Sentiment Analysis of Tweets in Three Indian Languages

Shanta Phani
Information Technology
IIEST, Shibpur
Howrah 711103, West Bengal, India
shantaphani@gmail.com

Shibamouli Lahiri
Computer Science and Engineering
University of Michigan
Ann Arbor, MI 48109
lahiri@umich.edu

Arindam Biswas
Information Technology
IIEST, Shibpur
Howrah 711103, West Bengal, India
abiswas@it.becs.ac.in

Abstract

In this paper, we describe the results of sentiment analysis on tweets in three Indian languages – Bengali, Hindi, and Tamil. We used the recently released SAIL dataset (Patra et al., 2015), and obtained state-of-the-art results in all three languages. Our features are simple, robust, scalable, and language-independent. Further, we show that these simple features provide better results than more complex and language-specific features, in two separate classification tasks. Detailed feature analysis and error analysis have been reported, along with learning curves for Hindi and Bengali.

1 Introduction

Sentiment Analysis (also known as *Opinion Mining*) refers to the problem of identifying the dominant sentiment in a given piece of text. The sentiment is usually modeled as a categorical variable with three values: positive, negative, and neutral. With the proliferation of social media data such as blogs, news articles and comments on them, YouTube comments, Amazon product reviews and Yelp reviews, online forum discussions, tweets, Facebook posts, and emails, we face an ever-increasing need to process this information and distill the evaluative sentiment present in these pieces of text, so that we can better identify and analyze the minds of the people – usually in order to make better policy decisions, be it in business or government.

Sentiment Analysis in Twitter data is relatively recent (we discuss relevant related work in Section 2), and sentiment analysis of tweets in Indian languages is more recent still. It was only last year, for example, that a sizable corpus of sentiment-annotated tweets was released as part of the SAIL task (Patra et al., 2015) in three different Indian languages – Bengali, Hindi, and Tamil.

In this paper, we have two goals:

1. Can we beat the performance of the systems that participated in the SAIL task?
2. Can we do so using a set of features that are simple, robust, scalable, and language-independent?

Note that language-independence is critical for Indian languages, because India has hundreds of languages,[1] and most of them are resource-poor.[2] Robustness and Scalability, on the other hand, are necessary to combat the exponential increase in content in Indian languages. At this point, it is useful to point out that the dominant categories of features used so far in sentiment analysis of Indian languages fail in at least one of the four criteria (Table 1). Syntax does not scale because we still do not have dependency parsers for Indian languages. WordNet is not robust (and does not scale) because it needs continuous improvement, hand-curation, regular maintenance, and management. Besides, its coverage is small. In this paper, we design a set of features that meet all four criteria, and still achieve state-of-the-art performance.

[1] https://en.wikipedia.org/wiki/Languages_of_India

[2] With more data, we can use deep learning for sentiment analysis (Zhang et al., 2015) for Indian languages, but we are not quite there yet. Indian languages still do not have sufficient annotated data for training convolutional neural networks.

Proceedings of the 6th Workshop on South and Southeast Asian Natural Language Processing,
pages 93–102, Osaka, Japan, December 11-17 2016.

	Simplicity	Robustness	Scalability	Language-independence
Syntax	FAIL	SO-SO	FAIL	FAIL
WordNet	OK	FAIL	FAIL	FAIL
N-grams	OK	OK	OK	OK
Surface	OK	OK	OK	OK

Table 1: Features used in sentiment analysis of Indian languages. First two rows are used in existing research, whereas we focus on the last two rows. WordNet refers esp. to SentiWordNet (Baccianella et al., 2010; Das and Bandyopadhyay, 2010).

The rest of this paper is organized as follows. We discuss relevant literature in Section 2. Section 3 gives details on the SAIL task, especially the data, task description, and our adaptation of it. We also describe our features, classifiers, and experimental methodology in this section. Section 4 provides experimental evaluation, along with feature ranking, error analysis, learning curves, and important insights. We conclude in Section 5, outlining our contributions, limitations, and directions for future research. Relevant terminology is introduced as and when they first appear in the paper.

2 Related Work

The overall task of sentiment analysis has been described in the books by Liu (2015), and Pang and Lee (2008). Essentially, the task is modeled as a three-way classification where a piece of text must be given one of the labels – positive, negative, or neutral. Sometimes the task is formulated as a regression problem, where a continuous output is desired. More details can be found in the surveys by Feldman (2013), and Montoyo et al. (2012).

