Final Project Report: Identifying Customer Complaints on Social Media

Group Members: Dylan Greenleaf, Jianzhe Liu, Yuanbo Wang, Antoine Rigaut

Problem Description

How do we identify, prioritize and classify customer complaints on social media and products? How can we facilitate response and decision-making for customer service? How do we resolve customer complaints quickly? How do we monitor brand image on Twitter and track social performance?

- Why is it important?

250%. This is the increase, from 2013 to 2015, in tweets directed to businesses (i.e. tweets using an at-mention such as *@company_name*). An increasing number of customers take to Twitter (instead of, say, your “support@company.com" address) and other social media to publicly complain about a brand’s products and/or services, or to simply make a request.

Customer complaints can become instantly viral on these platforms, with deleterious effects on brand image. In almost every case, the people complaining on Twitter are doing so because the company already failed to satisfy them in one or more traditional customer service channels. Failure to monitor brand image and customer service on social media can make firms vulnerable to situations that can quickly spiral out of control. The brand manager could be the last to know of something that could be potentially damaging.

Customers that resort to social media for customer service requests also have high expectations. They expect a fast response. Lithium Technologies, a leader in Social Customer Experience products and solutions, found that 53% of customers expect a response to their Twitter request within the hour. This proportion increases to 72% when the request is about a customer complaint.

- Who cares about it?

Many do care about it:
- Brand managers
- Customer service managers
- Marketing departments
- Product managers
- Public relations manager and communication departments

- Community manager, those in charge of handling a brand's social media communication

- Why does it remain unsolved?

It is difficult to sort the customer tweets into those that are legitimate complaints and those that are simply the musings of a potential Twitter troll. Moreover, Twitter does not close at 5 pm. In fact, customers may be more likely to use social media during the night and on weekends to complain. Where five years ago we could tolerate waiting 24 hours for a response from a customer service email, we are increasingly living in a real-time world where we want to know the answer right now. Hence we need a system that continually monitors social media.

This real-time issue, coupled with the huge number of tweets a company is receiving every day, creates a need to listen, prioritize and queue tweets. Response times need to go down, while focusing on genuine customer problems that the company can solve.

Objectives

- What are you proposing to do about the problem?

We created a customer complaint identification algorithm for Twitter, using tweets that mention the name of a brand either as an at-mention (@company-name) or as an hashtag (#company-name).

- How will you measure the success of your work?

There are two reasons why accuracy would be a poor choice as a metric:
- There is a class imbalance between customer complaints and non-customer complaints. This may result in an accuracy metric that is overly optimistic, since the classifier performs very well on the majority class but poorly on the minority class.
- The difference in the business cost of false positive and false negatives. It is relatively cheap for a human to go through the tweets classified as "complaints" and weed out the false positives - those that were classified as "complaints" but are not. False negatives are far more expensive to a business, because if we fail to classify an actual customer complaint as such, then there is a potential for bad reputation and customer anger. Therefore, we should optimize the false negatives rates, instead of the accuracy.

We used the AUC and recall as our metric of performance, in order to minimize the false negatives.

Related Work

- What have others (e.g., researchers, companies, etc.) done to address this problem?

Our work is closely related to the rising mantra “customer experience is the new marketing”, which proposes that in the era of social media the best way to market your brand is to solve problems for your customers.

However, most of past research has been focused on tracking brand sentiment, instead of customer experience. For instance, Misopoulos, Fotis, et al. have found that sentiment analysis could be used to analyze customer experience on Twitter. According to Chandra (2011), 30 percent of customers who wish to raise customer service issues do so via online social networking sites.

- What are you doing that is similar to past work?

Sentiment analysis has been used on the comments of online shopping websites before.

The following are some comments about American Airline on Twitter.

Topic modelling distinguishes different types of comments into different topics. The comments were also used for the sentiment analysis with hand-labeled comments. After the topic modeling was complete, an SVM model was fit using the gamma values as factors. The SVM model predicted the start score of a comment (out of 5) which was then compared to the actual star score. This sort of sentiment analysis on user comments can give a general glimpse into the strength of a product or a brand and help in making recommendations, dealing with complaints and even evaluating the performance of a product.

