# Sentiment Analysis of Movie Reviews in Hindi

**Devansh Gupta** **Aviral Gupta** **Ajinkya Shivashankar** **Shivam Bansal**
Indian Institute of Technology Guwahati
Group No: 9
{gupta170101022, avira170101014, ajink170101004, shiva170101063}@iitg.ac.in

## Abstract

An important part of our information-gathering behavior has always been to find out what other people think. With the growing availability and popularity of regional language such as Hindi, new opportunities and challenges arise to actively use information technologies to seek out and understand the opinions of people using the regional languages.Reviews and opinions play a very important role in understanding peoples satisfaction regarding a particular entity. A number of benchmark datasets for various languages have been made available for sentiment analysis. In this paper we assess the challenges of sentiment analysis in Hindi by providing a benchmark setup, where we create an annotated dataset and build machine learning models for sentiment analysis in order to show the effective usage of the dataset.

## 1 Introduction

Hindi has 600+ million global speakers and is the third most spoken language. With the globalization of Internet, web pages catering information in Hindi are increasing at tremendous pace. These days users are actively checking reviews before buying a product or a service. It has become important for both the customer and seller to easily identify sentiment of a review for their decision making process. One has to extract and read many reviews before coming to a conclusion, which is not an easy task to perform.

It is important to find effective methods of analysing this growing body of Hindi text available on the internet. Sentiment analysis is a natural language processing task that deals with the extraction of opinion from a piece of text with respect to a topic (Pang and Lee, 2008). But a crucial precursor for any reliable research to be done in this field is to have a standard, representative dataset to compare findings and benchmark models.

Annotated dataset is certainly the foremost requirement for sentiment analysis. Current progress in sentiment analysis involving Hindi is hampered by the non-availability of benchmark datasets and therefore, a good dataset both in terms of quality and quantity will have a great impact on the overall system performance. Datasets, specific to Hindi languages created by few of the research groups are very few in number (mostly in few 1000s) (Joshi et al., 2010) (Balamurali et al., 2012)

We create a large dataset from IMDb reviews available on Hindi movies. Review is translated to Hindi using Google translate. While previously Google translate provided low-quality translations going from English to Hindi (Bakliwal et al., 2012), recent advancement has improved translation accuracy (Aiken and Wong, 2019), justifying returning to this approach. Additionally, Google translate is able to correctly translate Hinglish text i.e. Hindi words typed in English script. Score provided by users helps in having granular labels representing sentiment of accompanying review. Polarity of a review belongs to one of the two possible classes: positive and negative.

In our opinion, these are still a few limitations of using translation method for extracting reviews, there is a some possibility of losing the context information and sometimes we may have translation errors.

## 2 Related Works

(Joshi et al., 2010) proposed a machine translated based approach to sentiment analysis. They extracted and manually annotated 250 hindi movie reviews collected from online blogs. (Akhtar et al., 2016) proposed an aspect based approach for sentiment analysis. Conditional Random Filed (CRF)

and Support Vector Machine (SVM) machine learning algorithms were used for sentiment analysis. They crawled some of the online product websites covering various domains. Their dataset consists of 5,417 reviews having 2,290 positive, 712 negative, 2,226 neutral and 189 conflict reviews which were spread across 12 domains.Their dataset was annotated with an aspect term and it's corresponding sentiment.
(Arora et al., 2012) created a resource for Hindi Polarity Classification. They used Hindi WordNet to retrieve synonyms and antonyms of a given word in Hindi for which they knew the polarity and then assigned the similar polarity to synonyms and opposite polarity to antonyms. For their dataset they translated an already existing dataset in Hindi.

## 3 Method

### 3.1 Dataset

**Dataset generation:** We begin by finding relevant movies and films for review extraction. We filter all movies available on IMDb to get movies that: are in Hindi language, feature film length, and have a specified minimum number of votes to ensure movies that are relatively popular.

Next we fetch and extract users reviews for each film. Complete user reviews can get very large, which may result in a decreased translation accuracy. To mitigate this we consider only titles of reviews, which were then translated to Hindi script.

This dataset contains movie reviews along with their associated binary sentiment polarity labels. It is intended to serve as a benchmark for sentiment classification in Hindi.

The core dataset contains 27,025 reviews. The overall distribution of labels is balanced (14,143 positive reviews and 12,882 negative reviews).

Reviews with scores $\leq 4$ out of 10 are labelled negative, and $\geq 7$ out of 10 are labelled positive. Thus reviews with more neutral ratings are not included in the train/test sets. In the unsupervised set, reviews of any rating are included and there are an even number of reviews $> 5$ and $\leq 5$.

**Dataset organization:** The reviews are organized first by sentiment – positive or negative. Then they are further grouped by the movie the review belongs to. For ease of use of the dataset, two additional files are provided. One contains all positive reviews and the other contains all negative reviews. Each review is one line only, and each review is placed on its own new line.

