Sentiment Analysis on historical book reviews with a Bayesian Classifie

Maurits van Bellen
6148085

Bachelor thesis
Credits: 18 EC

Bachelor Opleiding Kunstmatige Intelligentie

University of Amsterdam
Faculty of Science
Science Park 904
1098 XH Amsterdam

Supervisor
Dr.M.W. van Someren

Informatics Institute
Faculty of Science
University of Amsterdam
Science Park 904
1098 XH Amsterdam

Juli 24th, 2010

Contents

Abstract

1 Introduction

Within the scholarly communication system a book review can be used as a mega-citation.[1] To do this two general elements are measured: The scholarly credibility and the writing quality. Scholarly credibility gives an indication of the quality of the research that, the reviewer thinks, the writer has performed. Thus, scholarly credibility is conceived in terms of the book's academic value, while writing quality is conceived in terms of the reviewer's assessment of the author's writing style. These values can be used to create a book reviewing credibility -quality scale: In their paper Zuccalla and Bod state that there might be keywords which indicate a degree of scholarly value. However it would be far too much work to read and grade every review by hand.

2 Motivation and research question

Doing all the reviews by hand is a very slow and time consuming process, consequently an intelligent system that can grade reviews ons scholarly credibility and writing quality is needed. The problem at hand is to recognize and classify the sentences within a review that give an indication of the scholarly credibility and the writing quality. However, the lingo used in these reviews does not simply state: "This book has great scholarly quality". Thus a system must be designed that can classify a sentence on its scholarly and writing sentiment.

3 Related work

Generally the task of Sentiment Analysis boils down to classifying a fragment of text that has an opinion towards an objector its features, as positive or negative. This is called Sentiment Classification. This classification can be based on the overall sentiments of the sentences in the document or on the other hand on the sentiment orientation of individual sentences. The work in this paper tries to classify a document by using a review and an combination of both levels of classification.

In [2] they use documents as text and use the bag-of-word approach to classify the documents. Each document is labeled as positive or negative. In [3] they describe how an ontology can help when classifying data. In our approach we will

be combining the ontology with the bag-of-words to classify a document based on its review.

4 Automatic Document Classification

Automatic document classification can be defined as a process in which a classifier program determines to which class a document belongs. The main objective of a classification is to assign an appropriate class to a document in the document set D, in respect to a class set C. The result is a set of pairs where each document d_i is assigned a class c_i to form a pair $< d_i, c_i >$ In order to use machine learning for automatic document classification, two sets of documents are required: a training set and a test set . A training set (T_r) is used to teach the machine learning algorithm the distinct features for each class on every document in the test set. The test set (T_e) is then used to evaluate the performance of the learned features on unseen documents.

5 Approach

In this section we will give a more detailed description of our approach and describe the methods and concepts involved. Our approach decomposes the Sentiment Analysis problem into two individual tasks. First our approach tries to recognize whether or not a sentence is of interest by using a domain ontology about historical book's (Interest Detection phase). For each of the classes we use a different ontology Then we use a Naive Bayes methodology to classify a sentence of importance. This two step approach gives us the ability to only classify the sentences which contain sentiment about the scholarly credibility or writing quality. Our work is based on a number of historical book reviews from "The American Historical Review".

5.1 Dataset

The dataset consists of Historical Book Reviews, in order to train the classifier and verify/evaluate its performance we split the data set in 2 separate parts. The first part is the training set and consist of 80 percent of the reviews, this set is used to train the classifier. The second set is the test set, this set consist of 20

5.2 Domain Ontology

In theory, an ontology is a "formal, explicit specification of a shared conceptualisation"In other words, an ontology is a formal representation of a set of concepts withing a domain and the relationships between these concepts. In the historical book review domain these concepts are limited to four possible subjects of interest. Our study shows that the most used concepts are:
"$Peopleof Interest, Research, Argumentation$ and $Work$". Thus the domain ontology is based on these four concepts where every concept has a number of keywords that populate this concept. Figure 2 shows a small part of the scholarly ontology. The keywords that populate the concepts have been handpicked by studying the subjects of the sentences that carry sentiment. Then these subjects were grouped into one of the concepts mentioned above. Because the lingua within the Historical Book review domain is somewhat limited, (There are only so many ways one can say book or author) this way of populating the ontology is sufficient.

