Sentiment Analysis with a Multilingual Pipeline

Daniella Bal[1], Malissa Bal[1], Arthur van Bunningen[2], Alexander Hogenboom[1], Frederik Hogenboom[1], Flavius Frasincar[1]

[1] Erasmus University Rotterdam
PO Box 1738, NL-3000 DR, Rotterdam, The Netherlands
{daniella.bal, malissa.bal}@xs4all.nl,
{hogenboom, fhogenboom, frasincar}@ese.eur.nl
[2] Teezir BV
Kanaalweg 17L-E, NL-3526 KL, Utrecht, The Netherlands
Arthur.van.Bunningen@teezir.com

Abstract. Sentiment analysis refers to retrieving an author's sentiment from a text. We analyze the differences that occur in sentiment scoring across languages. We present our experiments for the Dutch and English language based on forum, blog, news and social media texts available on the Web, where we focus on the differences in the use of a language and the effect of the grammar of a language on sentiment analysis. We propose a multilingual pipeline for evaluating how an author's sentiment is conveyed in different languages. We succeed in correctly classifying positive and negative texts with an accuracy of approximately 71% for English and 79% for Dutch. The evaluation of the results shows however that usage of common expressions, emoticons, slang language, irony, sarcasm, and cynicism, acronyms and different ways of negation in English prevent the underlying sentiment scores from being directly comparable.

1 Introduction

The Web, being available to a continually expanding audience all over the world, dramatically increases the availability of data in the form of reviews, previews, blogs, forums, social media, etc. Because of increased transparency and convenience of gathering data on the Web, global competition between companies is also highly increased. In this expanding competitive landscape it becomes more important for a company to understand its market and target audience. With the vast amounts of information available on the Web it is impossible for companies to read everything that is written about their position between competitors and competing products manually. Moreover, a company's staff would have to master many languages to read all the relevant texts.

Sentiment analysis comes to answer this need, as it can be used as a marketing tool to find out what people are generally saying about their products or products of their competitors. Sentiment analysis – also referred to as opinion mining – is the process of retrieving the sentiment or opinion of the author from a text [13]. Yet, for international companies it is not only important to know what people are saying about them in one language; they need to know as well

what is said about them across the world in many languages. Being able to analyze documents written in multiple languages gives companies the opportunity to assess local opinions which can be of great help when setting out a marketing plan across different geographies. It can help companies understand where clients adore or dislike their products. It can also help them understand why people in different areas around the world think differently about their products and where they need to focus their marketing efforts. However, it is still not possible to accurately compare the results of sentiment analysis across different languages. Existing work on sentiment analysis either does not focus on the comparability of sentiment across languages or assumes sentiment across languages to be comparable. We argue that the assumption of comparability of sentiment analysis across languages is too simplistic.

Therefore, we propose to utilize a novel language-independent method for analyzing sentiment in order to uncover structural differences in the way sentiment is conveyed in different languages. These insights may contribute to future endeavors in the field of cross-language sentiment analysis. To this end, our framework identifies sentiment in any considered language in the same way, albeit based on language-specific sentiment lexicons. We focus on documents written in the English and Dutch language. Here, documents are constructs of natural language text in the form of news articles, blog posts, forum posts or reviews, but also comments on a review or messages on a social media platform.

The remainder of this paper is structured as follows. Sections 2 and 3 discuss related work on sentiment analysis and our derived hypotheses respectively. Sections 4 and 5 then discuss our framework for multilingual sentiment analysis and its implementation, respectively. The evaluation of the proposed framework is discussed in Section 6. Finally, we conclude in Section 7.

2 Related Work

Current research on sentiment analysis in different languages [1, 4–6, 8, 9, 11, 14] focuses mainly on how to create new sentiment lexicons [9, 14] in a different language or on how to create a new pipeline for a new language [1, 5, 6, 8], both by mainly using machine learning techniques.