### 3.2 Sentiment Classification

**Pre-processing:** Read dataset into memory to allow easier manipulation of data. Shuffle the data set, to properly mix up positive and negative reviews. Then we keep aside 25% of the reviews as an independent test set, using only 75% for training purpose. We remove any characters that are not part of Hindi script or punctuation marks. We remove Hindi language stop-words from the data set to increase performance of our models.

**Feature matrix generation:** Once the preprocessing is done we compute the feature matrix using the TF-IDF method. It is preferred over the bag-of-words model to extract better accuracy from our machine learning models.

- **TF-IDF algorithm**: It is used to convert a collection of raw documents to a matrix of TF-IDF features. The goal of using TF-IDF instead of the raw frequencies of occurrence of a token in a given document is to scale down the impact of tokens that occur very frequently in a given corpus and that are hence empirically less informative than features that occur in a small fraction of the training corpus.

**Classification:** We chose machine learning over deep learning as we wanted to retain control over the representation of data. Deep learning models are better utilised where the data representation also needs to be learned along with the patterns.

For classification, we utilise various models:

- **Support Vector Machine** (Platt et al., 1999): Support Vector Machine(also known as Support Vector Networks) is a supervised learning model which contains various learning algorithms to analyze classification and regression problems. It is also known as binary classifier which attempts to find a hyperplane that can separate two class of data by the largest margin. Given a set of points of two types in N-dimensional place, SVM generates a (N-1) dimensional hyperplane to separate those points into two groups. SVMs can be used to solve various real life problems like classification of images, recognizing hand written characters, categorizing text and hypertext, etc.

- **Multinomial Naive Bayes Classifier**: This is one variation of the Naive Bayes algorithm

fit to deal with the task of text classification (Kibriya et al., 2004). The family of Naive Bayes algorithms is a set of supervised learning methods developed by the application of Bayes' theorem with the naive assumption of conditional independence between every pair of features given the value of the class variable.

- **Decision Tree Classifier** (Breiman et al., 1984): Decision Tree classifiers are a non-parametric, supervised method of machine learning. Prediction is done by learning simple decision rules inferred from detected patterns in the data. They are particularly easy to train as they require little to none data preparation.

## 4 Results

Efforts put into generation of data set resulted in a mammoth data set of total 27,025 reviews in Hindi language in Hindi script.

| Sentiment class | Count |
|---|---|
| Positive sentiment | 14,143 |
| Negative sentiment | 12,882 |
| Total data points | 27,025 |

Applied machine learning models displayed similar accuracy results. Accuracy denotes percentage of accurate classifications on a held out test set.

| Model | Accuracy |
|---|---|
| Support Vector Machine | 75.8% |
| Naive Bayes | 74.3% |
| Decision Trees | 73.8% |

## 5 Conclusion

In this paper we propose a benchmark setup for sentiment analysis in Hindi. We have crawled the internet for movie reviews and annotated the data set with polarity classes. The data set comprises of labelled and unlabelled translated Hindi movie reviews crawled from the IMDb site. The resulting data set is the largest of its kind in open domain. The generated data set is a massive boon for future NLP research on Hindi language. We implement and run three different machine learning models for the task of sentiment classification based on text records.

### 5.1 Future Work

In future we would like to compare the performance our data set with other benchmark data sets containing text in Hindi language. Comparative tests may be carried out on sentiment classification task or any other NLP task.

In this paper we only used machine learning models, exploration of deep learning models is left for future work. These models of greater depth may result in greatly improved accuracy in the classification task.

## References


M Aiken and Z Wong. 2019. An updated evaluation of google translate accuracy. *Studies in linguistics and literature*, 3(3):253–260.

Md. Shad Akhtar, Asif Ekbal, and P. Bhattacharyya. 2016. Aspect based sentiment analysis in hindi: Resource creation and evaluation. In *LREC*.

Piyush Arora, Akshat Bakliwal, and Vasudeva Varma. 2012. Hindi subjective lexicon generation using wordnet graph traversal. *Int-l J. Computational Linguistics and Applications*, 3.

Akshat Bakliwal, Piyush Arora, and Vasudeva Varma. 2012. Hindi subjective lexicon: A lexical resource for hindi polarity classification. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC)*, pages 1189–1196.

AR Balamurali, Aditya Joshi, and Pushpak Bhattacharyya. 2012. Cross-lingual sentiment analysis for indian languages using linked wordnets. In *Proceedings of COLING 2012: Posters*, pages 73–82.

Leo Breiman, Jerome Friedman, Charles J Stone, and Richard A Olshen. 1984. *Classification and regression trees*. CRC press.

Aditya Joshi, AR Balamurali, Pushpak Bhattacharyya, et al. 2010. A fall-back strategy for sentiment analysis in hindi: a case study. *Proceedings of the 8th ICON*.

Ashraf M Kibriya, Eibe Frank, Bernhard Pfahringer, and Geoffrey Holmes. 2004. Multinomial naive bayes for text categorization revisited. In *Australasian Joint Conference on Artificial Intelligence*, pages 488–499. Springer.

Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Found. Trends Inf. Retr.*, 2(1–2):1–135.

John Platt et al. 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74.