PMI between our phrase and the words ”excellent” and ”poor” as Terney did, we calculate the PMI between our phrase and a given rating (e.g. wonderful device and being a 5-star review) This allows us to make the most use of our product corpus, which is large enough that most significant phrases will have several appearances within the corpus. In this way, the semantic orientation of a given phrase is determined by:

SO(phrase)=PMI(phrase, positive review) - PMI(phrase, negative review)

Phrase	Score
low end	-0.500535
foreign guitars	0
good value	1.68053
guitars now-a-days	0
few bucks	-0.744954
very good	0.6787406
different look	1.4063552
same electronics	-0.1786072
more expensive	0.17097719
too high	-0.7103277
fairly decent	-1.256609
quite pleased	2.5438587

Table 4: Same semantic scores outputs

(Turney, 2001). Table 4 displays the scores several phrases from a guitar review were assigned. Scores with zero values indicate that the phrase is used too seldom to have meaningful correlation with positivity or negativity. Considering the last step of our process is a summarization task, it is unlikely that sentences with these seldom used phrases could summarize the product, so lack of semantic information about these phrases is typically not an issue.

4.2 The Role of Negation in Semantic Orientation

One of the biggest difficulties in the field of sentiment analysis is the role of negating words in a given sentence. For instance, ”This product is absolutely great!” and ”This product is absolutely not great!” carry almost opposite meanings. To further complicate things, the structure of the negating sentence can vary greatly (e.g. ”Well, it is hardly the case that one could call this product absolutely great!”). For this reason we do not use Hu’s simple approach of reversing the semantic orientation if a negating word is within a certain distance from an adjective (Hu and Liu, 2004).Although there have been successful rule based methods to determine the scope of negation in a sentence, we determined that it is actually best to simplify the process by removing all sentences containing words indicating a negation, using the list generated by the rule-based approach of Councill et al. (Councill et al., 2010). This only removes a small fraction of review sentences and filters out many negative sentences. Further-more, the positive sentences with negation words tend not to be as emphatic as strong positive sentences, making the loss of these marginal.

5 Automatic Summarization

For automatic summarization, we view opinion sentences that we generated from the review set for one Amazon product as a single document. To summarize the top sentences that can better describe the product, we use the TextRank model (Mihalcea and Tarau, 2004) to rank the opinion sentences and output a score related to each sentence in the collection. Then we combined the effect of the semantic orientation score with the rank score to yield final score values, which ultimately determine which sentences are extracted.

5.1 Text to Graph

The PageRank algorithm (Page et al., 1999) used by Google is a graph-based ranking algorithm that counts the number of vertices and the weight of the edges to a certain vertex to output an estimate score of the vertex, which is the basis of importance measurement. To enable the PageRank algorithm for the ranking of opinion sentences, we need to construct a graph using the original set of text so that we can create a vertex for each sentence in the text. Instead of using the natural language of the sentences, we convert the text into a bag-of-words. This will return a matrix of lists, of which the row represents the sentences and the column represents all the words that have been tokenized. We regard each list as the vertex of the graph.

Here is a simple text documents contains two sentences:
(1) Hello world.
(2) Hello professor.
Based on these two sentences, a list of words is constructed as follows:
[“Hello”, “world”, “professor”]
Binary vector of term frequencies of words is constructed as follows:
$[1, 1, 0]$
$[1, 0, 1]$
1 represents the word is in the sentence, while 0 otherwise.

Instead of tokenizing sentences into single words, we also consider the bi-gram and tri-gram situation to see the correlation between words.

To enable the relations and interconnection between the vertices, we use undirected edges weighted by the similarity of sentences. For the similarity score, we calculated the TF-IDF score for words and transform them into similarities of sentences through transpose matrices. (Bohde, 2012). The key principle here is that the sentences that are most similar to all other sentences are probably the ones that capture the ideas of the text the best. Then we can have an undirected weighted graph.

5.2 Summarization by Combined Rank Score

For extractive summarization, we applied the PageRank algorithm to score the sentences based on the similarity score and to train the graph to make the score that assigned to each vertex converge in certain iterations. The simplified algorithm can be expressed by:

$$R(u) = \sum_{v \in B_u} \frac{R(v)}{N(v)} \tag{1}$$[1]

Since the PageRank algorithm was designed for directed graphs, and the edges between vertices of sentences in our graph are undirected, it will convert each undirected edge into two directed edges. After doing enough iterations before convergence, we will get a score for each vertex in the graph, that is, one TextRank score for each sentence in the text document.

Now for each sentence, there are two scores corresponding to it: one is given by summing up semantic orientation values of opinion phrases that show up in that sentence, and the other one is the correlated similarity value given by the TextRank algorithm.

We then use the following formula, which takes these two values as input and outputs the final value indicating the rank score. Let $S : s_1, s_2, \ldots, s_n$ be the set of sentences corresponding to a certain product. First sort the sentences to find the max SO score and max TextRank score. calculate average TextRank score(max TextRank score excluded) $\overline{TR}'$ and average SO score(max SO score excluded) $\overline{SO}'$ respectively. The sigmoid function (Wikipedia, 2016) is introduced to build the Combined Rank Score formula.

$$S(t) = \frac{1}{1 + e^{-t}}$$

The inputs: SO score and TextRank score are all non-negative values. To make outputs distribute

[1] u and v are two vertices in the graph, $R(u)$ and $R(v)$ are the ranking scores for the vertices. B_u is the set of vertices that point to u. N_u is the number of links from u.

more evenly, two ratio values $\frac{maxSO}{SO'}$ and $\frac{maxTR}{TR'}$ were used for rescaling. Therefore combining the two sigmoid functions together and plugging in inputs, we have the evaluation formula as:

$$Score(SO, TR) = \frac{c}{1 + (\frac{maxSO \cdot e}{SO'})^{-SO}} + \frac{1 - c}{1 + (\frac{maxTR \cdot e}{TR'})^{-TR}}$$

The value c is a arbitrary coefficient ($0 \leq c \leq 1$) that assigns weights between SO scores and TextRank scores. Final results can be improved by trying out different c values. Our final results used the value .6.

