KanCMD: Kannada CodeMixed Dataset for Sentiment Analysis and Offensive Language Detection

Adeep Hande[1], **Ruba Priyadharshini**[2], **Bharathi Raja Chakravarthi**[3]
[1] Indian Institute of Information Technology Tiruchirappalli, Tamil Nadu, India
[2] ULTRA Arts and Science College, Madurai, Tamil Nadu, India
[3]Insight SFI Research Centre for Data Analytics, National University of Ireland Galway
adeeph18c@iiitt.ac.in, rubapriyadharshini.a@gmail.com,
bharathi.raja@insight-centre.org

Abstract

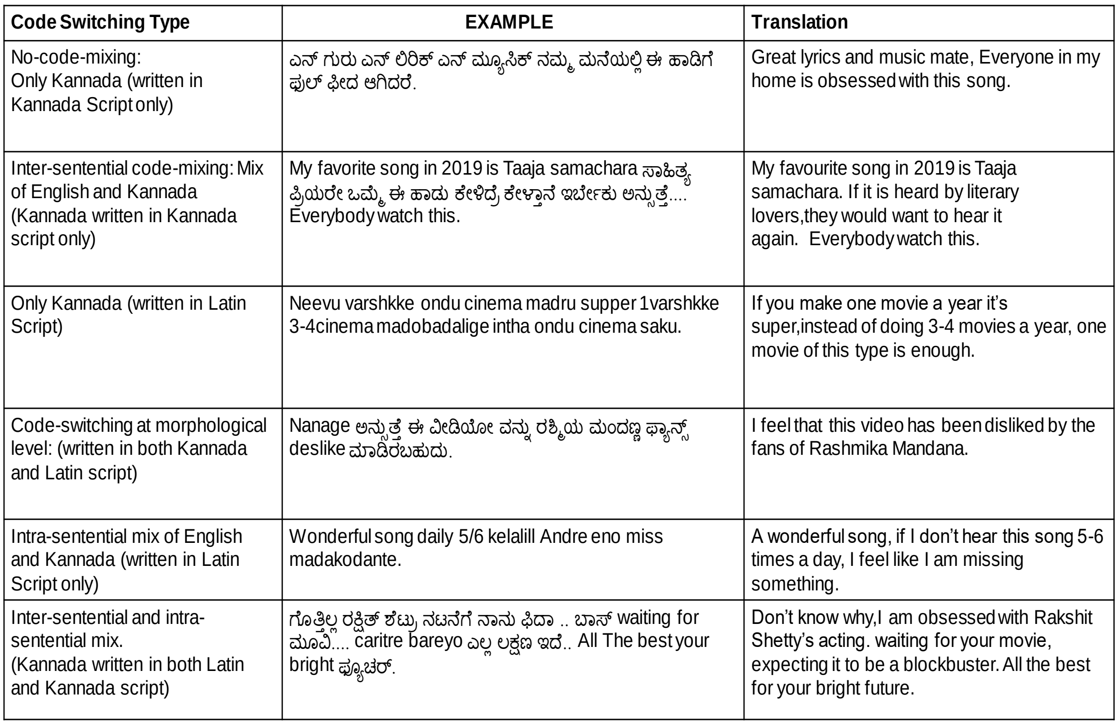
We introduce Kannada CodeMixed Dataset (KanCMD), a multi-task learning dataset for sentiment analysis and offensive language identification. The KanCMD dataset highlights two real-world issues from the social media text. First, it contains actual comments in code mixed text posted by users on YouTube social media, rather than in monolingual text from the textbook. Second, it has been annotated for two tasks, namely sentiment analysis and offensive language detection for under-resourced Kannada language. Hence, KanCMD is meant to stimulate research in under-resourced Kannada language on real-world code-mixed social media text and multi-task learning. KanCMD was obtained by crawling the YouTube, and a minimum of three annotators annotates each comment. We release KanCMD 7,671 comments for multitask learning research purpose.

1 Introduction

A surge in the active users on social media has given rise to people's engagement in expressing their opinions in the form of comments and reviews on social media platforms such as Facebook, YouTube and Twitter (Severyn et al., 2014; Clarke and Grieve, 2017; Tian et al., 2017). We see a lot of informal texts which do not follow any grammatical rules, sometimes code-mixed and even written in non-native scripts (Bali et al., 2014; Jose et al., 2020; Chakravarthi et al., 2020a). These texts can be offensive and may be directed towards an individual or community to show their dissent. Thus offensive language identification is essential for social media platforms to minimise these activities (Bohra et al., 2018). Sentiment analysis is used to interpret and classify emotions from text data after analysing it with various existing techniques (Pang and Lee, 2008). It has received a lot of interest in industry and research for identifying customer satisfaction on products and services, but there is no dataset available for code-mixed Kannada language. Code-mixing refers to the pairing of linguistic units from two or more languages into a single conversation (Pratapa et al., 2018; Priyadharshini et al., 2020). Code-mixed sentiment analysis and offensive language identification when succeed trained on code-mixed data while regular monolingual sentiment analysis might prove ineffective due to the large variations in the text (Banerjee et al., 2020).