Moens et al. [11] analyze the creation of different pipelines used for different languages with minimal human effort to develop them. They stress the choices one has to make when applying a machine learning approach and discuss differences in choices between different languages. Additionally, several hypotheses are tested to optimize the output of the algorithm that computes the sentiment scores. Moens et al. recommend a pipeline consisting of three layers. The three-layer pipeline is meant to create a fast way of computing sentiment scores. The first layer is very fast in its computations but is not highly accurate. When a computation is not accurate (measured by thresholds) the text will be passed on to the next more precise, but also slower, computation layer. The process is repeated on the third layer. If still no accurate score is computed, the score of layer two is kept.

Bautin et al. [4] analyze cross-lingual sentiment by machine translation. They translate all considered texts to the English language and perform sentiment analysis on the translated results. The authors assume that the results are comparable and that the errors made by the machine translation do not significantly influence the results of the sentiment analysis.

Wan [14] focuses on the creation of a lexicon in Chinese, based on an English lexicon using a co-training approach. The idea of the co-training approach is to have two sets of documents – one in English and one in Chinese. These sets can be split in annotated and unannotated documents. The goal is to obtain a set of all unannotated English texts and all annotated Chinese texts. This is achieved by machine translation of all annotated English texts to Chinese and all unannotated Chinese texts to English. These two sets are then used to compute a classifier for Chinese texts and the classifier is tested on all Chinese reviews. To test the validity of the classifier, the reviews are translated to English and tested against the existing English classifier. If classified correctly in both languages, the classifier reflects the positive or negative polarity of the documents.

The research reported on by both Wan [14] and Bautain [4] has been on the area of creating a new sentiment analysis framework from an existing one and comparability of the sentiment of documents written in different languages is assumed. However, we believe this assumption is risky, as the grammar of languages is different to such extents that the best performing self-learning algorithms for classifying positive and negative sentiment texts in different languages are not the same [11]. This would also imply that, with the current research, companies can still not effectively assess local opinions. Differences in opinion score might as well be the result of differences that occur in the sentiment lexicon or the grammar of the language.

In this paper, we focus on creating a pipeline for sentiment analysis that uses multiple languages and takes language specifics into account. We thus aim to make the obtained results comparable across languages. Optimizing the accuracy of sentiment scores for single languages is already subject of widespread research and is out of our scope. We address the question whether it is possible to create a pipeline for sentiment analysis that can handle documents in multiple languages without sacrificing accuracy on sentiment scores. In order to answer this question, we aim to assess whether the sentiment scores of documents from different languages are comparable and whether differences in grammar or usage of language influence the working of the pipeline and the results.

3 Hypotheses

Two hypotheses could be derived from the related work, both of which can help to answer the research question. First, we hypothesize that the difference in pipelines for different languages causes a difference in the final document score. Moens et al. [11] show that the optimal pipeline for different languages looks different. This is caused by the grammar of the language. An example of these differences can be found in the way different languages handle negation.

For example, English negation is most accurately found by extending the effect of a negation word until the first punctuation, while in the Dutch language it is better to use a set window frame of words around the negation word [11]. Since we are comparing the document scores of documents written in different languages we propose a general pipeline, handling both languages, to minimize these differences in results.

Our second hypothesis is that differences in the way people express themselves cause unwanted differences in the sentiment analysis. An important difference between languages is in the way of speech. For example, when an Englishman says: "that is not bad at all", he means he is very enthusiastic about it, yet for the sentiment analysis it is just the negation of bad. This may give documents lower overall sentiment scores while the text is either very positive or very negative. Additionally, a Dutch person would prefer to say "Dit is een goede Universiteit" (This is a good University) rather than "Ik houd van deze Universiteit"(I love this University). For English people, the opposite holds true. We want to quantify how much effect these differences have on the final document sentiment score and how we can compensate for these differences [11].

4 SAMP Framework Design

To support our research goals, we have developed a framework for Sentiment Analysis with a Multilingual Pipeline (SAMP), which is composed of three parts. Each of the three parts and their respective goal is discussed in the sections below. We start with a quick overview of the framework and give a short introduction to the different components discussed.