The task of Twitter Sentiment Analysis is relatively recent. One of the first studies (Go et al., 2009) looked into this problem as a *query-driven classification* task. Using emoticons as (noisy) labels, authors achieved an accuracy above 80%. Subsequently, Pak and Paroubek (2010) created a corpus of 300,000 tweets (balanced between positive, negative and neutral classes) by querying happy and sad emoticons, and newswire tweets. The authors analyzed the relationship between POS tags and sentiment label of a tweet. A classification framework was then designed to investigate the relationships between training set size and test F-score, accuracy and negation words, accuracy and n-gram size, and salience vs. entropy.

Kouloumpis et al. (2011) showed that part-of-speech features are not very useful for Twitter sentiment analysis, whereas Agarwal et al. (2011) reported that POS-specific prior polarity features and tree kernels result in a 4% increase in accuracy over state-of-the-art. Zhang et al. (2011) performed Twitter sentiment analysis at the *entity level*, and Wang et al. (2012) reported a real-time system for Twitter sentiment analysis of US Presidential elections.

A. R. et al. (2012) reported the first study in cross-lingual sentiment analysis for Indian languages, where they showed that using WordNet senses as features can successfully bridge the language gap, achieving an accuracy improvement of 14%-15% over an approach that uses a bilingual dictionary. Sharma et al. (2014) reported a survey of sentiment analysis in Hindi, and Pandey and Govilkar (2015) proposed a system for sentiment analysis of Hindi movie reviews using Hindi SentiWordNet. Patra et al. (2015) reported the SAIL task, and the data released as part of it. Six teams submitted their systems. One of the best-performing systems is reported in (Kumar et al., 2015) that used distributional thesauri and sentence-level co-occurrences to expand Indian sentiment lexicons. They achieved an accuracy of 43.2% and 49.68% for the constrained submissions for Bengali and Hindi, respectively. A second system, reported in (Sarkar and Chakraborty, 2015) achieved constrained accuracy of 41.2% and 50.75% for Bengali and Hindi, respectively, using Multinomial Naive Bayes classifier. Finally, Akhtar et al. (2016) created an annotated dataset for aspect-based sentiment analysis in Hindi, consisting of Hindi product reviews crawled from multiple websites. The authors obtained an average F-score of 41.07% for aspect term extraction, and an accuracy of 54.05% for sentiment classification.

	Training data			Development data		
	Positive	Negative	Neutral	Positive	Negative	Neutral
Bengali	277	354	368	24	29	0
Hindi	168	559	495	18	19	19
Tamil	387	316	400	–	–	–

Table 2: Statistics of the SAIL dataset that we used. Each number represents the number of tweets in the corresponding category. Note that the values reported here may slightly differ from those reported in (Patra et al., 2015), because some tweets have been deleted by their authors.

3 Task Description

The SAIL task (Patra et al., 2015) released a set of sentiment-annotated tweets in three languages – Hindi, Bengali, and Tamil. The statistics are shown in Table 2. Each tweet was human-annotated as positive, negative, or neutral. Training data was released for all three languages, whereas development data was available for only two languages – Hindi and Bengali. We did not have access to the test data for any of the languages, so we performed our experiments only on training and development data.

The SAIL task defined two types of submission – *constrained* and *unconstrained*. In the constrained submission, participants were only allowed to use the corpora released as part of the task, and the Indian SentiWordNet from Das and Bandyopadhyay (2010). In the unconstrained submission, participants were additionally allowed to use any external resources such as POS-taggers, named entity recognizers, parsers, and additional data. Participants were requested to report the external resources they used.

At the end of the task, it was observed that constrained systems performed better than unconstrained ones (please see Table 3 of (Patra et al., 2015)). We therefore chose to work with the constrained version of the task. We ran two types of classification experiments: (1) **2-class classification:** positive and negative tweets only; (2) **3-class classification:** positive, negative, and neutral tweets. As mentioned before, we did not have access to the test data, so we performed stratified 10-fold cross-validation on the training data, chose the best model (features + classifier) from the cross-validation experiments, retrained the model on whole training data, and tested it on the development data. Final accuracy values are thus reported on the development data. Note that Tamil did not have a development dataset, so for Tamil we only report accuracy values from 10-fold cross-validation.

We experimented with four categories of features: (1) **Word n-grams** (n = 1, 2, 3) with and without stop words.[3], (2) **Character n-grams** (n = 1, 2, 3) with and without space characters and punctuation symbols, (3) **Surface features** (described later), and (4) **SentiWordNet features** (Das and Bandyopadhyay, 2010) (described later).