5.3 Concept recognition

The next step in this phase is to recognize the concept of a new sentence. This is done by comparing each word in the sentence to each of the keywords in the ontology. Whenever one or more of these keywords have been located within a sentence we can say that this sentence is about the concept the keyword belongs to. For example if the sentence was "The book gives a excellent insight in the history of America during its Civil war." The word "book" would signal the concept "Work" and thus flagging the sentence as a sentence of interest. However, if a sentence does not contain any of the keywords, it is not flagged as a sentence of interest, and thus not used when classifying the sentences to determine the sentiment of the review.

5.4 Data preprocessing

The data was preprocessed by having an expert in the field of historical book reviews label them. Meaning that for every review all the sentences that contain either scholarly credibility of writing quality were marked. This gives the system an indication of what sentences in a review give an indication of the sentences which contain scholarly credibility and the sentences that contain writing quality. Furthermore every review was given a scholarly credibility and writing quality

value. This value was based on the number of sentences with scholarly credibility or writing quality in the review, for scholarly quality the following formula was used:
$\frac{1}{2} * (SCPlus - SCMinus)$
Where SCplus is the number of sentences indicating a positive scholarly credibility of the book and SCMinus the number of sentences indicating a negative scholarly credibility of the book and for writing quality:
$\frac{1}{2} * (WQPlus - WQMinus)$
Where WQPlus is the number of sentences indicating a positive writing quality of the book and WQMinus the number of sentences indicating a negative writing quality of the book. These scores are the scores used by Zuccalla and Bod to create a book reviewing credibility -quality scale.

5.5 Algorithm

Since our dataset consists of files which need classification based on the sentiment of the sentences in that file, a double classifier is required. The first level will handle classification of the individual sentences, while the next level classifies a file based on the number of sentences in each class.

5.6 First Level

In order to classify our individual sentences we used our labeled data, recall that an expert labeled the sentences carrying sentiment. Based on this data we created wordlist, containing every word in our labeled sentences. We then split this list in multiple lists based on their sentiment, this gave us the following lists:

- Scholarly Credibility Plus
  List containing all the sentences marked as positive scholarly credibility
- Scholarly Credibility Negative
  List containing all the sentences marked as negative scholarly credibility
- Writing Quality Plus
  List containing all the sentences marked as positive writing quality
- Writing Quality Plus
  List containing all the sentences marked as negative writing quality

Next we broke every sentence down into its individual words and counted these so that the score for each word per list would become:
$value = Count(WordperCategory)$

5.6.1 Stopwords

When using world-lists it is a common practice to use stopwords to filter out utterances that are very common in a language but carry no sentiment whatsoever. The goal of using stopwords is to remove these frequent utterances so they can not influence the calculation of the sentiment for a sentence. Without using these it is possible that the system learns that the word He is a very positive words because it is a frequent utterance and there are more positive sentences than negative ones. While the word He carries no real sentiment. In this case a stopwordlist for english was used. This list is attached in appendix A.

5.6.2 Standardization

Now we have four lists containing N words and the counts of these words, however these are still meaningless. The next step is to standardize these counts. To achieve this goal we simply count the total number of words in a list and multiply this value with the occurrences of these words: $TotalinCatagory = \sum_{i=1}^{n} Word_i * Count_i$. Then we divided every count with the total number of words in each list $count_i = \frac{Count_i}{TotalinCatagory}$. This was done because the dataset was slightly skewed towards the positive side. So if we were to use the total number of words in every list combined as the denominator, words used in negative ways would have a lesser value because it was a negative word. The resulting values are the chance of a category given a word is:
$P(Word|C) = \frac{Count}{TotalinCatagory}$
To determine the a priory chance of a word in a given category we calculated the a priory chance of the entire category by adding all the TotalinCatagory together and dividing each TotalinCatagory with this value :
$Allwords = \sum_{i=n}^{4} Totalwords_i$
and:
$P(Category) = \frac{TotalinCatagory}{Allwords}$
Thus the chance of a category given a word is the count of the word divided with the total number of words in this category multiplied, with the a priory chance for that category:

$P(Category|Word) = \frac{Count(word)}{TotalinCatagory} * \frac{TotalinCatagory}{Allwords}$
Ultimately this gives us the influence of a word on a category.

