Combining Textual Pre-game Reports and Statistical Data for Predicting Success in the National Hockey League

Josh Weissbock and Diana Inkpen

University of Ottawa, Ottawa, Canada
{jweis035,diana.inkpen}@uottawa.ca

Abstract. In this paper, we create meta-classifiers to forecast success in the National Hockey League. We combine three classifiers that use various types of information. The first one uses as features numerical data and statistics collected during previous games. The last two classifiers use pre-game textual reports: one classifier uses words as features (unigrams, bigrams and trigrams) in order to detect the main ideas expressed in the texts and the second one uses features based on counts of positive and negative words in order to detect the opinions of the pre-game report writers. Our results show that meta classifiers that use the two data sources combined in various ways obtain better prediction accuracies than classifiers that use only numerical data or only textual data.

Keywords: machine learning, natural language processing, sentiment analysis, monte carlo method, ice hockey, NHL, national hockey league.

1 Introduction

Sports prediction, especially in ice hockey, is an application in which automatic classifiers cannot achieve high accuracy [1]. We believe this is due to the existence of an upper bound as a result of parity, the difference in skill between the best and worst teams, and the large role of random chance within the sport.

We expand upon our previous work in machine learning to forecast success in a single hockey game [1]. Similar to our previous model, we train a classifier on statistical data for each team participating. In addition, we use pre-game textual reports and we apply sentiment analysis techniques on them. We train a classifier on word-based features and another classifier on the counts of positive and negative words. We then use these individual classifiers and create a meta-classifier, feeding the outputs of the individual classifiers into the second level classifier, in a cascade. We compare several meta-classifiers, one uses a cascade-classifier and the other two use majority voting and the highest confidence of the first level classifiers.

This method returns an accuracy that improves upon our previous results and achieves an accuracy that is higher than “the crowd” (gambling odds) and expert statistical models by the hockey prediction website `http://puckprediction.com/`.

M. Sokolova and P. van Beek (Eds.): Canadian AI 2014, LNAI 8436, pp. 251–262, 2014.

This application is of interest to those who use meta-classifiers as it is successfully being used in an area that has little academic research and exposure, as well as improves upon the results from traditional approaches of a single classifier.

2 Background

There is little previous work in academic on sports predictions and machine learning for hockey. This is likely because the sport itself is difficult to predict due to the low number of events (goals) a match, and the level of international popularity for ice hockey is mcuh lower than other sports. Those who have explored machine learning for sports predictions have mainly looked at American Football, Basketball and Soccer.

Within hockey, machine learning techniques have been used to explore the attacker-defender interactions to predict the outcomes with an accuracy over 64.3% [2]. Data mining techniques have been used to analyze ice hockey and create a model to score each individual players contributions to the team [3]. Ridge regression to estimate an individual players contributions to his team's expected goals per 60 minutes has been analyzed [4]. Poisson process have been used to estimate rates at which National Hockey League (NHL) teams score and yield goals [5]. Statistical analysis of teams in the NHL when scoring and being scored against on the first goal of the game [6]. Due to the low number of events (goals) they found that the response to conceding the first goal plays a large role in which team wins. Using betting line data and a regression model and it has been found that teams in a desperate situation (e.g., facing elimination) play better than when not playing under such pressures [7].

Other sports have used machine learning to predict the outcome of games and of tournaments. In soccer, Neural Networks have achieved a 76.9% accuracy [8] in predicting the 2006 World Cup by training on each stage of the tournament (a total of 64 games). Neural Networks have also predicted the winners of games in the 2006 Soccer World Cup and achieved a 75% accuracy [9].

Machine learning has been used in American football with success. Neural networks have been employed to predict the outcome of National Football League (NFL) games using simple features such as total yardage, rushing yardage, time of possession, and turnover differentials [10]. Training on the first 13 weeks and testing on the 14th and 15th week of games they achieved 75% accuracy. Neural networks were able to predict individual games [11], at a similar accuracy of 78.6%, using four statistical categories of yards gained, rushing yards gained, turnover margin and time of possession.