Compared with our project which aims to identify the customer complaints on Twitter, the comments under the product on online shopping websites are much easier for the companies to handle because they are not restricted to a certain number of characters. However, if we lack a CRM system, it would be much more difficult for the companies to monitor or distinguish the tweets. In addition, our project has more specific target on the complaints instead of the general scope.

- Are there commercial products that accomplish what you are trying to do? What are their characteristics? Where are their gaps?

Sprout Social:

Twitter has been collaborating with Sprout Social and Oracle to develop social customer experience data products. Sprout Social integrates social communications across Twitter, Facebook, LinkedIn, Instagram and Google+. Sprout Social offers a social CRM system that allows the customer service manager to view his social conversation history with his customers. It also give users some contextual information in the form of their customers' contact information, number of followers, etc. Sprout Social also provides key-word search and hashtag search features. Also, a dashboard provided by Sprout Social allows customer service managers to track some metrics of social engagement and general performance.

However Sprout Social does not offer specific features for identifying, queuing and responding to customer complaints. In fact, the product offers a better fit with the responsibilities of a community manager, than with a customer service manager.

Conversocial:

Conversocial aims to enable companies to effectively engage with their customers on Twitter through an integrated platform that allows the companies to effectively prioritize and monitor customer-care issues. According to Conversocial, their suite of products "is the leading solution for turning unstructured, chaotic social noise into organized and meaningful dialogue." The company is not very forthcoming in how they go about prioritizing conversations for the companies that subscribe to their platform but it is reasonable to assume that they have an algorithm that identifies high-urgency tweets based on contextual data.

- What about your work is novel? What gaps does it fill?

Most existing solutions do not prioritize between customer tweets or attempt to separate customer requests according to whether they relate to genuine customer problems or not.

Imagine two different tweets directed to an airline company. One takes issue with the delay in plane departure as a result of bad weather. Another customer has an urgent need for the items in his luggage, the luggage that was lost by the airlines company on a recent plane trip.

The difference between these two tweets is that the first is a customer problem which cannot be solved by the company without establishing control over the weather. The latter is a customer problem that the company can actually solve.

Most available commercial solutions do not attempt to make such distinctions between categories of customer requests and to identify customer complaints - genuine, urgent problems that the brand can solve for its customers. We want to complement existing solutions with a customer complaint algorithm that will sift through the masses of tweets directed at brands on a daily basis to identify the urgent, genuine problems that brands can solve for its customers.

Data

The first part of our project dealt with acquiring a data set of labeled tweets directed to two industries of interest in the American economy: commercial airlines and telecom companies. We streamed 6000 tweets from the Twitter streaming API that were directed to the following companies

- Commercial airlines: Delta, AmericanAir, SouthwestAir, United & JetBlue
- Telecom companies: Comcast, Verizon, AT&T, T-Mobile, Hulu, & DishNetwork.

These tweets had to be hand-labeled by humans as “complaints” or “non-complaints”. Our labeling design operated as following:

1. Divide the four-people team into two subteams and assign each subteam 3000 tweets (1500 airlines + 1500 telecom).
2. Each judge on the subteam independently labels the set of 3000 tweets as “complaint” or “non-complaint”.
3. When done, the subteams switch sets. Each team receives the two independent series of labels of the other team and solves any disagreement between the labels, assigning a definitive label to each tweet in the set.

A Cohen’s Kappa of 55% to 65% was observed.

Approach 1: Sentiment analysis

We first performed a “naïve” sentiment analysis on these tweets. It is “naïve” in the sense that we postulate that a positive vs. negative sentiment analysis can accurately classify tweets between customer complaints and non-customer complaints.

We first extracted the text of each tweet from our hand-labeled csv file. Then we used the NLTK package to separate all words in each tweet, removed irrelevant words, such as modal verbs and articles. Then we changed the verbs’ tense to present tense. After this, we attached a part-of-speech tag to each word in each tweet. At last, we calculated the sentiment score of each word by using its NLTK-assigned positive score minus its negative score.

The team then built a logistic regression classifier using the sentiment score as the sole predictor.

Approach 2: Bag-of-word logistic classifier

Our objective in this section was to build a linear classifier based on a bag-of-word representation of the tweets.