4.1 Overview

As shown in Fig. 1, the framework consists of three main components:

1. **Language Selection** Determine the language of the text. Select either Dutch or English, based on the likelihood of the text to be written in the respective languages.
2. **Text Preparation** Text preparation consists of two parts. The first part is text cleaning. This involves replacing all capital letters with small letters, remove diacritics, etc. until a clean text remains. The second part involves word typing, i.e. finding the Part-Of-Speech (POS) of all words and determining whether each word exists in the lexicon to find the word category (i.e., opinion or modifier).
3. **Sentiment Score Computation** The sentiment score computation is divided in three separate steps. The first step is document sentiment scoring: calculate the sentiment score of a document. The second step is document relevance scoring, i.e., determining the relevance of a document with respect to an arbitrary topic. Step three is topic sentiment score computation, which involves computing a weighted average of the relevance and sentiment score of a batch of documents with respect to the topic of this collection.

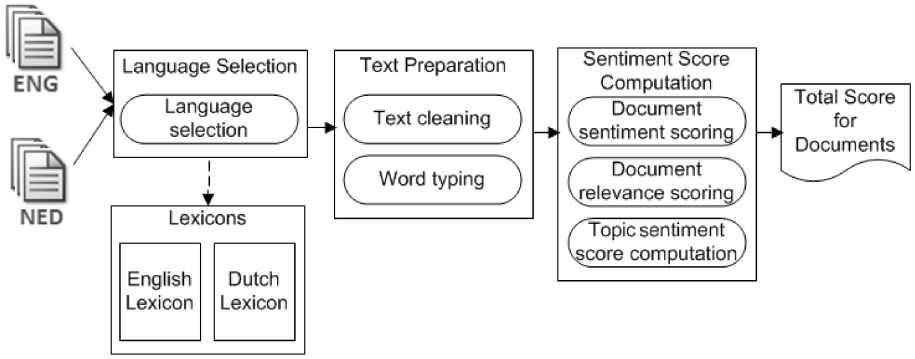

Fig. 1. The SAMP framework overview.

4.2 Language Selection

Algorithm 1 shows the algorithm used for the language selection. The input is a document or a list of documents. For each document, the language is determined as Dutch or English. The language is found with the help of letter n-grams. For example, if one would make letter 3-grams for the sentence "The cat was happy.", the grams are: ([The] [he␣] [e␣c] [␣ca] [cat] [at␣] [t␣w] [␣wa] [was] [as␣] [s␣h] [␣ha] [hap] [app] [ppy] [py.]), where "␣" is represents an empty space. Some n-grams are language-specific, e.g., [the] for English and [sch] for Dutch.

Nowadays, the usage of English words in the Dutch language – especially in short texts like tweets with a maximum of 140 characters – may confuse the document language detection components. We assume all documents to be either Dutch or English, but never a mix of both. The language with the highest likelihood is selected.

Algorithm 1: Language Selection.

input : Documents from batch D_{batch}
output: Lexicon Lex with the sentiment lexicon associated with the documents' language

```
1 nGrams = findNGrams(D_batch);
2 probDutch = compareTo(nGrams, Lexicon.NL);
3 probEnglish = compareTo(nGrams, Lexicon.EN);
4 if probDutch > probEnglish then
5     Lex = Lexicon.NL;
6 else
7     Lex = Lexicon.EN;
8 end
9 return Lex
```

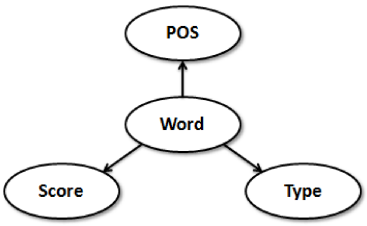

Fig. 2. Database entities.

We use two similar sentiment lexicons for English and Dutch. Both lexicons are simple structured databases with the entities in the database shown in Fig. 2. The Dutch lexicon is provided by Teezir. For each entry in the Dutch lexicon, the English lexicon contains a manual translation with the same meaning as the Dutch entry. A manual translation helps us provide two similar lexicons in order to rule out biased results due to differences in lexicons. The lexicon has been translated by three experts until they reached consensus.

4.3 Text Preparation

Before computing the sentiment of a batch of documents, we preprocess the documents by means of Algorithm 2. First of all, stop words like "a", "an", "the", etc., are removed, as they are of no use in the sentiment analysis and they unnecessarily affect the relevance score of a text. Furthermore, the documents are cleaned from diacritics (e.g., "ongeëvenaard" is replaced with "ongeevenaard") in order to optimize the likelihood of matches with words in the sentiment lexicon. Additionally, words are tagged with their associated POS. In this process, non-textual elements (e.g., tags, images, or videos) are not parsed by a POS tagger. The result is a batch of documents consisting of plain text which can be parsed and labeled.