Basketball has had plenty of coverage in game and playoff prediction with the use of machine learning. Basketball games can easily have over 100 events a night and this is reflected in the higher accuracies. In prediction of single games, neural networks have predicted at 74.33% [12], naive bayes predicts at 67% [13], multivariate linear regression predicts at 67% [14], and Support Vector Machines predict at 86.75% [15]. In terms of predicting playoff tournaments, Support Vector Machines trained on 2400 games over 10 years and predicted

30 playoff games with an accuracy of 55% (despite his higher accuracy over 240 regular games) [15]. Naive Bayes have been trained on 6 seasons of data to predict the 2011 NBA playoffs [16]. The prediction were that the Chicago Bulls will win the championship, but they were ultimate eliminated in the semi-finals.

In our previous work [1] we explored predicting the outcome of a single game in hockey. We used 14 different statistical data for each team as features. These features included both traditional statistics that are published by the league (e.g., Goals For, Goals Against, Wins, Location etc) and Performance Metrics which are used by hockey analysts (e.g. Offensive Zone Time Estimates, Estimations on the effects of Random Chance, Goals For/Against Rates). After trying a number of machine learning algorithms, our best results came from using a tuned SVM that acheived an accuracy of 59.3%. Further work showed that by using a voting meta-classifier with SVM, NaiveBayes and NeuralNetworks we could increase that accuracy to 59.8%. Using the Correlation-based Feature Subset Selection from Weka [17] we found the most important features to predicting a single game were: Goals For, Goals Against and Location. Traditional statistics outperformed the Performance Metrics in machine learning despite the fact that performance metrics have been shown to be better predictors in the long term.

3 Upper Bounds

We found in our previous experiments that no matter what we tried we were not successful in predicting the NHL with an accuracy higher than 60%. We decided to explore this further and it is our assumption that there is an upper bound that exists in sport predictions that makes it improbable to predict at 100%.

We used a method similar to Burke [18] who looked at prediction within the NFL by comparing observed, theoretical and a mixed-variation win/loss records. His findings conclude that the NFL has an upper bounds of approximately 76%. This seems to hold with the NFL-related research, as the authors have not been able to achieve higher results.

Rather than look at win/loss records we compared the observed win percentages of all teams between the 2005-2006 NHL season (since the last labour lockout) and 2011-2012 (the last full NHL season played) to a number of simulated seasons. The observed standard deviation (St.Dev) of win-percentage (win% — the number of games a team wins in the year that they play) over this time is 0.09.

Next, we simulated an NHL season 10,000 times, using the Monte Carlo method and on each iteration every team was given a random strength. When using the rule that the stronger team always wins ("all skill"), the St.Dev of win% is 0.3. When we changed the rule so that each team has a 50% chance of winning ("all-luck") the St.Dev of win% drops to 0.053. This suggests the observed NHL is closer to an "all-luck" league.

We changed the rule to determine who wins a match by varying the amount of random chance ("luck") and skill is required to win a game. If a randomly generated number is less than the pre-determined luck%, then the game has a 50% chance of being won; otherwise the strong team always wins. We varied the amount of luck and skill to win a game and we found the NHL was most similar to a league that is made up of 24% skill and 76% luck. The results of the various skill/luck Monte Carlo iterations can be seen in table 1, as well as the statistical tests to compare similarities to the observed win%.

Table 1. Monte Carlo Results

Luck	Skill	Theoretical Upper Bound	St.Dev Win%	F-Test p-value
0	100	100.0%	0.3000	4.90×10^{-16}
100	0	50.0%	0.0530	0.029
50	50	75.0%	0.1584	0.002
75	25	62.5%	0.0923	0.908
76	24	62.0%	0.8980	0.992
77	23	61.5%	0.0874	0.894

We can use statistical tests to identify which simulated distribution is most simular to our observed distribution. With a p-value of 0.992 it appears that the simulate league with 24% skill and 76% luck is the most similar to our observed data. To use the similar conclusion as [18], "The actual observed distribution of win-loss records in the NHL is indistinguishable from a league in which 76% of games are decided at random and not by the comparative strength of each opponent." What this means for machine learning is that the best classifier would be able to predict 24% of games correctly, and would be able to guess half of the other 76% of games. This suggests there is an upper bound for prediction in the NHL of $24\% + (76\%/2) = 62\%$.

4 Data

For the new experiments that we present in this paper, we used the data from all 720 NHL games in the 2012-2013 NHL shortened season, including pre-game texts that we were able to mine from `NHL.com`. The text report for each game discusses how the teams have been performing in the recent past and their chance of winning the upcoming game. Most reports are composed of two parts, one for each team. This was the case for 708 out of the 720 games. Since we need to extract separate features for each team, we used only these 708 pre-game reports in our current experiments. An example of textual report for one game can be seen in table 2.