Tweets preprocessing

To obtain this bag-of-word representation, the team performed the following preprocessing steps.

We first used the Twitter-specific Part-of-Speech (POS) tokenizer and tagger built by Olutobi Owoputi & al. at Carnegie Mellon University. Information about the software and the conceptual approach is available at http://www.ark.cs.cmu.edu/TweetNLP/. The tagger is reported to have an accuracy of 93%.

We ran the tweets through the POS tagger and retained only the tokens which the tagger classified with a confidence above 60% in the following categories: *common noun, proper noun, proper noun + possessive, verb, proper noun + verbal, adjective, adverb, existential "there" + verbal* and *hashtags.*

This means that we suppressed, among other categories, *pronouns, nominal + possessive, interjections, determiner, pre- or postposition, coordinating conjunction, at-mention* (indicating a user as the recipient of a tweet), *URLs, numerals, punctuation and emoticons*. This POS tagger essentially performed a Twitter-specific stopwords removal

Our second step consisted in a series of standard preprocessing operations:

- Removal of stopwords from a standard list of English stopwords (as provided in R's *tm* package)
- Removal of words such as "https", "southwestair", "comcastdoesntcare", "americanairlines", "delta", "united", "deltaassist", "americanair", "jetblue", "comcast", "comcastcares","verizonsupport","vzwsupport", "verizon", "att", "attcares", "tmobilehelp", "dish", "hulu_support", "dish_answers", "hulu", "tmobile", and "comcastsucks". Those words were bound to occur in the tweets, given how we streamed the Twitter API, but they were not informative for a generalizable classifier.
- Word stemming
- Removal of all terms which have a length of 1 or 2 characters
- Removal of all terms that appear in 4 tweets or less. We contend that such terms would be too infrequent to be meaningful for a generalizable classifier. In addition, we need to reduce dimensionality to a manageable level. Removing these infrequent terms reduced the number of terms fivefold from 6000 to 1200.

Industry-indifference assumption

The team built a first bag-of-word based classifier by using lasso-regularized logistic regression on a dataset that consisted of a document-term matrix after preprocessing, as well as the

sentiment score of each tweet. Here, the team maintained the industry-indifference assumption and kept all tweets regardless of industry. The predictors in this model were hence the sentiment score and the word count for each term in a given tweet.

The penalty term was optimized using standard R packages to minimize variance, yielding a logistic model where the coefficients of the least informative terms were zeroed.

Our second step was to relax the industry-indifference assumption in order to carry out industry-specific model-building.

This was done by breaking down the Document-Term matrix by industry and carrying out separate lasso-regularized logistic regression on each dataset.

Our analysis hence yielded 3 models:

- A main classifier using all the tweets regardless of industry
- An airlines classifier
- A telecom classifier

Approach 3: SVM classifier based on LDA topics

Another approach that our team decided to take was to fit an SVM classification model to the data that we collected and labeled. The tweets were tokenized, combined into a corpus, preprocessed, and finally converted into a document-term matrix just as they were for the bag-of-words logistic classifier. After this was completed, a function to optimize the tuning parameters for the SVM was ran using the words as features. We quickly realized that the feature space was too large however, as the process was still running after 6 hours. We then set out to reduce the dimensionality of our feature space. After considering various options we decided to use LDA to reduce our feature space. LDA was ran on our document term matrix for topic number values ranging from 15 to 35 in increments of 5. Based simply on quick visual validation, the LDA model with 25 topics seemed to produce a greater proportion of coherent topics compared with the other models. The gamma values for each document given by this model were extracted and combined with the sentiment scores and complaint labels into a single data frame. Observations from this data set were then split into a training data set with 70% of the observations and a validation data set with 30% of the observations. We then ran the SVM tuning optimization function on the training data set with complaint as the response and sentiment score and the 25 gamma values as the features. We found that an epsilon value of 0.44 and a cost of 4 minimized the error of the model as determined through 10-fold cross-validation. The plot below shows the resulting errors of the various SVM models that were fit in optimizing the tuning parameters.