Algorithm 2: Text preparation.

input : Documents from batch D_{batch} and their sentiment lexicon Lex
output: A batch of preprocessed documents D_{proc}

$D_{proc} = D_{batch}$.CleanText();
foreach *sentence* ***in*** D_{proc} **do**
    foreach *word* ***in*** *sentence* **do**
        $POS = sentence.FindPOS(word)$;
    end
    D_{proc}.addPOS(POS);
end
return D_{proc}

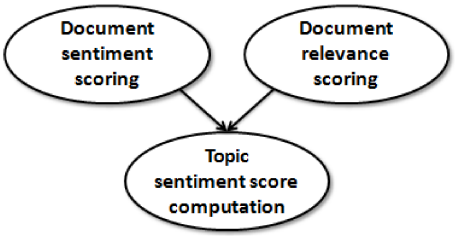

Fig. 3. Total sentiment score computation.

All words and their associated POS are subsequently compared to the lexicon of the matching language. Each word's sentiment score thus retrieved is considered in the sentiment computation, while taking into account the word type. We consider two types of words here – opinion terms and modifier terms. Modifier terms modify opinion terms. For example, in the sentence "The book was not good", "not" modifies the opinion term "good". For some words, it depends on their POS whether they are a modifier term or an opinion term. For example, in the sentence "The movie was pretty good", the adjective "pretty" is a modifier term as it increases the sentiment of the opinion term "good". Conversely, in the sentence "The blonde girl is very pretty", the adverb "pretty" is an opinion term as it expresses sentiment associated with the blonde girl.

4.4 Determine Sentiment Score

Fig. 3 shows the steps in creating a total score for a batch of documents. First, the sentiment of a document is determined. The sentiment S_i for document i is a function of all n opinion terms and their modifiers:

$$S_i = \frac{\sum_{o=1}^{n} \text{modifierterm}_o \times \text{opinionterm}_o}{n}, \tag{1}$$

assuming that opinion terms and modifier terms form couples. When a word does not have an opinion term, the value of the couple is set to 1.

In order to compute the sentiment associated with a batch of documents with respect to a certain topic, the sentiment scores S_i of each individual document i are weighted for their associated document relevance R_i:

$$\text{Topic Sentiment Score} = \frac{\sum_{i=1}^{n} S_i R_i}{\sum_{i=1}^{n} R_i}, \tag{2}$$

where the document relevance with respect to the query is computed based on the PL-2 standard [3] combined with document length normalization. Document length normalization means that longer texts have a relatively higher relevance (e.g., a news article is more relevant than a tweet of at most 140 characters).

5 SAMP Implementation

In order to evaluate our framework, we created a program that can parse texts in both Dutch and English and is designed to help the user find differences in the way texts are parsed and labeled. The application is written in C# and pre-processes texts with a maximum-entropy based POS tagger, which can process English as well as Dutch texts. Our implementation has three major user interfaces. We distinguish between a Specific Result Screen (SRS), a General Result Screen (GRS), and a Graph Screen (GS).

5.1 Specific Result Screen

The SRS, depicted in Fig. 4, is created to give more insight in the process of classifying a text as positive or negative. Annotated texts can be analyzed in both languages in order to analyze whether the pipeline works properly. Together with the GRS (described below), this screen also provides insight in the differences between languages. Because of the nature of the screen, documents can easily be analyzed manually. This analysis can reveal differences in the use of language, like those discussed in Section 6.

The text is displayed with positive opinion terms in green, negative opinion terms in red and modifier terms in purple ①. Below the text, a list of words is produced ② together with the type of the words (opinion terms or modifier terms) ③ and their score ④. Below the list, the screen shows the computation ⑤ and the document score ⑥. This way, the user can validate whether or not the correct words are flagged as sentiment words, the words have the correct score and the computation of the document score is correct.