We calculated statistical data for each game and team by processing the statistics after each game from the 2012-2013 schedule. As we learned in our previous work [1], the most important features were Goals Against, Goal Differential and Location. Given the difficulty of trying to recreate some of the performance metrics, we only used these three features in the numerical data classifier.

Table 2. Example of Pre-Game text, pre-processed

Text	Label
There are raised expectations in Ottawa as well after the Senators surprised last season by making the playoffs and forcing the top-seeded Rangers to a Game 7 before bowing out in the first round. During the offseason, captain Daniel Alfredsson decided to return for another season. The Senators added Marc Methot to their defense and Guillaume Latendresse up front, while their offensive nucleus should be bolstered by rookie Jacob Silfverberg, who made his NHL debut in the playoffs and will skate on the top line alongside scorers Jason Spezza and Milan Michalek. "I don't know him very well, but I like his attitude – he seems like a really driven kid and I think he wants to do well" Spezza told the Ottawa Citizen.	Win
Over the past two seasons, Ondrej Pavelec has established himself as a No. 1 goaltender in the League, and while Andrew Ladd, Evander Kane, Dustin Byfuglien and others in front of him will go a long way in determining Winnipeg's fortunes this season, it's the 25-year-old Pavelec who stands as the last line of defense. He posted a 29-28-9 record with a 2.91 goals-against average and .906 save percentage in 2011-12 and figures to be a workhorse in this shortened, 48-game campaign.	Loss

For the text classification experiments we used both traditional Natural Language Processing (NLP) features and Sentiment Analysis features. For the NLP features, after experimenting with a number of possibilities, we represented the text using Term Frequency/Inverse Document Frequency (TF-IDF) values, no stemmer and 1,2 and 3 grams. For the Sentiment Analysis we used the AFINN [19] sentiment dictionary to analyze our text. Other sentiment lexicons (MPQA [20] and Bing Lius [21] lexicon) were explored, but it was the AFINN lexicon that led to the best results in early trials. We computed three features: the number of positive words, the number of negative words and the percentage difference between the number of positive and the number of negative words $((\#positive\_words - \#negative\_words)/\#words)$.

As each pre-game report had two portions of text, one for the home team and one for the away team, we had two data vectors to train on for each game. In total, for 708 games, we had 1416 data vectors; each vector was from the perspective of the home and away team, respectively. The team statistical features were represented as the differentials between the two teams, similar to our method in our previous experiments [1].

5 Experiments

For the first experiment, we tried a cascade classifier. In the first layer, we trained separate classifiers on each of the three sets of features: the numerical features, the words in the textual reports, and the polarity features extracted from the textual reports, until the best results were achieved for each set. A number of Weka algorithms were attempted including MultilayerPerceptron (NeuralNetworks), NaiveBayes, Complement Naive Bayes, Multinomial NaiveBayes, LibSVM, SMO (Support Vector Machine), J48 (Decision Tree), JRip (rule-based learner), Logistic Regression, SimpleLog, and Simple NaiveBayes. The default parameters were used, as a large number of algorithms were being surveyed.

As we had 708 games to train on (and 1416 data sets), we split this up into 66% for training and the other 33% for testing. As each game had two data vectors, we ensured that no game was in both the training and the test set. In this way, when we received the output from all three classifiers in the first-layer, we knew which game the algorithm was outputting its guess for ("Win" or "Loss") and the confidence of the prediction.

The results from all three classifiers were post-processed in a format that Weka can read and was feed back into the Weka algorithms. The features that were used include the confidence of the classifiers' predictions and the label that was predicted. The labels for each game were either "Win" or "Loss". In the second layer of the cascade-classifier, the outputs from all three classifiers were feed back into the Weka algorithms (six features, two from each algorithm) and the new prediction results decided the final output class. We also used two other meta-classifiers: one chose the output based on the majority voting of the three predictions from the first layer and the other chose the class with the highest confidence.

Results of the first layer can be seen in table 3 and results of the second layer can be seen in table 4. Further details of the two layers of the meta-classifiers are presented in the following sections.