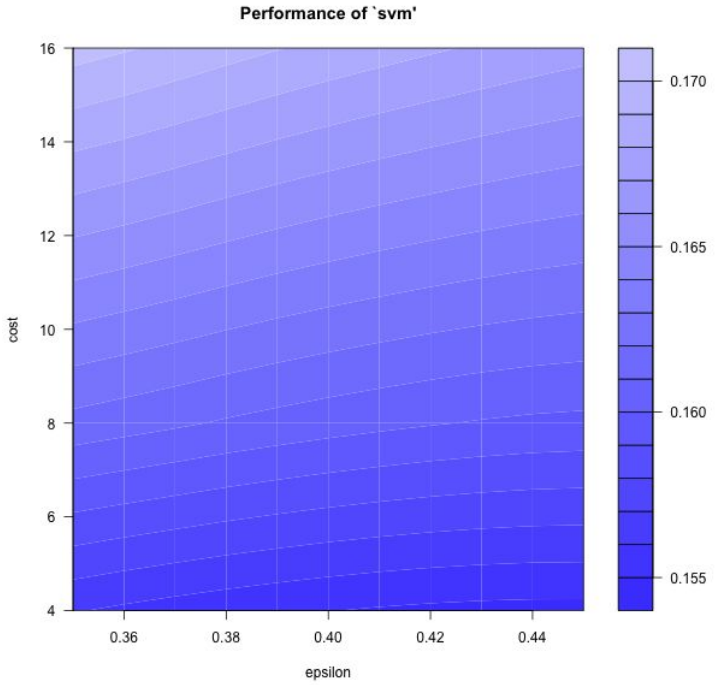

Figure 1. Performance of SVM depending on cost and epsilon

Discussion

Sentiment analysis results

The sentiment analysis found the sentiment score to be a statistically significant feature with a negative coefficient value, indicating that the more negative the tweet the more likely it is to be a complaint. However this model performed poorly as the resulting ROC curve had a meager AUC value of 0.56.

We still preserved the sentiment score as a predictor in later analyses, given the statistically significant relationship we found with complaint.

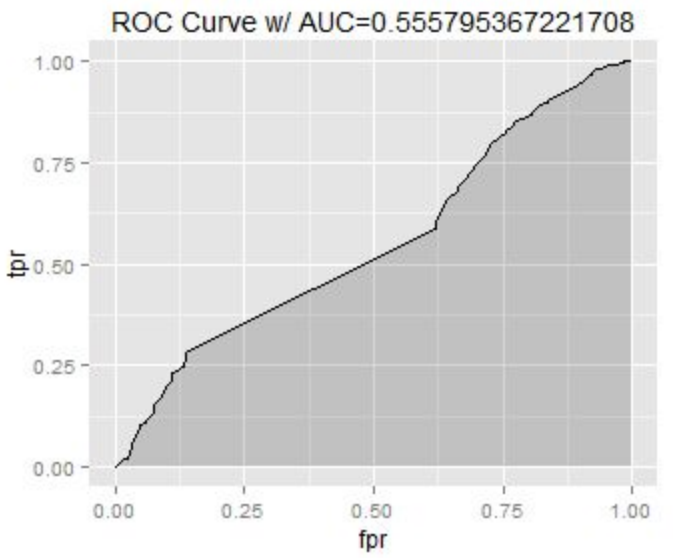

Figure 2. ROC curve of a logistic regression classifier using the sentiment score as the only predictor.

Logistic classifier results

We found our full logistic regression model to have 83% accuracy, 55% recall and 77% precision. It should be noted that lasso regularization zeroed the sentiment score parameter. Since companies tend to lay more emphasis on tweets that are complaints more than those that are not, we would like to increase our recall rate in the end.

We thus relaxed the industry-indifference assumption and carried out industry-specific model-building.

The following table is the top 15 most predictive terms of a complaint given by the main and industries model with their coefficients.