5.2 General Result Screen

The GRS (Fig. 5) is designed to provide more insight in the process of computing sentiment scores for batches of documents. It shows the average score of the documents ①, how many documents constitute the score ② and a list of documents with their score, sorted on relevance ③. The user can thus analyze whether the documents are correctly labeled and ranked on relevance. This screen can help to provide answers for the second type of experiments.

5.3 Graph Screen

The GRS gives the possibility to show the GS presented in Fig. 6. The GS visualizes the distribution of document-level sentiment scores in the corpus. For example, if the score for both Dutch and English equals 0.50, it is not required that people in both countries think the same. It may very well be that, in The Netherlands, half of the texts result in a score of 0 and the other half results in a score of 1, while in England all the texts yield a 0.50 score.

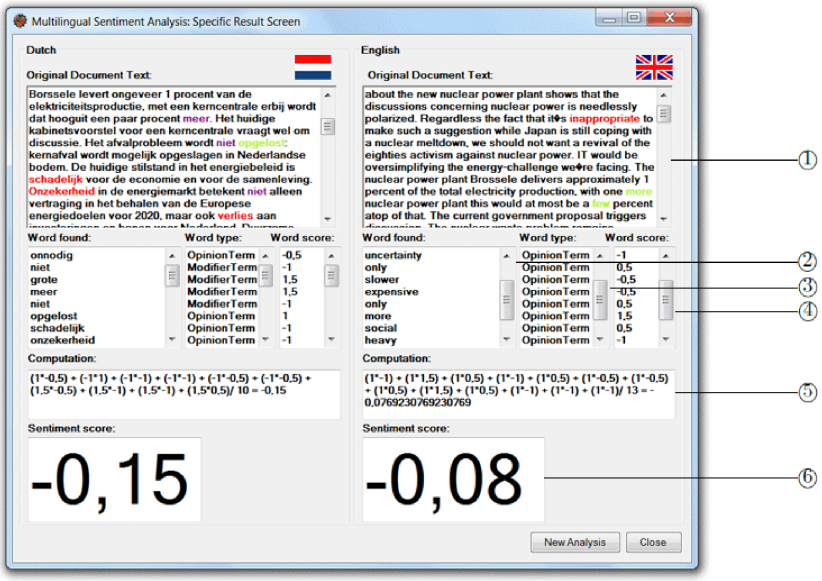

Fig. 4. Specific Result Screen.

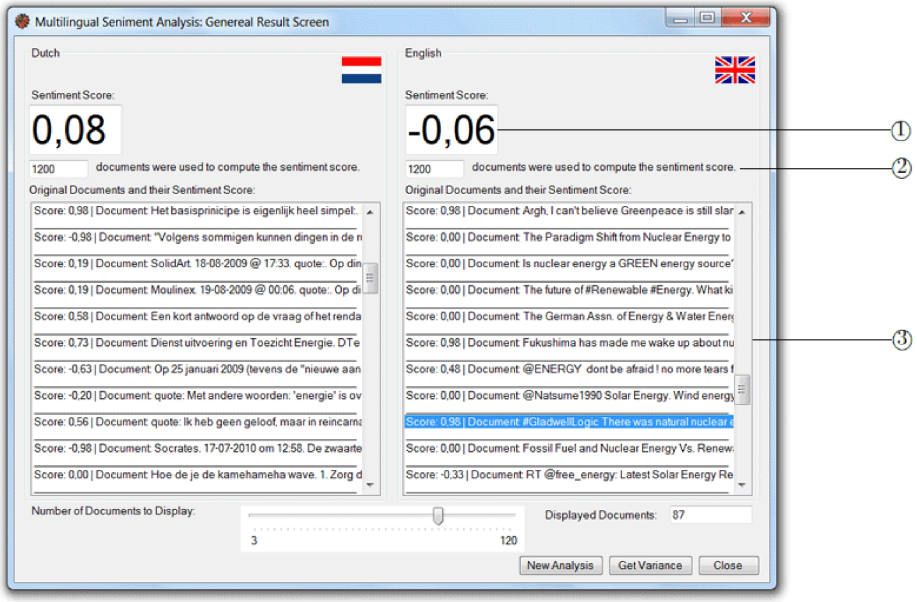

Fig. 5. General Result Screen.