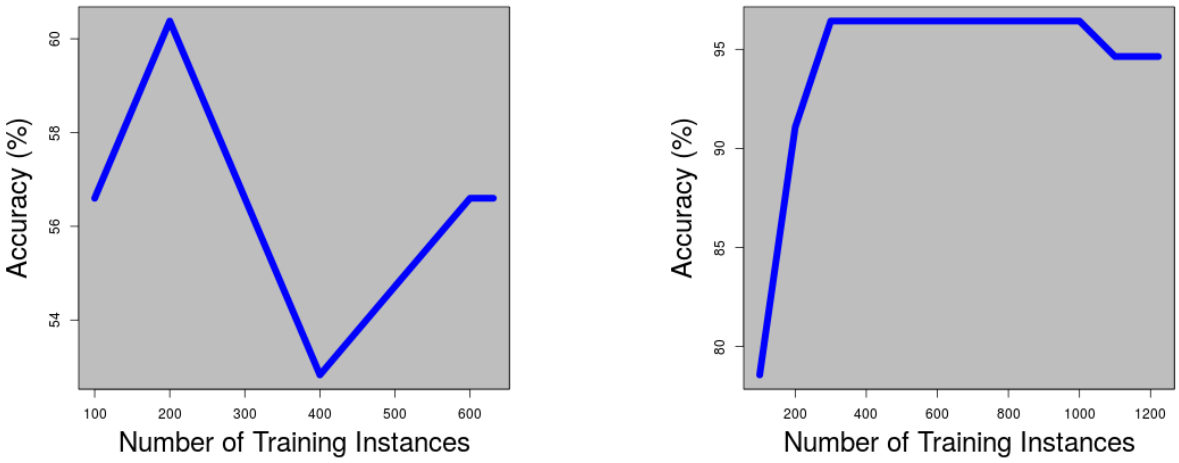
Note that the first three categories of features meet the simplicity, scalability, robustness, and language-independence criteria outlined in Section 1 (Table 1). For the word and character n-gram features, we experimented with three representations: binary (presence/absence), term frequency (tf), and tfidf. For the surface features, we used twelve of them, as follows: (1) Number of words in the tweet, (2) Number of characters in the tweet, (3) Number of hashtags in the tweet, (4) Number of English-character segments in the tweet, (5) Average English segment length in words, (6) Average English segment length in characters, (7) Number of "@" symbols in the tweet, (8) Number of "RT @" symbols in the tweet (*retweets*), (9) Number of "http:/" in the tweet (*hyperlinks*), (10) Number of punctuation characters in the tweet, (11) Number of punctuation characters, without leading and trailing periods, (12) Type-token ratio of the tweet (number of unique words divided by number of words).

Most of the surface features are derived from a manual inspection of the training data. For the SentiWordNet features, we constructed a vocabulary from all unique words given in the SentiWordNet files released by Das and Bandyopadhyay (2010). Then we used the following encoding: Positive word: +5,

[3]We used the Bengali and Hindi stop word lists available from `http://fire.irsi.res.in/fire/static/resources`, and combined them with the stop word lists available from `https://github.com/6/stopwords-json`. Tamil does not have a stop word list available online, so for Tamil we used all words.

1	করুন(0.008495)	1	शुभ(0.013068)	1	RT(0.014762)
2	ও(0.007727)	2	गणपति(0.008396)	2	மாதிரி(0.007428)
3	মানুষকে(0.007608)	3	विकास(0.008145)	3	ஒரு(0.006276)
4	চলে(0.005347)	4	रात्रि(0.007924)	4	இந்தியா(0.006193)
5	স্বাধীনতা(0.005317)	5	तेरी(0.007762)	5	பெண்(0.005044)
6	হয়নি(0.005224)	6	भगवान(0.007182)	6	கூட்டம்(0.004547)
7	বিরুদ্ধে(0.004992)	7	सहायता(0.006611)	7	என்ற(0.004177)
8	চাই(0.004845)	8	है(0.005303)	8	என்றால்(0.004078)
9	যত(0.004717)	9	माफ़(0.004996)	9	இந்த(0.003452)
10	গান(0.004509)	10	श्री(0.004764)	10	எப்படி(0.003425)
11	বিচারকদের(0.004318)	11	Twitter(0.004646)	11	இந்தியாவில்(0.003411)
12	কামনা(0.004270)	12	जीत(0.004527)	12	கேட்டா(0.003077)
13	তুলি(0.004240)	13	और(0.004441)	13	நான்(0.002980)
14	একটু(0.004225)	14	आपके(0.004422)	14	டேய்(0.002868)
15	প্রধানমন্ত্রী(0.004097)	15	चोट(0.004148)	15	அவங்க(0.002816)
16	দাও(0.004074)	16	बदलने(0.004004)	16	நீங்கள்(0.002770)
17	ImHassi(0.004066)	17	मोरया(0.003947)	17	இந்தப்(0.002759)
18	আনন্দ(0.004016)	18	हार्दिक(0.003940)	18	SettuSays(0.002680)
19	পারে।(0.003869)	19	आर्थिक(0.003839)	19	போதும்(0.002663)
20	দেবে(0.003748)	20	कंपनी(0.003782)	20	நானும்(0.002649)

Figure 1: Feature ranking for Bengali, Hindi, and Tamil. Top 20 features are shown in each case.