Model A: full	Coefficient	Model B: airlines	Coefficient	Model C: telecom	Coefficient
disappoint	2.58	charg	3.74	fraud	3.70
screw	2.45	enough	3.43	broke	3.51
human	2.42	philli	3.42	gotten	3.43
incompet	2.42	entertain	3.39	guid	3.35
delay	2.40	avoid	3.14	complain	3.30
complaint	2.38	refund	3.09	board	3.30
horribl	2.32	mess	3.07	disappoint	3.28
fraud	2.22	liter	2.94	spend	3.19
luggag	2.17	human	2.94	human	3.18
suggest	2.068	transfer	2.93	rip	3.13
appar	2.05	quit	2.81	delet	3.12
alon	1.98	worst	2.76	liter	2.94
unaccept	1.93	disappoint	2.73	stupid	2.92
suck	1.85	someth	2.67	receipt	2.83
refund	1.79	style	2.66	major	2.83

SVM Classifier Based on Topic Models Results

After the SVM model was fit it was then used to predict the probabilities for the validation set. Assigning validation set observations with probabilities greater than 0.25 as customer complaints achieved the best balance in our test metrics in terms of what we are interested in: 71% accuracy, 73% recall, and 50% precision. While the precision is less than ideal (only 1 of every 2 flagged tweets are actually complaints), this model does very well in terms of its recall as it only lets about 1 in 4 actual customer complaints slip through unnoticed. In addition to computing these metrics, we also plotted an ROC curve and calculated its AUC value, which turned out to be around 0.78.

- In what situations does your approach perform well? Provide examples.

Our approach depends on factors such as what's the minimum frequency of each word that we will take in order to build our document term matrix. (For example, at least 5 times has a better accuracy, recall and precision than at least 7 times).

As mentioned before, we are willing to sacrifice other precisions in order to have a higher recall rate. Therefore, if we overfit our model with more complaints data, then we can have a better recall outcome.

- Where does your approach break down? Provide examples.

We found that there were a lot of tweets containing url links. Some of these links were actually embedded pictures. For this case, we need to apply some image processing algorithms instead of the text mining tools. In other words, we need to understand the meaning of the picture the Twitter user attached to their tweets.

The following is an example illustrating the importance of this:

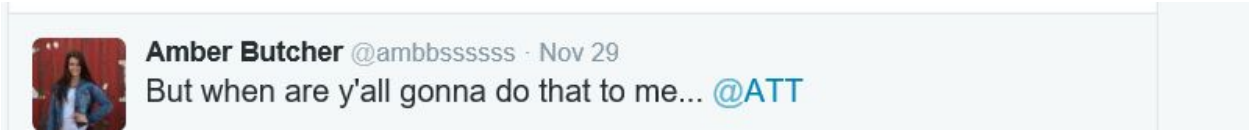

Figure 3. Example of tweets for image processing Part 1

If we ignored the picture below, we could hypothetically categorize this tweet as a complaint.

Figure 4. Example of tweets for image process Part 2

However, the picture shows that the AT&T randomly *increased* someone's data from 6GB to 12GB. It clearly shows that Amber also wants AT&T to increase her data but it is clearly not a customer complaint. Better understanding images that are posted with tweets would allow us to more effectively classify tweets.

- How does your approach stack up against other known approaches? Direct comparisons on shared test sets are best.

Our logistic regression models and our SVM classifier based on LDA topics all performed better than the the most widely known approach which is sentiment analysis using NLTK in python. The latter has an accuracy of 70.7%, recall of 29.1%, and precision of 45.0%.

Comparison of the Models

Model	Accuracy	Recall	Precision	AUC
Logistic with only sentiment analysis	74%	0.5%	30%	0.55
Logistic with all tweets	82%	54%	75%	0.88
Logistic for airline industry	81%	63%	74%	0.87
Logistic for telecom industry	84%	53%	72%	0.88
SVM with topics model	71%	73%	50%	0.77

The following graph is the ROC curve for each model that we built.