5.1 Numeric Classifiers

The numeric classifier used only team statistical data as features for both teams. As we learned from our previous experiments, the most helpful features to use are cumulative Goals Against and Differential, and Location (Home/Away). For each data vector, we represented the values of the teams as a differential between the two values, for each of the three features.

After surveying a number of machine learning algorithms the best results for this dataset came from using the Neural Network algorithm MultilayerPerceptron. The accuracy achieved on the testing data was 58.57%.

5.2 Word-Based Classifier

After experimenting with a number of Bag-of-Word options to represent the text, we settled on using the text-classifier with TF-IDF, no stemmer and 1,2 and 3 grams. Other options that were analyzed included: Bag-of-Words only, various stemmers and with and without bigrams and trigrams. The best result came from this combination.

In pre-processing the text, all stopwords were removed, as well as all punctuation marks. Stopwords were removed based on the Python NLTK 2.0 English stopword corpus. All text was converted to lowercase.

In a similar fashion to the Numeric Classifier, a number of machine learning algorithms were surveyed. The best accuracy came from using JRip, the rule-based learner, on the pre-game texts for both teams. The accuracy achieved on the same test data was 57.32%, just slightly lower than the numeric classifier.

5.3 Sentiment Analysis Classifier

The third and final classifier in the first level of the cascade-classifier is the Sentiment Analysis Classifier. This classifier uses the number of positive and negative words in the pre-game text, as well as the percentage of positive words differential in the text. These three features were feed into the algorithms in a similar fashion and the highest accuracy achieved was from Naive Bayes at 54.39%, lower than the other two classifiers.

Table 3. First Level Classifier Results

Classifier	Algorithm	Accuracy
Numeric	MultilayerPerceptron	58.58%
Text	JRip	57.32%
Sentiment Analysis	NaiveBayes	54.39%

5.4 Meta-classifier

In the second layer of the cascade-classifier, we fed the outputs from each of the three first-level classifiers. As we separated the testing and training data, we were able to label each game with the confidence of the predicted output from the three classifiers, as well as their actual output label. We then experimented with three different strategies. The first was to feed the data into machine learning classifiers, the second was to pick the output with the highest confidence, and the third was to use a majority vote of the three classifiers.

With the first approach, we surveyed a number of machine learning classifiers in the same fashion as the first layer. The highest accuracy came from the Support Vector Machine algorithm SMO and it was 58.58%.

For the next two approaches, we used a Python script to iterate through the data to generate a final decision and compare it to the actual label. In the first method of picking the choice of the highest confidence, the label of the classifier that had the highest confidence in its decision was selected. It achieved an accuracy of 57.53%. In the second approach, the three generated outputs were compared and the final decision was based on a majority vote from the three classifiers. This method returned an accuracy of 60.25%.

Table 4. Second Level Classifier Results

Method	Accuracy
Cascade Classifier using SVM (SMO)	58.78%
Highest Confidence	57.53%
Majority Voting	60.25%

For comparison, we placed all three features sets for each game into a single feature set and fed it into the same machine learning classifiers to see what accuracy is achieved and to compare it to the cascade-classifier. The results can be seen in table 5.

Table 5. All-in-One Classifier Results

Algorithm	Accuracy
NaiveBayes	54.47%
NaiveBayesSimple	58.27%
libSVM	51.56%
SMO	53.86%
JRip	54.62%
J48	50.20%

6 Results and Discussion

In order to put the results into perspective, we need a baseline to compare against. As each game has two data vectors, for win and for loss, a random choice baseline would have an accuracy of 50%. In hockey, there appears to be a home-field advantage where the home team wins 56% of matches; for our dataset, this heuristic would provide a baseline classifier with an accuracy of 56%. With an upper bound of 62% and a baseline of 56%, there is not a lot of room to see improvement with hockey predictions in the NHL. Other hockey leagues have higher upper bounds of prediction, but we could not find pre-game reports for other leagues to run a similar experiment on.

When analyzing the first level results in the cascade-classifier, the accuracy values are not that impressive. Sentiment analysis does worse than always selecting the home team. Using just the pre-game reports does better than the baseline of just selecting the home team, but does not do as well as the numeric data classifier. The classifier based on numerical features performs the best, and it is comparable with the numerical data classifiers that we tested in our provisional work [1], which used many advances statistics in addition to the ones that we selected for the current experiments.