(a) Learning curve for Bengali

(b) Learning curve for Hindi

Figure 2: Learning curves for Bengali and Hindi. Y-axis is the % Accuracy on the **development set**.

Neutral word: +4, Negative word: +3, Ambiguous word: +2, and added up the scores for each unique word. So for example, if the word "ABCD" appears twice as positive and once as neutral, then its score will be $2 \times 5 + 4 = 14$. With these scores as our vocabulary, we experimented with three representations – binary, tf, and tfidf – as features. Note that SentiWordNet does not meet all the criteria outlined in Table 1. However, we still used it to compare other features with SentiWordNet, and see how it performs.

We used six different classifiers from the scikit-learn package (Pedregosa et al., 2011) with default parameter settings: **Multinomial Naive Bayes** (NB), **Logistic Regression** (LR), **Decision Tree** (DT), **Random Forest** (RF), **SVM SVC** (SV), and **SVM Linear SVC** (LS). In the next section, we will see how the combinations of different features and classifiers perform. Classifiers are written as "NB", "LR", etc.

4 Results

We show the results for 2-class classification in Table 3, and the results for 3-class classification in Table 4. Results from term frequency (tf) and tfidf representations have been omitted due to space restrictions;

Feature Representation	Feature Category	Feature Type	NB	LR	DT	RF	SV	LS
Bengali	Surface features		51.66	53.88	48.81	50.87	52.61	49.29
Binary (Presence/Absence)	SentiWordNet		56.1	56.1	56.1	56.1	56.1	56.1
	Word unigrams	AW	67.67	64.03	59.75	59.75	56.1	64.03
		NS	65.61	62.28	56.89	60.22	56.1	62.28
		OS	56.26	56.89	55.94	56.74	56.1	56.89
	Word bigrams	AW	60.86	58.64	55.47	59.27	56.1	57.53
		NS	57.21	58.95	51.51	58.64	56.1	57.69
		OS	56.42	55.78	55.78	55.47	56.1	54.83
	Word trigrams	AW	57.37	59.43	51.19	53.25	56.1	58.95
		NS	56.26	59.43	53.09	53.09	56.1	58.95
		OS	56.1	56.1	55.94	56.1	56.1	56.1
	Character unigrams	AC	55.78	57.05	50.55	56.89	55.63	57.69
		SS	55.78	56.89	49.92	56.58	55.47	57.37
		PP	53.88	56.42	49.6	56.42	56.74	56.58
		SP	53.88	56.58	52.3	54.83	56.74	56.58
	Character bigrams	AC	51.82	53.09	53.88	58.16	56.1	51.66
		SS	51.19	52.93	55.78	56.89	56.1	52.3
		PP	52.46	53.41	54.04	58.8	56.1	54.2
		SP	52.14	54.52	58.8	59.75	56.1	54.68
	Character trigrams	AC	54.04	56.1	53.57	58.16	56.1	53.72
		SS	50.55	52.14	49.45	55.47	56.1	51.98
		PP	52.77	55.47	53.72	60.38	56.1	52.61
		SP	52.3	51.51	54.68	58.32	56.1	50.71
Hindi	Surface features		64.79	75.79	66.99	75.1	78.4	65.75
Binary (Presence/Absence)	SentiWordNet		76.89	76.89	76.89	76.89	76.89	76.89
	Word unigrams	AW	78.68	81.57	76.62	79.92	76.89	80.61
		NS	73.04	80.33	75.93	79.78	76.89	79.78
		OS	73.59	74.97	67.68	76.89	76.89	71.53
	Word bigrams	AW	35.49	78.82	77.17	78.82	76.89	79.23
		NS	29.57	78.95	78.68	78.95	76.89	78.82
		OS	67.26	78.4	58.46	66.99	76.89	68.5
	Word trigrams	AW	28.2	78.68	78.54	78.82	76.89	78.82
		NS	30.4	78.68	78.82	78.82	76.89	78.95
		OS	46.22	78.4	77.44	75.93	76.89	76.48
	Character unigrams	AC	69.74	75.52	68.09	78.27	76.89	74.42
		SS	69.88	75.52	68.5	77.99	76.89	74.42
		PP	69.46	75.93	68.5	78.13	76.89	75.38
		SP	69.46	75.93	67.4	77.99	76.89	75.38
	Character bigrams	AC	75.24	76.48	71.11	78.82	76.89	70.29
		SS	75.1	75.24	69.19	78.13	76.89	71.11
		PP	74.42	77.17	70.84	78.95	76.89	73.18
		SP	74.83	75.93	69.88	78.4	76.89	72.21
	Character trigrams	AC	76.2	74.83	73.45	79.23	76.89	71.25
		SS	76.07	75.38	74.55	79.23	76.89	70.7
		PP	75.93	75.24	71.66	79.37	76.89	70.84
		SP	76.34	75.79	74.0	78.95	76.89	68.5
Tamil	Surface features		53.06	56.47	50.64	54.48	54.34	51.92
Binary (Presence/Absence)	SentiWordNet		55.05	55.05	55.05	55.05	55.05	55.05
	Word unigrams	AW	62.16	60.17	56.76	57.61	55.05	59.46
	Word bigrams	AW	49.36	56.9	56.19	56.76	55.05	55.76
	Word trigrams	AW	44.95	57.04	56.76	56.9	55.05	57.18
	Character unigrams	AC	57.89	58.04	49.22	57.18	57.47	58.46
		SS	57.89	58.04	50.64	57.61	57.61	58.46
		PP	57.04	55.48	50.36	55.76	57.33	56.47
		SP	56.76	55.76	51.64	57.04	57.18	56.61
	Character bigrams	AC	58.32	57.18	53.49	59.89	55.05	52.35
		SS	59.46	57.04	49.93	59.17	55.05	53.63
		PP	55.48	54.62	52.49	59.74	55.05	54.91
		SP	55.76	55.33	53.91	59.89	55.05	55.62
	Character trigrams	AC	56.9	55.05	54.05	60.31	55.05	52.2
		SS	58.32	57.75	55.76	59.89	55.05	56.47
		PP	56.05	53.34	53.77	59.6	55.05	52.06
		SP	56.33	56.05	55.62	57.04	55.05	54.34