5.6.3 LaPlace Correction

While the previous steps provided us with a list of seen words and their influence on a category, we still have no way of dealing with unseen words. Even worse for every category an unseen word will return a chance of zero: :
$P(Category|unseenWord) = \frac{0}{TotalinCatagory} * \frac{TotalinCatagory}{Allwords} = 0$. So we need a way to deal with unseen words or we can not classify sentences which contain unseen words. A proposed solution is the LaPlace-Correction, for every seen word add one to the count and at two to the totalwords:
$P(Word|C) = \frac{Count+1}{TotalinCatagory+2}$
And for every unseen word use one as the count and add 2 to the totalwords:
$P(unseenWord|C) = \frac{1}{TotalinCatagory+2}$
Thus the resulting chances become:
for seen words : $P(Category|Word) = \frac{Count(word)+1}{TotalinCatagory+2} * \frac{TotalinCatagory}{Allwords}$
and for unseen words: $P(Category|Word) = \frac{1)}{TotalinCatagory+2} * \frac{TotalinCatagory}{Allwords}$

5.6.4 Naive Bayes

In order to classify a sentence based on the words that it contains, we use a Naive Bayes classifier. A Naive Bayes classifier is a statistical classifier commonly used for text based classifications. A Naive Bayes classifier makes the assumption that each word is independent of the next or previous words. And every sentence gets, for each class, a chance to belong to that class. This is achieved by calculating the chances for that class given the words in the sentence and dividing that by the total number of words in the sentence:
$P(C|W_1, W_2, W_3....W_n) = \sum_{i=1}^{n} \frac{P(C|W_i)*P(C)}{n}$
Then we look at which class has the highest chance and classify this sentence as that class:
$classify(W_i...W_n) = argmax(C = c) \sum_{i=1}^{n} \frac{P(c|W_i)*P(c)}{n}$
In our approach we decided to only distinguish between positive and negative for each category, so our Naive Bayes classifier was unable to classify a sentence as neutral.

5.6.5 Neutral Classification

Because our classifier is only able to distinguish between positive and negative for each category (Scholarly credibility and Writing Quality), we need a way to check whether a positive or negative flagged sentence should not be neutral. This is achieved by setting two thresholds, one for positive and one for negative. In order for a sentence to be flagged as either it's chance for that class must be equal or above the threshold set for that class. If it is not, the sentence is classified as neutral.

5.6.6 Boosting

In order to increase the performance of the classifier on the dataset we used boosting. Meaning that for each sentence in the training set we verified if the classifier had classified that sentence correctly by checking it with the marked sentences from the human expert. If a sentence was incorrectly classified our boosting algorithm would up the count of every word in this sentence with one in wordlist of the correct category. Thus making the words in that sentence more important indicators for that category. If however a sentence was classified correctly we changed nothing. After classifying and verifying every sentence in every review in the train set we would classify the sentences again with our classifier and our updated worldlist. This process was repeated until the performance of the classifier could not be improved further. The performance was measured by:
$PerformancePositive = \frac{TP-(FP+NP)}{TotalPositive}$
Where TP is is number of sentences correctly classified as positive, FP is number of sentences incorrectly classified as positive, NP is number of sentences failed to classify as positive and TotalPositive the total number of positively marked sentences by an expert in the trainingset.
$PerformanceNegative = \frac{TN-(FN+NN)}{TotalNegative}$
Where TN is is number of sentences correctly classified as negative, FN is number of sentences incorrectly classified as negative, NN is number of sentences failed to classify as negative and TotalNegative the total number of negative marked sentences by an expert in the trainingset. This gave us the formule to measure performance:
$Performance = \frac{abs(PerformancePositive+PerformanceNegative)}{2}$

