A Sentiment model for Swedish with automatically created training data and handlers for language specific traits

Michelle Ludovici and Rebecka Weegar

Department of Computer and Systems Sciences, Stockholm University
Postbox 7003, 164 07 Kista, Sweden
michelleludovici@gmail.com, rebeckaw@dsv.su.se

1. Introduction

Sentiment analysis (also called opinion mining) aims to extract and classify subjective expressions from written text.

Extensive research has been done in the field of sentiment analysis. Programs, tools and models for classification of words, sentences and texts into sentiment categories are available. Sentiment categories are typically ”positive”, ”negative”, and ”neutral”. (Vinodhini and Chandrasekaran, 2012).

However, most of this work has been done for English, and there is a lack of openly available resources for other languages (Medhat et al., 2014). This is also true for Swedish, and therefore the aim of this study is to build and evaluate a model for sentiment analysis of Swedish text. Creating labelled training data is a time consuming task, therefore an approach with automatically labelled training data was applied.

Sentiment models can be trained on and applied to both general and specific domains. Sentiment models for specific domains, for example movie reviews, usually perform better than models aimed at general domains (Andreevskaia and Bergler, 2008). The model presented here have been trained and evaluated on news articles and the sentiment classification has been done on sentence level using support vector machines.

2. Materials

The data set used for training and evaluating the model originates from MittMedias database containing articles from five regional newspapers from the years 2002-2015. From this database articles in the topic ”school” were extracted, in total 16,865 articles with an average of about 15 sentences per article.

After splitting the articles into sentences, a number of preprocessing steps were applied. The preprocessing included tokenization, lemmatization using the MaltParser (Nivre et al., 2007), stop word filtering, and filtering of words only appearing once in the data set. The preprocessing reduced the number of unique features from 65,303 to 18,737.

3. Creation of training data

To reduce the need for manual annotation, two existing models were combined to label the sentences. Firstly, the sentences were translated into English using Google translate and classified into the categories positive and negative by the Stanford RNN classification algorithm (Socher et al., 2013). The algorithm classified 142,376 of the sentences as positive.

Secondly, the same sentences were classified using an openly available sentiment lexicon containing Swedish sentiment bearing verbs, adverbs and adjectives. The sentiment lexicon was created by seed word expansion on a small initial set. Synonyms and antonyms of the original seed words were iteratively added to the lexicon. The lexicon contains 5,641 classified words and multiword expressions. The words in the sentiment lexicon are classified as either positive or negative (Ludovici and Bignon, 2015b).

When classifying a sentence each verb, adverb, and adjective was extracted from the sentence. The polarity of each of these words was calculated using maximum likelihood estimation and the Naive Bayes assumption of independence. After assigning a polarity to the individual words, the sentence was classified according to a normalized sum of these probabilities (Ludovici and Bignon, 2015a).

The Naive Bayes classifier classified 181,234 sentences as positive and 80,182 sentences as negative. When the two classifiers agreed on sentiment category (positive/negative) the sentence was kept and otherwise discarded. The classifiers agreed on 22% (57,935) of the sentences.

The data was then split into three sets, a training set containing 79.45% of sentences, a validation set for parameter tuning containing 19.86% of the sentences and a test set containing 399 sentences.

4. Labelling of test data

The test set was manually labelled by 3 native Swedish speakers and used to evaluate the final model. The annotators classified sentences in the test set into the categories ”positive”, ”negative” and ”neutral”. Inter-annotator agreement calculated using Fleiss' kappa (Fleiss, 1971) on the test set was 0.69, corresponding to ”substantial agreement” (Landis and Koch, 1977).

5. Language specific handlers

A language specific handler is a method for adapting sentiment analysis to specific languages. Three language specific handlers were constructed with the intention to improve the accuracy of the sentiment classification by giving special consideration to features specific to Swedish. The three features were idiomatic expressions, phrasal verbs, and negations.

Idiomatic expressions and phrasal verbs are specific to one language and not likely to appear with the same form in other languages (Sjöholm, 1993), and they have in common with negations that they involve more than one word which have a particular meaning when considered together. The main idea of the language specific handlers is to aggregate these groups of words together into single features for the classifier.

The polarity (negative/positive) of negations and idiomatic expressions was determined by using a lexicon of sentiment bearing words an a list of classified idiomatic expressions (Ludovici and Bignon, 2015b).

A sentiment bearing word was determined to be negated if the negation cues *ej* or *inte* (non/not) appeared in a window of three words before or after the sentiment word. Negated sentiment bearing words were reversed in polarity by exchanging the cues and the words with either the word *bra* (good) or the word *dålig* (bad).

When an idiomatic expression was found, the whole expression was replaced with either *bra* or *dålig*.