Table 3: % accuracy of 2-class classification for three languages on 10-fold cross-validation on the training data. AW = all words, NS = all words except stop words, OS = only stop words. AC = all characters, SS = all characters except space, PP = all characters except punctuation, SP = all characters except space and punctuation. Rightmost six columns are classifiers, as indicated at the end of Section 3.

Feature Representation	Feature Category	Feature Type	NB	LR	DT	RF	SV	LS
Bengali	Surface features		34.43	42.64	34.93	38.14	39.44	36.44
Binary (Presence/Absence)	SentiWordNet		36.84	36.84	36.84	36.84	36.84	36.84
	Word unigrams	AW	47.65	51.25	47.75	48.05	36.84	50.05
		NS	46.95	49.85	44.14	49.35	36.84	50.15
		OS	37.74	39.54	39.84	39.84	37.74	39.24
	Word bigrams	AW	45.45	48.55	44.74	46.05	36.84	48.15
		NS	43.84	47.45	44.04	45.75	36.84	47.55
		OS	36.14	36.04	37.64	38.14	36.84	35.44
	Word trigrams	AW	42.34	46.35	43.84	44.54	36.84	46.35
		NS	42.04	45.95	42.74	44.14	36.84	45.35
		OS	35.44	36.84	36.04	36.04	36.84	36.24
	Character unigrams	AC	39.14	43.34	38.14	45.95	40.24	43.54
		SS	39.24	43.44	38.04	44.84	40.24	43.34
		PP	36.74	41.34	39.54	43.14	40.04	41.54
		SP	36.84	41.14	39.34	43.34	40.04	41.54
	Character bigrams	AC	37.24	39.24	40.04	43.44	36.84	40.54
		SS	38.04	38.74	37.14	44.14	36.84	38.44
		PP	37.24	39.44	39.34	44.14	36.84	39.94
		SP	36.54	39.84	38.84	44.34	36.84	37.44
	Character trigrams	AC	38.24	41.04	39.14	44.44	36.84	40.94
		SS	36.94	39.44	35.74	45.25	36.84	37.04
		PP	36.84	38.34	37.34	47.75	36.84	37.84
		SP	37.64	38.04	41.54	44.94	36.84	36.14
Hindi	Surface features		41.82	48.2	43.86	45.99	46.4	33.47
Binary (Presence/Absence)	SentiWordNet		45.74	45.74	45.74	45.74	45.74	45.74
	Word unigrams	AW	54.83	55.32	49.92	56.38	45.74	56.38
		NS	51.55	54.75	48.85	52.13	45.74	55.48
		OS	46.24	50.25	42.06	47.87	45.66	49.51
	Word bigrams	AW	25.2	52.45	48.28	47.95	45.74	52.86
		NS	21.19	46.07	43.45	47.71	45.74	46.64
		OS	44.68	50.49	46.15	48.12	45.74	47.3
	Word trigrams	AW	19.23	47.05	45.5	45.74	45.74	45.91
		NS	22.75	47.71	43.7	44.27	45.74	44.35
		OS	33.72	47.22	43.7	45.42	45.74	46.15
	Character unigrams	AC	45.25	50.98	42.39	51.31	49.59	50.82
		SS	45.17	50.98	43.13	52.45	49.59	50.82
		PP	45.25	49.26	43.37	50.41	49.75	48.61
		SP	45.01	49.18	41.0	51.47	49.84	48.61
	Character bigrams	AC	46.24	50.08	43.21	52.54	45.74	45.99
		SS	47.71	49.51	43.45	51.72	45.74	46.56
		PP	45.99	50.9	45.66	53.19	45.74	46.15
		SP	46.07	51.39	42.55	52.05	45.74	49.26
	Character trigrams	AC	48.04	47.38	45.17	53.03	45.74	45.5
		SS	47.38	47.05	43.86	52.29	45.74	46.64
		PP	48.2	48.2	42.31	53.11	45.74	45.42
		SP	46.89	49.02	44.84	51.8	45.74	46.15
Tamil	Surface features		39.08	43.52	34.27	37.17	39.17	36.9
Binary (Presence/Absence)	SentiWordNet		36.26	36.26	36.26	36.26	36.26	36.26
	Word unigrams	AW	40.71	39.53	39.26	42.07	36.26	38.8
	Word bigrams	AW	36.08	40.25	38.08	39.26	36.26	40.34