5.6.7 Learning the thresholds

After reaching optimal performance we let the system learn the best thresholds for a positive classification and a negative classification per category. This is done by setting an initial threshold value and a δ. After classifying every sentence in the train-set we measure the performance with the above described formula. Then we add the δ to the threshold value and measure its performance after boosting again. If the performance has increased we keep adding δ to the threshold, after each change we make to the threshold we run boosting again to find the optimal wordlist. If it decreases we subtract δ instead. This is done until the performance converges.

5.7 Second Level

The second level takes the number of positive and negative classified sentences per category from the first level as its input and calculates the classification of the review based on this number. This done using the experts method by subtracting the amount of negative sentences from the amount of positive ones and dividing this by 2:
$\frac{SentencesPos - SentencesNeg}{2}$
We also introduced a new formula to test if the relation between $SentencesPos$ and $SentencesNeg$ is important.
$\frac{SentencesPos+1}{SentencesNeg+1}$
This was done to see whether or not we could improve on the experts method.

6 Results and Evaluation

In this chapter we will present the results of our approach, we will start with the results from our ontology. Next we will look at the results for the Sentiment classification of sentences and finally we will look at the results of the classification of the review. Before discussing the results we will explain some concepts used in this chapter.

	Program says yes	Program says no
Expert says yes	tp	fn
Expert says no	fp	tn

Table 1: Confusion table

Precision(P) = $\frac{tp}{tp+fp}$ Recall(R) = $\frac{tp}{tp+fn}$ Accuracy = $\frac{tp+tn}{tp+fp+fp+fn}$ F_1 = $\frac{2*P*R}{P+R}$

To evaluate our work we compared the classifications of the sentences and the documents that our program returned with those set by the expert.

6.1 Ontology

As stated before the goal of the ontology was to remove a large number of sentences that did carry sentiment but were however not about the book.

	Scholarly Credibility	Writing Quality
Without Ontology	1107	614
With Ontology	42	23

Table 2: Number of sentences classified as carrying a sentiment but not about the book

Table 2 shows the results of adding a the Ontology to the classifier. The number of sentences classified as carrying sentiment but not about the book is based on the total number of sentences that were classified as either positive or negative by our Naive Bayes classifier on the test set.

	Scholarly Credibility	Writing Quality
Without Ontology	0.10	0.07
With Ontology	0.84	0.67

Table 3: Results of the ontology on the classification

Table 3 shows the effect of the ontology on the precision of our classifier. However the use of this ontology did lower the amount of correctly classified sentences, as shown in table 4.This is because some of the sentences that did carry sentiment about the book were discarded by the ontology.

	Scholarly Credibility	Writing Quality
Without Ontology	132	63
With Ontology	129	52

Table 4: Number of correctly classified sentences

The positive effect is of the ontology is bigger on the Scholarly Credibility classification while the negative effect is bigger on the Writing Quality classification. We think this is due to the language used by the reviewers to describe

the different classes. The language used to describe the Scholarly Credibility has proven to be somewhat restricted. This however is not the case with the sentences describing Writing Quality, the range of words here is much larger and we most likely have not encountered enough in our train set. Our ontology is lacking the keywords to used describe Writing Quality because this data is so sparse.

Caute provides a conscientious overview of the texts
Many of the chapters of this book would work well in.....
He closely examines the debate itself ....
This book makes an important contribution to...
Lewis L. Goulds study guides readers through Senate history

Table 5: Examples of sentences describing Scholarly Credibility

Lewis uses content end notes to handle these historiographical conundrums...
The books title misleads somewhat"
Flynt writes in a pleasing narrative style that at times runs folksy...
He confuses giving satisfaction with exoneration, when it ....
Flynt fails to define the significant differences...