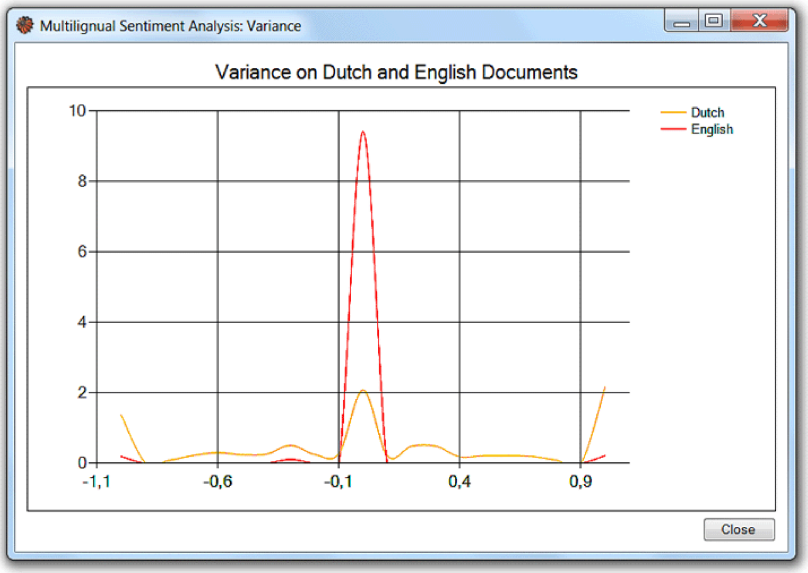

Fig. 6. Graph Screen.

6 Evaluation

For the evaluation of the performance of our framework, we first of all use 75 randomly selected Dutch movie reviews from the Dutch movie website FilmTotaal [7]. These reviews have been annotated on a five-star scaling from 1 to 5 by the respective writers of the reviews. We consider reviews with a rating of 1 or 2 stars to be negative, whereas reviews with higher ratings are considered to be positive.

Additionally, we use 75 randomly selected English movie reviews from the IMDb movie website [10]. These reviews have been annotated by their respective writers on a five-star scaling from 0 to 4. Pang and Lee [12] have already created positive and negative subsets of these reviews, which are readily available to us.

The sources of both considered sets of reviews have similar audiences between the age of 18 and 54, with no kids and some degree of college education. Our source of English reviews is slightly more favored by men compared to the source of Dutch reviews, whereas the source of Dutch reviews is favored by slightly better educated visitors [2].

For further (qualitative) analysis of differences in the way in which sentiment is conveyed in different languages, we consider two additional batches of documents, varying from news articles to blog posts, social media, reviews, etc. One batch consists of documents about the Xbox Kinect, whereas the other batch contains documents about nuclear power. These documents have been randomly selected from Dutch and English topic-related forums. Each document in this

collection is available in both English and Dutch. Documents have been translated and annotated for sentiment by three experts until they reached consensus. Per topic, half of the documents was originally published in Dutch, whereas the other half of documents was originally published in English.

The first additional batch of Kinect documents is chosen since the Kinect is relatively new in the console game industry. Therefore, a lot of information is generated on the Web in the form of product reviews, comments on social media and news articles. The great variety in information about the topic and the large daily amount of new information generated, results in a high likeliness of useful and reliable results when applying sentiment analysis.

In order to ensure that the results are not topic-dependent, our second additional batch contains documents about a topic completely different from consumer products. After the recent earthquake and subsequent tsunami disaster in Japan, nuclear power has recently become a well-debated issue. Different countries have different opinions about nuclear power mainly correlated to their dependency on nuclear power.

On our Dutch and English movie corpora, our pipeline shows to be approximately 79% and 71% accurate, respectively, when classifying documents as either positive (for sentiment scores equal to or higher than 0) or negative (for negative sentiment scores). Table 1 shows the confusion matrix for the Dutch and English movie corpora. Table 2 shows the recall, precision and F_1 measure for both positive and negative annotated texts for English and Dutch, as well as the overall accuracy per language and the macro-level F_1 measure. The accuracy of the language selection is approximately 95% for both languages.