	Word trigrams	AW	30.92	37.99	37.35	37.99	36.26	38.17
	Character unigrams	AC	43.52	40.98	36.08	43.16	43.25	40.71
		SS	43.52	41.07	36.9	44.15	43.34	40.34
		PP	43.25	41.25	39.98	41.07	43.16	40.62
		SP	43.43	41.25	39.44	39.89	43.06	40.34
	Character bigrams	AC	42.25	41.34	39.89	43.61	36.26	37.81
		SS	41.7	41.34	37.81	44.24	36.26	39.44
		PP	39.17	39.08	39.98	42.52	36.26	38.89
		SP	39.44	40.07	37.81	41.61	36.26	40.16
	Character trigrams	AC	39.26	38.8	37.53	43.34	36.26	35.63
		SS	39.89	41.98	37.53	43.16	36.26	39.89
		PP	38.44	39.26	37.99	40.89	36.26	38.53
		SP	38.62	40.98	35.9	41.98	36.26	38.89

Table 4: % accuracy of 3-class classification for three languages on 10-fold cross-validation on the training data. AW = all words, NS = all words except stop words, OS = only stop words. AC = all characters, SS = all characters except space, PP = all characters except punctuation, SP = all characters except space and punctuation. Rightmost six columns are classifiers, as indicated at the end of Section 3.

(a) Bengali Negative example confused as Positive

(b) Bengali Positive example confused as Negative

(c) Hindi Neutral example confused as Negative

(d) Hindi Positive example confused as Neutral

Figure 3: Error cases for Bengali and Hindi.

however, they are qualitatively similar to the binary feature representation.[4] There are several observations to be made from Tables 3 and 4. The first observation is that overall, Hindi has the highest accuracy values, followed by Tamil, followed by Bengali. This indicates that sentiment classification in Bengali is the most difficult, followed by Tamil and Hindi. Later, we will perform error analysis to see which cases are the most difficult. Note further that we did not have access to a Tamil stop word list, hence the absence of "OS" and "NS" types for Tamil. Another interesting observation is that the accuracy values from 2-class classification are substantially higher – cell for cell – than those from 3-class classification. This shows that the 2-class classification task is substantially easier than the 3-class classification task.

Surface features performed very well in this task, which is a surprising finding. It shows that a handful of manually chosen features can go a long way when the features are inspired by the data. Feature ranking by importance showed that tweet length in words and characters, and the number of punctuation symbols were the most important features in this category. SentiWordNet features performed comparably to surface features; however, their performance was not affected by the classifier used or the feature representation (binary/tf/tfidf). We believe that the reason this happened is because SentiWordNets are highly language-specific, and any feature representation would perform equivalently good (or bad) depending on what language we are dealing with, not what classifiers we have at our disposal.