Table 6: Examples of sentences describing Writing Quality

As we can see from the tables 5 and 6 the sentences that describe Scholarly Credibility have a much more limited way to describe the work of the writer. It is mostly accompanied by a word such as book, text or study. On the other hand the language used in the sentences for Writing Quality is much more divers. This means that our ontology works best on the Scholarly Quality. It would be possible to create a better ontology for the Writing Quality however more data is required.

6.2 Sentiment Classification on sentence level

Following our approach we will discuss the results of classifying the sentences that our ontology gives us using our trained Naive Bayes classifier. We will also discuss the results of the different amounts of data while training the classifier.

We will discuss the results of the training separate for every category, starting with the Scholarly Credibility. Our classifier achieved an 81.4 percentage correct classification after boosting and learning the optimal boundaries.

To find the best threshold we did several experiments, in each experiment we let the classifier learn till it converged on a point. Once this was achieved we tried another experiment, this time with a different seed value for the threshold. Table 7 shows the results of some selected points.
As can be see in Table 7 the best results came at the 74.5 $\%$ mark. Meaning that a sentence has to have a chance of 0.71.5 or greater for a category to be assigned to that category. It also shows increasing the threshold generally leads to a better performance, until a certain point, as can be seen at number 4. This is because every sentence contains at least one word that is not bound to a clear indicator to either category, thus lowering the chance for this sentence to be classified as a category.

No	Thresholds	Accuracy($\%$)
1	P(C)$\geq$ 0.50	55.4
2	P(C)$\geq$ 0.70	72.8
3	P(C)$\geq$ 0.74.5	81.4
4	P(C)$\geq$ 0.90	71.2

Table 7: Number of correctly classified sentences

Next we looked at how the number of sentences we train on influenced the classification. Each review contains approximately 28 sentences and we increased the training set with 10 reviews at a time.

As figure 1 shows, the accuracy of the classifier increases almost linear until approximately 1200 sentences. At this point we start to see the line moving towards its horizontal asymptote. To get from 81.1$\%$ to 81.4$\%$ we needed 280 new sentences meaning that every sentence only adds 0.01$\%$ accuracy. This means that we are reaching optimal performance and adding more training data will not help much.

Next we look at the results on the Writing Quality category. As table 8 shows, the overall performance of the classifier on the Writing Quality is much lower. This is partly due to the ontology, if a sentence carrying sentiment about the writing is not passed on to the classifier it can not classify it. Unfortunately as shown before the ontology is needed to distinguish between sentences about the work and sentences not about the book. The uncertainty of the classifier is also much higher for this category, this is most likely due to the diversity of ways in which

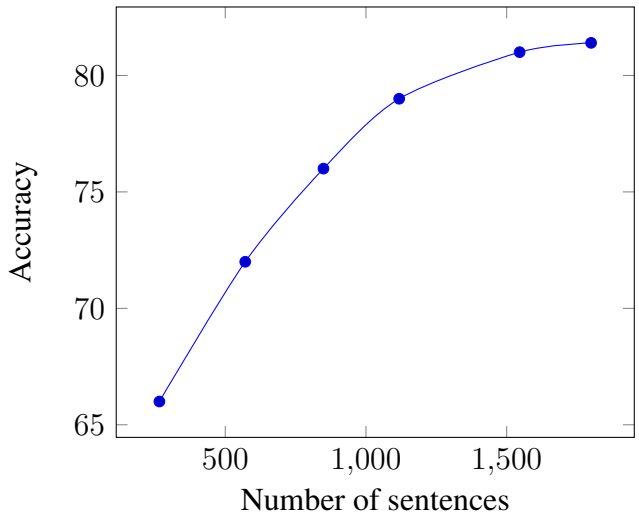

Figure 1: Accuracy %

the reviewers describe the writing quality of an author.