Further analysis of our considered sets of documents reveals that we still miss the identification and correct interpretation of specific expressions like: "A piece of cake" and "Easy peasy". These sentences often carry sentiment, very different from the sentiment found when parsed and labeled with a normal dictionary. Some of the encountered Dutch expressions include: "Daar gaat Shahada *de mist mee in*" ("This is where Shahada messes up"), "Hij *blijkt niet in staat* om ..." ("Apparently he is not capable of ...") and "Deze 3D animatiefilm, *is in geen enkel opzicht het geld of de tijd waard*" ("This 3D animation is not worth spending any of your time and money on in any respect"), where the text in italics highlights the common expressions.

The next observation is the common use of slang language in documents. A few examples are: "Nevertheless, the cinematography (Sven Nykvist) and the sets (Mel Bourne) were pretty *blah*." and "If you like Russian style humor or if you like Monty Python style British humor, you will probably *go gaga* over this show.", where the italic part of the sentence highlights the slang.

We continue by pointing out the difference in expressing negation in English and Dutch. In English, negation tends to be more ambiguous than in Dutch. For example, in English, the word "no" can be used as a negation keyword, whereas the Dutch equivalent for "no" ("nee") cannot. Moreover, in English, negation can occur in the form of a verb with the affix "n't", which further complicates negation in the English language, as compared to negation in Dutch.

Table 1. Movie review classifications (columns) per target classification (rows).

	Positive Dutch/English	Negative Dutch/English
Positive	36/31	7/15
Negative	9/7	23/22

Table 2. Performance measures for movie review classifications.

	Precision pos/neg	Recall pos/neg	F_1 measure pos/neg	Accuracy	Macro F_1 measure
Dutch	0.80/0.77	0.84/0.72	0.82/0.74	0.79	0.78
English	0.82/0.59	0.67/0.76	0.74/0.66	0.71	0.80

Furthermore, the Dutch language knows semantics that are very hard to identify using our pipeline. For instance, the sentence "*Misschien dat sommige kinderen nog wel zullen lachen om de stupide grappen, maar of ouders daar nu blij mee moeten zijn?*" is a (rhetorical) question which carries no direct sentiment. Conversely, it suggests that the answer to the question would be negative and therefore the sentence appears to the reader to have a negative sentiment. Although the sentence cannot be directly translated in its true form a rough translation would be: "Maybe some of the children would laugh at such easy jokes, but would parents be happy with that?".

Many acronyms like "lol" ("laughing out loud"), "afaik" ("as far as I know") and "ga" ("go ahead"), but also emoticons like :D (happy emoticon), :'((crying emoticon) and -.- (extremely bored emoticon) are used to express feelings. Those emoticons and acronyms have great influence on the sentiment of small texts and often help humans to recognize the true sentiment of a text. For example, let us consider the next sentence that we came across in the Kinect batch: "That's... Freaking Awesome!". This sentence, as our parser interprets it, appears to show a very positive sentiment towards the Kinect. However, the actual sentence was: "That's... Freaking Awesome! -.- ", which shows a very strong negative sentiment towards the Kinect. In other words, the emoticon allows the reader to understand that the comment is meant as sarcasm, but the parser is oblivious of this additional information.

Of course, we also found many errors in the parsing of ways of speech like sarcasm, irony and cynicism. The sentence "What a fine movie this turned out to be!" could either be interpreted as a compliment because someone really thinks the movie turned out to be fine or it could be an expression of sarcasm when someone thinks the movie is rubbish. The current sentiment analysis will always interpret the sentence as having the first meaning.

Furthermore, in Dutch it is possible to split a verb. For example, in the sentence "De help-desk loste het probleem maar niet op." ("The help-desk failed to solve the problem"), "loste" and "op" are together a conjugation of the verb "oplossen" ("to solve"). Yet, the parser does not recognize this as such and misinterprets the word "loste" as "unloaded".

Finally, our results show that in English, the stars that are used to rate a review are better reflected by the sentiment score that is given to the document, as the English language is more based on explicitly mentioned sentiment. Conversely, the Dutch tend to have a more reserved way of expressing themselves. However, the results shown on the GS (Fig. 6) proved to be highly influenced by the large number of 0-sentiment documents in the chosen batches. Nevertheless, it did show for both batches that English documents either have no sentiment or very strong positive or negative sentiment while the sentiment in Dutch documents shows a more uniform distribution across the -1 to 1 scale.