The best performance numbers came from word and character n-grams, thereby showing beyond doubt that simple, robust, scalable, and language-independent features outperform complex, fragile, cumbersome, and language-dependent features. The best-performing feature-classifier combinations are as follows:[5]

- **Bengali, 2-class:** Word unigrams, no stop words, tfidf, NB classifier (67.83%).
- **Bengali, 3-class:** Word unigrams, all words including stop words, binary, LR classifier (51.25%).
- **Hindi, 2-class:** Word unigrams, all words including stop words, binary, LR classifier (81.57%).
- **Hindi, 3-class:** Word unigrams, all words including stop words, tf, LR classifier (56.96%).
- **Tamil, 2-class:** Word unigrams, all words including stop words, binary, NB classifier (62.16%).
- **Tamil, 3-class:** Character unigrams, all characters, tf, RF classifier (45.24%).

Note that our best 3-class accuracy values are better than the best reported accuracy values in (Patra et al., 2015). We obtained 51.25% for Bengali compared to 43.2%, 56.96% for Hindi compared to 55.67%, and 45.24% for Tamil compared to 39.28% – in the constrained version of the task. With the best combinations, we went ahead and trained them on the *whole* training data, and tested the models on the *development data* made available for Hindi and Bengali. For Hindi, we used the 3-class model because the development data had 3 classes, whereas for Bengali we used the 2-class model, because Bengali development data did not have any samples from the "neutral" class. We obtained 94.64% accuracy on the Hindi development data (which widely beats the 55.67% reported in (Patra et al., 2015)), and 56.6%

[4]All results are available in the supplementary PDF at `http://web.eecs.umich.edu/~lahiri/WSSANLP_supplement.pdf`.

[5]For full results, please see the supplement at `http://web.eecs.umich.edu/~lahiri/WSSANLP_supplement.pdf`.

accuracy on the Bengali development data (which also handily beats the 43.2% reported by Patra et al. (2015)) – showing again the importance of simple, robust, scalable, and language-independent features.

One question that arises at this point, is: *which features are the most important* in these top-performing models? We ranked the features by their importance in the training data, and show them in Figure 1. Note that each ranking has at least one English segment – which shows that English words can be important in discriminating between sentiment classes. Note further that the Bengali words are more abstract, such as "freedom", "people", "wish", and "judges", with only one positive word – "joy" – in the end. Hindi words, on the other hand, are more direct: "auspicious", "development", "God", "help", "apology", "grace", "victory", "change", and "hearty". We believe that the reason this happened is the data collection process. The Bengali tweets that were collected reflect a more *general* view, whereas Hindi tweets reflect a more *personal* view.

Another question that arises, is: *how sensitive is the development accuracy on the size of the training data?* In other words, if we varied the training set size, how would the development accuracy change? To answer this question, we varied the number of training samples for Hindi and Bengali from 100 to the maximum – in steps of 100, trained the best-performing model on this *reduced* training set, and tested the resulting model on the development set. This gave us two *learning curves*, as shown in Figure 2. First, note that the model *overfits* beyond a certain number of training instances, and the development accuracy drops beyond this point. Second, we do not need all training instances to obtain optimal development accuracy. For Bengali, the optimum comes at 200 training instances (60.38% accuracy), whereas for Hindi, the optimum comes at 300 instances (96.43% accuracy).

The last question that we investigated, is: *what are the error cases that our best-performing models did not get right?* Do they have any specific properties that make them hard to classify? To answer this question, we looked into the cases our models misclassified on the *development set* for Hindi and Bengali. Hindi had only three cases misclassified out of 56, and Bengali had 23 cases misclassified out of 53. We show four examples in Figure 3 that have been misclassified with relatively high confidence. Among these cases, Figure 3d is a case where the classifier truly misclassified a positive example as a neutral one. However, Figure 3c's neutral-ness is debatable, because it is describing a somewhat negative and pessimistic aphorism on the transience of life. Similarly, the example shown in Figure 3b is not uniformly positive, because it starts to describe a set of financial impediments to the successful implementation of some of the policy recommendations by the United Nations. Also, Figure 3a's dominant sentiment is negative, but it begins to provide a sense of hope, faith, and enlightenment towards the end. These examples show that although our best classifiers are not perfect, they misclassified examples that are truly *hard* to classify, and in fact may even be hard to classify by a human being.