No	Thresholds	Accuracy(%)
1	P(C)≥ 0.50	50.8
2	P(C)≥ 0.60	59.6
3	P(C)≥ 0.65.5	67.4
4	P(C)≥ 0.90	20.4

Table 8: Number of correctly classified sentences

Furthermore we looked at how the number of sentences we train on influenced the classification. Each review contains approximately 28 sentences and we increased the training set with 10 reviews at a time.

As figure 2 shows the accuracy of the classifier increases much more linear than that of the Scholarly Credibility category. This implies that more data would greatly help this classifier, however it does seem that the performance of the classifier starts to peak near 70%. This is because the data on Writing Quality is much more sparse than the of the Scholarly Credibility, out of 2279 total sentences ,labeled by the expert, there are 484 sentences labeled as Scholarly Credibility sentiment and only 117 as Writing Quality. This combined with the diversity of writing made the Writing Quality data very sparse.

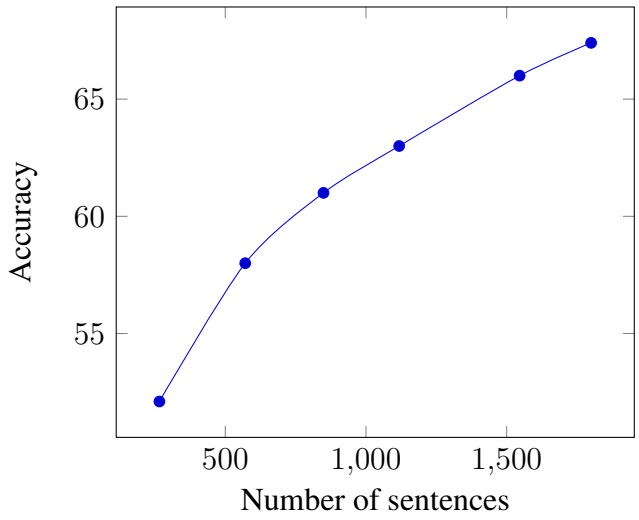

Figure 2: Accuracy %

6.3 Document Classification

In this chapter we will refer to $\frac{SentencesPos-SentencesNeg}{2}$ as method 1 and to $\frac{SentencesPos+1}{SentencesNeg+1}$ as method 2.

Category	Method	Threshold	Accuracy (%)
Positive	1	≥ 0.4	74
Positive	2	≥ 1.5	81
Neutral	1	$\leq 0.3 \& \geq -0.2$	74
Neutral	2	$\leq 1.4 \& \geq -1$	81
Negative	1	≤ 0.3	74
Negative	2	≤ 1	81

Table 9: Number of correctly classified sentences

The results in table 9 show that approach 2 has a higher accuracy than approach 1. This means that method 2 gives a better indication of the sentiment in the category then method 1 while using this classifier.

7 Conclusion

In conclusion we think we are now able to answer our research question, our approach is able to classify books based on their reviews using a Bayesian classifier when classifying the Scholarly Credibility, however it is much less accurate when classifying the Writing Quality. In order to work, our approach needs a lot of pre-labeled data done by an expert.

8 References

[1] Zuccala,A Bod, R. Book reviews as mega-citations: a fresh look at citation theory
[2]B. Liu. Sentiment analysis and subjectivity. Handbook of Natural Language Processing, pages 9781420085921, 2010.
[3]Kazemi, H.A Semi-Supervised Approach to Context-Based Sentiment Analysis
[4]Prabowo,R, Thelwall, M Sentiment Analysis: A Combined Approach

The dataset consists of Historical Book Reviews, in order to train the classier and verify/evaluate its performance we split the data set in 2 separate parts. The rst part is the training set and consist of 80 percent of the reviews, this set is used to train the classier. The second set is the test set, this set consist of 20.Â In conclusion we think we are now able to answer our research question, our ap-proach is able to classify books based on their reviews using a Bayesian classier when classifying the Scholarly Credibility, however it is much less accurate when classifying the Writing Quality. In order to work, our approach needs a lot of pre-labeled data done by an expert.Â Sentiment analysis and subjectivity. Handbook of Natural Language Processing, pages 9781420085921, 2010.