When we look at the results of the second level of the cascade classifier, we see more interesting results. Using the machine learning algorithms on the output from the algorithms in the first layer, we see a little improvement. When we look at the methods of selecting the prediction with the highest confidence and the majority voting, the results improve even more with majority voting, achieving the best accuracy 60.25%.

It was surprising to see that the all-in-one data set did not do very well across all the algorithms that we had earlier surveyed. None of the accuracies were high; except for Naive Bayes Simple[1], none of these algorithms were able to achieve an accuracy higher than selecting the home team. This means that it was a good idea to train separate classifiers on the different features sets. The intuition behind this was that the numerical data provides a different perspective and source of knowledge for each game than the textual reports.

Overall, we feel confident that this method of a cascade classifier to forecast success in the NHL is successful and can predict with a fairly high accuracy, given the small gap of improvement available between 56% (home field advantage) and 62% (the upper bound).

For more comparison, we contrasted our results to PuckPrediction[2] which uses a proprietary statistical model to forecast success in games in the NHL season, each day, and compares their results to "the crowd" (gamblers odds). So far in the 2013-2014 season, PuckPrediction has made predictions on 498 games and the model has guessed 289 correct and 209 incorrectly (58.03%). The crowd has performed slightly better at 296 correct and 202 incorrect (59.44%). While predicting games in different seasons, our cascade-classifier method has achieved an accuracy that is higher than both of their methods. Additionally, their accuracies continue to suggest that it is improbable to predict at an accuracy higher than the upper bound of 62%, as the two external expert models have not broken this bound.

One interesting issue we discovered is which words are adding the most to the prediction. We looked at the top 20 InfoGain values of the word features, with the results seen in table 6. As we did not remove team or city names from the text, it is interesting to see that 7 of the top InfoGain values were referring to players, coaches and cities. This list has picked up on the team of Chicago Blackhawks, who had a very dominant season and ended up winning the NHL post-season tournament, the Stanley Cup Championship. The Pittsburgh Penguins were also considered a top team and had a high InfoGain value. Coach Barry Trotz of the Nashville Predators is a curious pick; it shows up 4 times and although the Nashville Predators were neither a very good or a very bad team in the 2012-2013 season; they did not have any activity that would make them stand out.

This suggests that it would be difficult to train on text across multiple years, as we would start to see evidence of concept drift, where the data the algorithms are learning on changes with time. A team might be really good in one year, but due to losing players in free agency and trade, may be a terrible team the next year. This suggests we should not be training and testing across more than a season or two.

---

[1] A Weka implementation of Naive Bayes where features are modelled with a normal distribution.

[2] `http://www.puckprediction.com`, accessed 15 December 2013.

Table 6. Info Gain values for the word-based features

Info Gain	ngram	Name/Place?
0.01256	whos hot	No
0.01150	whos	No
0.01124	hot	no
0.00840	three	no
0.00703	chicago	yes
0.00624	kind	no
0.00610	assists	no
0.00588	percentage	no
0.00551	trotz	yes
0.00540	games	no
0.00505	richards said	yes
0.00499	barry trotz	yes
0.00499	barry	yes
0.00499	coach barry	yes
0.00497	given	no
0.00491	four	no
0.00481	pittsburgh penguins	yes
0.00465	body	no
0.00463	save percentage	no

Similarly, we looked at the learnt decision tree from J48 on the pre-game texts and we can see a similar trend. With the top of the tree formed by ngrams of player and city names, this could have dramatic effects if you train on one year where the team is a championship contender and test on the next year when the team may not qualify for the post-season tournament.

7 Conclusion

In these experiments, we built meta-classifiers to forecast the outcome of games in the National Hockey League. In the first step, we trained three classifiers using three sets of features to represent the games. The first classifier was a numeric classifier and used cumulative Goals Against and Differential as well as the location (Home/Away) of both teams. The second classifier used pre-game texts that discuss how well the teams have been performing recently in the season up to that game. We used TF-IDF values on ngrams and did not stem our texts. The third classifier used sentiment analysis methods and counted the number of positive, negative and percentage of positive word differential in the texts.

The outputs were fed into the second layer of the cascade-classifier with the confidence and the predicted output from all three initial classifiers as input. We used machine learning algorithms on this set of six features. In addition, we used two other meta-classifiers, highest confidence and majority voting, to determine the output from the second layer. The best results came from the majority voting within the second layer.