5 Conclusion

In this paper, we performed tweet sentiment analysis of three Indian languages – Bengali, Hindi, and Tamil. We experimented with a set of simple, robust, scalable, and language-independent features, and showed that they achieve performance superior to the state-of-the-art, and also superior to language-specific features. We performed detailed error analysis, and found out that in most cases, our models were performing well, and they only got confused when the sample was truly confusing – perhaps even to a human being. We performed feature importance ranking to identify words that were relevant to the task of sentiment classification in three different languages, and showed the variations thereof. We also showed how the development accuracy changed in response to the size of the training data. Our limitations include: not having access to the test data, and the stop word list for Tamil. However, our results demonstrably overcame these limitations. Future research should look into collecting more sentiment-annotated tweets to get a better handle on the underlying psychological phenomena of *opinion* and *subjectivity*, and using existing NLP tools in Bengali, Hindi, and Tamil to see how they perform in this very interesting, but also challenging task.

References

Balamurali A. R., 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, Mumbai, India, December. The COLING 2012 Organizing Committee.

Apoorv Agarwal, Boyi Xie, Ilia Vovsha, Owen Rambow, and Rebecca Passonneau. 2011. Sentiment Analysis of Twitter Data. In *Proceedings of the Workshop on Languages in Social Media*, LSM '11, pages 30–38, Stroudsburg, PA, USA. Association for Computational Linguistics.

Md Shad Akhtar, Asif Ekbal, and Pushpak Bhattacharyya. 2016. Aspect based Sentiment Analysis in Hindi: Resource Creation and Evaluation. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Helene Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2010. SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner, and Daniel Tapias, editors, *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may. European Language Resources Association (ELRA).

Amitava Das and Sivaji Bandyopadhyay. 2010. SentiWordNet for Indian Languages. In *Proceedings of the Eighth Workshop on Asian Language Resouces*, pages 56–63, Beijing, China, August. Coling 2010 Organizing Committee.

Ronen Feldman. 2013. Techniques and Applications for Sentiment Analysis. *Commun. ACM*, 56(4):82–89, April.

Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter Sentiment Classification using Distant Supervision. Technical report, Stanford University.

Efthymios Kouloumpis, Theresa Wilson, and Johanna Moore. 2011. Twitter Sentiment Analysis: The Good the Bad and the OMG!

Ayush Kumar, Sarah Kohail, Asif Ekbal, and Chris Biemann. 2015. IIT-TUDA: System for Sentiment Analysis in Indian Languages using Lexical Acquisition. In *Proceedings of the Third International Conference on Mining Intelligence and Knowledge Exploration - Volume 9468*, MIKE 2015, pages 684–693, New York, NY, USA. Springer-Verlag New York, Inc.

Bing Liu. 2015. *Sentiment Analysis: Mining Opinions, Sentiments, and Emotions*. Cambridge University Press.

Andrés Montoyo, Patricio Martínez-Barco, and Alexandra Balahur. 2012. Subjectivity and sentiment analysis: An overview of the current state of the area and envisaged developments. *Decision Support Systems*, 53(4):675 – 679. 1) Computational Approaches to Subjectivity and Sentiment Analysis 2) Service Science in Information Systems Research : Special Issue on {PACIS} 2010.

Alexander Pak and Patrick Paroubek. 2010. Twitter as a Corpus for Sentiment Analysis and Opinion Mining. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner, and Daniel Tapias, editors, *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may. European Language Resources Association (ELRA).

Pooja Pandey and Sharvari Govilkar. 2015. A Framework for Sentiment Analysis in Hindi using HSWN. *International Journal of Computer Applications*, 119(19):23–26, June.

Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1-2):1–135, January.

Braja Gopal Patra, Dipankar Das, Amitava Das, and Rajendra Prasath, 2015. *Shared Task on Sentiment Analysis in Indian Languages (SAIL) Tweets - An Overview*, pages 650–655. Springer International Publishing, Cham.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Kamal Sarkar and Saikat Chakraborty, 2015. *A Sentiment Analysis System for Indian Language Tweets*, pages 694–702. Springer International Publishing, Cham.

Richa Sharma, Shweta Nigam, and Rekha Jain. 2014. Opinion Mining In Hindi Language: A Survey. *CoRR*, abs/1404.4935.

Hao Wang, Dogan Can, Abe Kazemzadeh, François Bar, and Shrikanth Narayanan. 2012. A System for Real-time Twitter Sentiment Analysis of 2012 U.S. Presidential Election Cycle. In *Proceedings of the ACL 2012 System Demonstrations*, pages 115–120, Jeju Island, Korea, July. Association for Computational Linguistics.

Lei Zhang, Riddhiman Ghosh, Mohamed Dekhil, Meichun Hsu, and Bing Liu, 2011. *Combining Lexicon-based and Learning-based Methods for Twitter Sentiment Analysis*. 89 edition, June.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level Convolutional Networks for Text Classification. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 649–657. Curran Associates, Inc.