Kannada language (ISO 639-3:kan) belong to Dravidian language family (Chakravarthi et al., 2019b; Chakravarthi, 2020), spoken predominantly by the people of Karnataka State in south India. Kannada language uses Kannada script also called Caranese which is derived from Bhattiprolu Brahmi script (Chakravarthi et al., 2020c). It is a phonemic adugida written from left to right (Chakravarthi et al., 2019a). We observed six combinations among code-mixed sentences such as no-code-mixing only Kannada written in Kannada script or Kannada written in the Latin script, inter-sentential code-mixing, code-switching at morphological level, intra-sentential code-mixing, inter-sentential and intra-sentential mix. Most comments were written in Kannada script either with Kannada grammar with English lexicon or English grammar with Kannada lexicon. Some comments were written in Kannada script with English expressions in between. Figure 1 illustrate the different level of code-mixing in our dataset.

Proceedings of the Third Workshop on Computational Modeling of PEople's Opinions, PersonaLity, and Emotions in Social media, pages 54–63
Barcelona, Spain (Online), December 13, 2020.



Code Switching Type	EXAMPLE	Translation
No-code-mixing: Only Kannada (written in Kannada Script only)	ಎನ್ ಗುರು ಎನ್ ಲಿರಿಕ್ ಎನ್ ಮ್ಯೂಸಿಕ್ ನಮ್ಮ ಮನೆಯಲ್ಲಿ ಈ ಹಾಡಿಗೆ ಫುಲ್ ಫೀದ ಆಗಿದರೆ.	Great lyrics and music mate, Everyone in my home is obsessed with this song.
Inter-sentential code-mixing: Mix of English and Kannada (Kannada written in Kannada script only)	My favorite song in 2019 is Taaja samachara ಸಾಹಿತ್ಯ ಪ್ರಿಯರೇ ಒಮ್ಮೆ ಈ ಹಾಡು ಕೇಳಿದ್ರೆ ಕೇಳ್ತಾನೆ ಇರ್ಬೇಕು ಅನ್ಸುತ್ತೆ.... Everybody watch this.	My favourite song in 2019 is Taaja samachara. If it is heard by literary lovers,they would want to hear it again. Everybody watch this.
Only Kannada (written in Latin Script)	Neevu varshkke ondu cinema madru supper 1varshkke 3-4cinema madobadalige intha ondu cinema saku.	If you make one movie a year it's super,instead of doing 3-4 movies a year, one movie of this type is enough.
Code-switching at morphological level: (written in both Kannada and Latin script)	Nanage ಅನ್ಸುತ್ತೆ ಈ ವೀಡಿಯೋ ವನ್ನು ರಶ್ಮಿಯ ಮಂದಣ್ಣ ಫ್ಯಾನ್ಸ್ deslike ಮಾಡಿರಬಹುದು.	I feel that this video has been disliked by the fans of Rashmika Mandana.
Intra-sentential mix of English and Kannada (written in Latin Script only)	Wonderful song daily 5/6 kelalill Andre eno miss madakodante.	A wonderful song, if I don't hear this song 5-6 times a day, I feel like I am missing something.
Inter-sentential and intra-sentential mix. (Kannada written in both Latin and Kannada script)	ಗೊತ್ತಿಲ್ಲ ರಕ್ಷಿತ್ ಶೆಟ್ರು ನಟನೆಗೆ ನಾನು ಫಿದಾ .. ಬಾಸ್ waiting for ಮೂವಿ.... caritre bareyo ಎಲ್ಲ ಲಕ್ಷಣ ಇದೆ.. All The best your bright ಫ್ಯೂಚರ್.	Don't know why,I am obsessed with Rakshit Shetty's acting. waiting for your movie, expecting it to be a blockbuster. All the best for your bright future.

Figure 1: Examples of code mixing in our dataset.