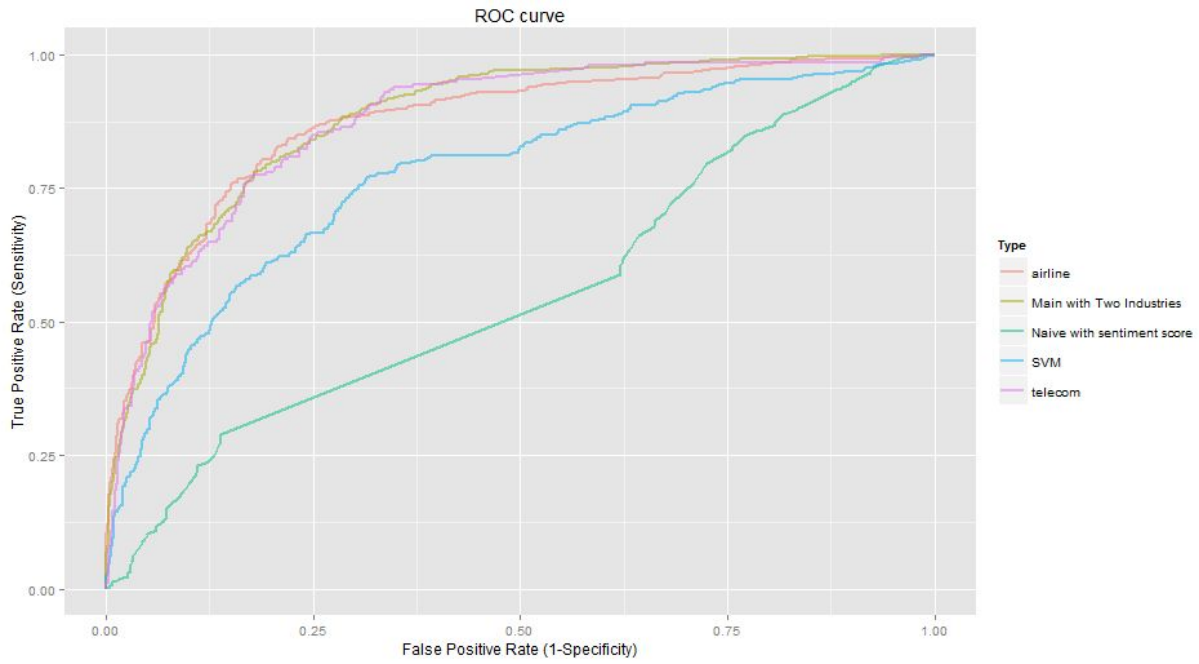

Figure 6. ROC plots for our 5 models

Conclusions and Recommendations

Our work taught us that while it is possible to create models that classify tweets as customer complaints with reasonable accuracy, it is much harder to do so in a manner that results in high recall values. Because of this, our models tended to let a lot of customer complaints pass through unnoticed, which is obviously not optimal. Thus, we believe that there is certainly room to improve upon the work that we have began in this project. A first step in building a better model would be to resolve the issue relating to ambiguity in determining whether or not any given tweet is a customer complaint. To address this we recommend working with industry experts with specific domain knowledge to acquire a greater amount of information on what they consider to be solvable customer complaints. Another step we could take that would lead to more robust models would be to acquire larger data sets made up of much more than just 6,000 tweets. We recommend accomplishing this through the use of Amazon Mechanical Turks.

Beyond improving the specific models that we have developed in this project, future work could also expand on this by developing a ranking algorithm that organizes tweets and prioritizes reducing response time to high-urgency customer complaints. At this point the collection of models becomes a viable data product with real-world value and could be incorporated into a web-application with a customer-service oriented dashboard.

References

Lithium.com. “Consumers will punish brands that fail to respond on Twitter quickly”. 2013. <http://www.lithium.com/company/news-room/press-releases/2013/consumers-will-punish-brands-that-fail-to-respond-on-twitter-quickly> (Accessed Nov. 19, 2015)

Misopoulos, Fotis, et al. "Uncovering customer service experiences with Twitter: the case of airline industry." *Management Decision* 52.4 (2014): 705-723.

McCullock, A., (May 11, 2015), “Coping with Demand for Social Customer Care in 2015”. *socialbakers.com.* <http://www.socialbakers.com/blog/2406-coping-with-demand-for-social-customer-care-in-2015> (Accessed Nov. 19, 2015)

Chandra, N. (2011), “Social media as a touch point in reverse logistics: scale development and validation”, International Journal of Business Research, Vol. 11 No. 3, pp. 76-83.

Twitter, “Customer Service on Twitter Playbook” <https://twitter.app.box.com/customer-service-on-twitter>