7 Conclusions and Future Work

Our evaluation indicates that differences in grammar and usage of language indeed influence the results, e.g., because of common expressions, usage of slang language, difference in negation, Dutch semantics, acronyms and emoticons, sarcasm irony and cynicism, splitting of verbs in Dutch and the extremes in the English explicit way of expressing sentiment. Even though the sentiment classification in both languages remains comparable, it is currently not possible to compare the overall sentiment scores directly, as these scores are affected by many different language-specific phenomena, preventing these scores to be trustworthy representations of authors' sentiment. However, this does not imply that these differences cannot be dealt with. Looking at the current accuracy of approximately 79% for Dutch and 71% for English, we do believe it is possible to create a sentiment analysis pipeline that can handle documents of multiple languages, even without losing accuracy. These results suggest a need for normalization of sentiment scores in order to make them comparable across languages. Alternatively, differences across language can be dealt with explicitly.

To further implement multiple languages in a single pipeline, we would like to implement some of the findings in this paper, including detection of common expressions, acronyms and emoticons, difference in negation and usage of slang language. This will leave the challenges concerning semantics, sarcasm, irony and cynicism, that are very different for every language, to future research.

References

1. Abbasi, A., Chan, H., Salem, A.: Sentiment Analysis in Multiple Languages: Feature Selection for Opinion Classification in Web Forums. ACM Transactions on Information Systems 26(3) (2008)
2. Alexa Internet Inc.: Alexa the Web Information Company (2011), available online, `http://www.alexa.com/`
3. Amati, G., van Rijsbergen, C.: Probabilistic Models of Information Retrieval Based on Measuring the Divergence from Randomness. ACM Transactions on Information Systems 20(4), 375–389 (2002)
4. Bautin, M., Vijayarenu, L., Skiena, S.: International Sentiment Analysis for News and Blogs. In: 2nd International Conference on Weblogs and Social Media (ICWSM 2008). pp. 19–26. AAAI Press (2008)

5. Dai, W., Xue, G., Yang, Q., Yu, Y.: Co-clustering Based Classification. In: 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2007). pp. 210–219. ACM (2007)
6. Dai, W., Xue, G., Yang, Q., Yu, Y.: Transferring naive bayes classifiers for text classification. In: 22nd Association for the Advancement of Articifial Intelligence Conference on Artificial Intelligence (AAAI 2007). pp. 540–545. AAAI Press (2007)
7. FilmTotaal: Film Recensies en Reviews op FilmTotaal (2011), available online, `http://www.filmtotaal.nl/recensies.php`
8. Gliozzo, A., Strapparava, C.: Cross Language Text Categorization by Acquiring Multilingual Domain Models from Comparable Corpora. In: ACL Workshop on Building and Using Parallel Texts (ParaText 2005). pp. 9–16. ACL (2005)
9. Hofman, K., Jijkoun, V.: Generating a Non-English Subjectivity Lexicon: Relations that Matter. In: 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2009). pp. 398–405. ACL (2009)
10. IMDb.com Inc.: The Internet Movie Database (IMDb) (2011), available online, `http://www.imdb.com/`
11. Moens, M., Boiy, E.: A Machine Learning Approach to Sentiment Analysis in Multilingual Web Texts. Information Retrieval 12(5), 526–558 (2007)
12. Pang, B., Lee, L.: A Sentimental Education: Sentiment Analysis using Subjectivity Summarization based on Minimum Cuts. In: 42nd Annual Meeting of the Association for Computational Linguistics (ACL 2004)). pp. 271–280. ACL (2004)
13. Pang, B., Lee, L.: Opinion Mining and Sentiment Analysis. Foundations and Trends in Information Retrieval 2(1), 1–135 (2008)
14. Wan, X.: Co-Training for Cross-Lingual Sentiment Classification. In: Joint Conference of the 47th Annual Meeting of ACL and the 4th International Join Conference on Natural Language Processing of the AFNLP (ACL 2009). pp. 235–243. ACL (2009)