6 Evaluation

The training dataset we used is the review set for Amazon product: B0002CZUUG, an electric guitar provided by Epiphone. After sentence extraction, the training set contains 99 sentences.

At first, we use only TextRank algorithm without SO scores to rank the sentences, the resulting output is as follows:

Top five sentences:

1) I know that corners have to be cut to get to the price point of these budget guitars but all in all the Special ll is good to go right out of the box with just a little tweaking.

2) I realize that flatwound strings will require a new setup and intonation, but considering how good the guitar is and how relatively little it cost me, the new strings and work will be an investment.

3) I am very happy with this Epiphone Les Paul and will proudly play and OWN it.

4) I have other, way more expensive guitars, like a Gibson ES-335, Dean 12 string, and a Fender Strat, but I gotta tell you, I like this one soooo much I play it more than the others!

5) There are a lot of good value well made guitars nowa-days, I just found at this price vs quality ratio the Epiphone Special ll really is special.

Bottom five sentences:

95) I just purchased an electric/acoustic Epiphone ukelele today for my Chiropractor.

96) Had I gone with a more expensive model, I'd feel locked in.

97) Please note that most tuning problems are from a binding improperly cut nut.

98) I edited yet again as someone didn't appreciate my ”gloating” over other guitars I own.

99) It shipped the very next day and came VERY well protected in perfect condition.

However we found that the top sentences are not straightforward enough to tell the advantage of the product and the bottom sentences contain suitable sentences for advertising. Therefore, we decide to add the SO scores to the rank in order to emphasize the effect of positive phrases. The result is as follows:

Top five sentences:

1) I realize that flatwound strings will require a new setup and intonation, but considering how good the guitar is and how relatively little it cost me, the new strings and work will be an investment.

2) Listen,If you want an inexpensive guitar with great sound and easy playability, buy this one.

3) The strings were setup on the nut close to the fret board the way I like it already when I got it.

4) However, now that it's been fully set up and new strings installed, I am quite pleased with this guitar.

5) I am very happy with this Epiphone Les Paul and will proudly play and OWN it.

Bottom five sentences:

95) Get one, get it set up for a few bucks if you can't do it yourself, and you will have a good instrument.

96) It didn't take much to get it set up, small truss rod adjustment, new set of 10's for strings.

97) Bearing in mind that he had a financial incentive to steer me toward replacing them, he instead assured me that they were fairly decent, and advised against changing them unless a problem develops.

98) Other complaints included poor overall quality and terrible playability.

99) My LP Studio had it's pickups replaced with the Zakk Wylde 81/85's which made for good metal but it lost some of that bluesy tone that passives have.

After training several times with different parameters(including the c value), the result is remarkably improved. We can see the second sentence in the top five is a pretty good advertisement summarization for the product. Also, the 98^{th} sentence in the bottom five is correctly classified as a poor advertisement because of the negativeness of the content.

Then we applied the c value to the testing dataset, which is the review set for Amazon product: B0002CZVXM, a Strap Retainer provied by Jim Dunlop. The extracted text contains 62 sentences. The output advertisement summarization result is as follows:

1) Good Quality, great price, easy to install, easy to lock and unlock, fast delivery.
2) This is a great system, and I would highly recommend it for anyone who uses a strap on any electric guitar.
3) A good guitar is a big investment and you owe it to yourself to protect your investment with a good strap locking system.
4) They aren't much larger than the originals that came with the guitar and that extra piece of mind is great!
5) Very happy with my purchase.

The top three sentences show a good result for advertisement potential.

7 Conclusion

In this paper, we introduce a process for automatic advertisement creation and show how sentiment analysis and summarization might be combined to extract sentences that could serve as a good advertisement for the product when inserted into a generic template. We develop a Combined Rank Score to score the sentences according to their SO score and TextRank score and trained the model on several text sets in order to find a better c value before trying it out on the test set. As a result, we achieve relatively good candidate sentences in terms of enthusiasm for the product and coverage of its positive features.

References

Josh Bohde. 2012. Document summarization using textrank.

Kenneth Ward Church and Jeffrey D. Ullman. 1989. Word association norms, mutual information, and lexicography. *Proceedings of the 27th annual meeting on Association for Computational Linguistics*.

Isaac G. Councill, Ryan McDonald, and Leonid Velikovich. 2010. What's great and what's not: Learning to classify the scope of negation for improved sentiment analysis. *Proceedings of the Workshop on Negation and Speculation in Natural Language Processing*.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviewsl. *Proceedings of the 2004 ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Julian Mcauley, Christopher Targett, Qinfeng Shi, and Anton Vanden Hengel. 2015. Image-based recommendations on styles and substitutes. *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into texts. Association for Computational Linguistics.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab, November. Previous number = SIDL-WP-1999-0120.

Peter D. Turney. 2001. Thumbs up or thumbs down? semantic orientation applied to unsupervised classificiation of reviews.

Wikipedia. 2016. Sigmoid function — Wikipedia, the free encyclopedia. [Online; accessed 17-December-2016].