The data was further matched against a list of phrasal verbs (Swedish phrasal verbs, 2015). Found phrasal verbs were aggregated into single words, for example *lära ut* (teach) would be rewritten as *lära_ut*.

6. The sentiment model

The Support vector machines (SVM) classifier was chosen since it has repeatedly been proven to be the best performing algorithm for sentiment analysis (Vinodhini and Chandrasekaran, 2012). Each sentence from the news articles corresponds to a data point and was represented by a numerical vector. The SVM algorithm tries to find a hyperplane that separates the vectors into the two sentiment categories.

The libSVM implementation of SVM (Chang and Lin, 2011) was used and the parameters of the model were tuned following the recommendations by Hsu et al. (2003). The investigated parameters were kernel type, C (cost), γ, feature weighting and scaling. Kernels can be used by SVM to map the data points to a higher dimensional space, which can be helpful if the data is not linearly separable. Here, the linear kernel and the radial basis function kernel (RBF) were evaluated.

The C parameter controls the cost of misclassification, a high C value may lead to overfitting and a low C to underfitting. The γ parameter impacts the shape of the separating plane. A lower γ gives a smoother separating plane between data points and a higher γ increases risk of overfitting. Grid search was performed to find a suitable ranges for the C and γ parameters.

Weighting of features decides a single features impact on the classification problem. Here, term-frequency - inverse document frequency (TF-IDF) was used. Several schemes are available for calculating TF-IDF, here three schemes have been evaluted; o(BM25)TF-Δ(k)IDF, Tfx-nfx and binary TF-IDF.

Scaling is done to scale the feature values of the vectors in the interval 0 to 1.

7. Evaluation and Results

The model was evaluated using precision, recall, F-score and accuracy. Accuracy was used to find the best combination of parameters on the validation set. The best result was achieved with RBF, without scaling and with binary TF-IDF. The optimal C value was 50 and γ was set to 0.0001. The model was finally evaluated on the test set with a precision of 89.50%, a recall of 82.42% and an F-score of 85.80.

The language specific handlers did not improve the results, individually or in combination. For most configurations the results were decreased when handlers were included, at best, the results were minimally improved.

8. Conclusions and future work

Even though the automatically created labelled data included mislabelled neutral sentences it was possible to train a model on the data that gave results comparable to previous studies. The results were on the lower end of those focusing on narrow domains and on the higher end of the range of results for sentiment classification in broad domains (Vinodhini and Chandrasekaran, 2012; Lambov et al., 2011). News text falls into the broad domains, but was narrowed by topic filtering.

Possible reasons for the language specific handlers not improving results is that they increase the feature space but provide little new information, and that individual handler values can be rare and therefore have little impact on the classification. Language specific features could be further investigated. Are there other features that would improve classification, or are there other classifiers more suited to deal with language specific traits?

Future work also includes refining the methods for labelling data. Semi-supervised techniques could perhaps help reduce errors introduced by the automatic approach.

References

A Andreevskaia and S Bergler. 2008. When specialists and generalists work together: Overcoming domain dependence in sentiment tagging. In *Proceedings of 46th Annual Meeting of the Association for Computational Linguistics*, pages 290–298.

CC Chang and CJ Lin. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27. Software available at http://www.csie.ntu.edu.tw/˜cjlin/libsvm.

JL Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378–382.

Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin. 2003. A practical guide to support vector classification. *Technical report*.

D Lambov, S Pais, and G Dias. 2011. Merged agreement algorithms for domain independent sentiment analysis. *Procedia-Social and Behavioral Sciences*, 27:248–257.

JR Landis and GG Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.

Ludovici and Bignon. 2015a. Sentiment analysis of

swedish texts. *Student report, Dept. of Computer and Systems Sciences, Stockholm University*.

Ludovici and Bignon. 2015b. *https://github.com/ michelle ludovici/SynonymProject*.

W Medhat, A Hassan, and H Korashy. 2014. Sentiment analysis algorithms and applications: A survey. *Ain Shams Engineering Journal*, 5(4):1093–1113.

J Nivre, J Hall, J Nilsson, A Chanev, G Eryigit, S Kübler, S Marinov, and E Marsi. 2007. MaltParser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(02):95–135.

K Sjöholm. 1993. Patterns of transferability among fixed expressions in second-language acquisition. *Current issues in European second language acquisition research*, 378:263.

R Socher, A Perelygin, JY Wu, J Chuang, CD Manning, AY Ng, and C Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the conference on empirical methods in natural language processing (EMNLP)*, pages 1631–1642.

Swedish phrasal verbs. 2015. *https://en.wiktionary.org /wiki/Appendix:Swedish_phrasal _verbs*.

G Vinodhini and RM Chandrasekaran. 2012. Sentiment analysis and opinion mining: a survey. *International Journal of Advanced Research in Computer Science and Software Engineering*, 2(6):282–292.