This method returned an accuracy of 60.25% which is higher than any of the results from the first layer, much higher than the all-in-one classifier which uses all the features in a single data set, and it improves on our initial results from the numeric dataset from our previous work.

It is difficult to predict in the NHL as there is not a lot of room for improvement between the baseline and the upper bound. Selecting the home team to always win yields an accuracy of 56%, while the upper bound seems to be around 62%. This leaves us with only 6% to improve our classifier. While our experiments with numerical data from the game statistics were helping in the prediction task, we were happy to see that the pre-game report are also useful, especially when combining the two sources of information.

References

1. Weissbock, J., Viktor, H., Inkpen, D.: Use of performance metrics to forecast success in the national hockey league. In: European Conference on Machine Learning: Sports Analytics and Machine Learning Workshop (2013)
2. Morgan, S., Williams, M.D., Barnes, C.: Applying decision tree induction for identification of important attributes in one-versus-one player interactions: A hockey exemplar. Journal of Sports Sciences (ahead-of-print), 1–7 (2013)
3. Hipp, A., Mazlack, L.: Mining ice hockey: Continuous data flow analysis. In: IMMM 2011, The First International Conference on Advances in Information Mining and Management, pp. 31–36 (2011)
4. Macdonald, B.: An expected goals model for evaluating NHL teams and players. In: Proceedings of the 2012 MIT Sloan Sports Analytics Conference (2012), http://www.sloansportsconference.com
5. Buttrey, S.E., Washburn, A.R., Price, W.L.: Estimating NHL scoring rates. J. Quantitative Analysis in Sports 7 (2011)
6. Jones, M.B.: Responses to scoring or conceding the first goal in the NHL. Journal of Quantitative Analysis in Sports 7(3), 15 (2011)
7. Swartz, T.B., Tennakoon, A., Nathoo, F., Tsao, M., Sarohia, P.: Ups and downs: Team performance in best-of-seven playoff series. Journal of Quantitative Analysis in Sports 7(4) (2011)
8. Huang, K.Y., Chang, W.L.: A neural network method for prediction of 2006 world cup football game. In: The 2010 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. IEEE (2010)
9. Huang, K.Y., Chen, K.J.: Multilayer perceptron for prediction of 2006 world cup football game. Advances in Artificial Neural Systems 2011, 11 (2011)
10. Kahn, J.: Neural network prediction of NFL football games. World Wide Web electronic publication (2003), http://homepages.cae.wisc.edu/~ece539/project/f03/kahn.pdf
11. Purucker, M.C.: Neural network quarterbacking. IEEE Potentials 15(3), 9–15 (1996)
12. Loeffelholz, B., Bednar, E., Bauer, K.W.: Predicting NBA games using neural networks. Journal of Quantitative Analysis in Sports 5(1), 1–15 (2009)
13. Miljkovic, D., Gajic, L., Kovacevic, A., Konjovic, Z.: The use of data mining for basketball matches outcomes prediction. In: IEEE 2010 8th International Symposium on Intelligent Systems and Informatics (SISY), pp. 309–312 (2010)

14. Miljkovic, D., Gajic, L., Kovacevic, A., Konjovic, Z.: The use of data mining for basketball matches outcomes prediction. In: IEEE 2010 8th International Symposium on Intelligent Systems and Informatics (SISY), pp. 309–312 (2010)
15. Yang, J.B., Lu, C.H.: Predicting NBA championship by learning from history data (2012)
16. Wei, N.: Predicting the outcome of NBA playoffs using the naïve bayes algorithms. University of South Florida, College of Engineering (2011)
17. Witten, I.H., Frank, E.: Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann (2005)
18. Burke, B.: Luck and NFL outcomes 3 (2007), `http://www.advancednflstats.com/2007/08/luck-and-nfl-outcomes-3.html` (accessed February 16, 2014)
19. Nielsen, F.Å.: A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. arXiv preprint arXiv:1103.2903 (2011)
20. Wiebe, J., Wilson, T., Cardie, C.: Annotating expressions of opinions and emotions in language. Language Resources and Evaluation 39(2-3), 165–210 (2005)
21. Hu, M., Liu, B.: Mining and summarizing customer reviews. In: Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 168–177. ACM (2004)