In multitask learning, the objective is to utilize the process of learning multiple tasks in order to improve the performance of the system (Martínez Alonso and Plank, 2017). Sentiment analysis and offensive language identification are related and has common aspects between them. Having the model to learn both tasks would be advantages to utilise some cues from one task to improve the other. Since Kannada is morphologically rich and under-resourced language (Prabhu et al., 2020) to improve the performance of the classification system, we annotate dataset for multitask learning. To detect customer satisfaction and eliminate offensive language in these platforms, we release KanCMD, a dataset of YouTube video comments in code-mixed Kannada-English.

2 Related Work

Sentiment analysis has become one of the primary areas of research with applications across many trades and industries(such as finance, online marketing, political science) (Severyn et al., 2014). Over the last 20 years, social media networks have contributed immensely to the availability of rich data sources for analysis of sentiment (Clarke and Grieve, 2017; Tian et al., 2017). This combined with efforts directed towards the compilation of sentiment lexicons (Turney, 2002; Lal et al., 2019) have resulted in this branch of natural language processing maturing out. In the early years of research, n-grams were used for classification of sentiments carried by the datasets. Recently, these methods have been replaced by neural model architectures. However, sentiment analysis in Kannada (Hegde and Padma, 2015; Kumar et al., 2015) has not achieved this.

Aggression identification in social media (Kumar et al., 2018) and offensive language identification (Zampieri et al., 2019) shared task has been conducted to improve the research in this area. Offensive language identification dataset was released for Greek (Pitenis et al., 2020). However, offensive language identification has not been made for the Kannada language. For language identification (LID) systems in code-mixed Languages, a Kannada-English dataset containing English, Kannada and several word-level code-mixed words was created by Sowmya Lakshmi and Shambhavi (2017). A stance detection system was employed to detect stance in Kannada social media code-mixed text using sentence embeddings. Machine learning models such as logistic regression and a distributed memory model for sentence vectors were among the models to experiment (Skanda et al., 2017). Distributed representations of texts through neural networks method has experimented for sentiment analysis on Kannada-English code-mixed dataset, which had three tags, Positive, Negative and Neutral (Shalini et al., 2018). However, the dataset for Kannada was not easily available for research purpose. Following (Chakravarthi et al., 2020a), we downloaded the YouTube comments for Kannada and annotated. In our research, we release code-mixed dataset for under-resourced Kannada for sentiment analysis and offensive language identification as multi-task learning dataset.

3 Dataset Construction

We create a dataset for two tasks, namely, sentiment analysis and offensive language identification. We collected comments from YouTube using YouTube Comment Scrapper [1]. We collected comments from 18 videos on different topics ranging from movie trailers, current trends about the ban on mobile apps in India, India-China border issue, Mahabharata, and Transgenders. We used these keywords to find the video and then from the videos we collected the comments. This was collected between Feburary, 2020 and August, 2020.

3.1 Sentiment Analysis

For sentiment analysis, we adopted the approach taken by Chakravarthi et al. (2020b), and a minimum of three annotators annotated each sentence according to the following schema:

- **Positive state:** Comment contains an explicit or implicit clue in the text suggesting that the speaker is in a positive state.
- **Negative state:** Comment contains an explicit or implicit clue in the text suggesting that the speaker is in a negative state.
- **Mixed feelings:** Comment contains an explicit or implicit clue in both positive and negative feeling.
- **Neutral state:** Comment does not contain an explicit or implicit indicator of the speaker's emotional state.
- **Not in intended language:** For Kannada if the sentence does not contain Kannada written in Kannada script or Latin script then it is not Kannada.

3.2 Offensive Language Identification

We constructed offensive language identification for the Kannada language at different levels of complexity following Zampieri et al. (2019) work. More generally it expands to three-level hierarchical annotation schema. To simplify, we have split it into six labels.

- **Not Offensive**: Comments does not contain offence or profanity.
- **Offensive Untargeted**: Comments contain offence or profanity without any target. These are comments which contain unacceptable languages that do not target anyone.
- **Offensive Targeted Individual**: Comments contains offence or profanity which targets the individual.
- **Offensive Targeted Group**: Comments contains offence or profanity which targets the group.
- **Offensive Targeted Other**: Comments contains offence or profanity which does not belong to any of the previous two categories(e.g., a situation, an issue, an organization or an event).
- **Not in indented language**: Comments not in the Kannada language.

[1] https://github.com/philbot9/youtube-comment-scraper-cli

Gender	Male	2
	Female	3
Higher Education	Undegraduate	1
	Graduate	2
	Postgraduate	2
Medium of Schooling	English	4
	Kannada	1
Total		5

Table 1: Annotators

Language pair	Kannada-English
Number of Tokens	64,997
Vocabulary Size	20,667
Number of Posts	7,671
Number of Sentences	8,472
Average number of Tokens per post	8
Average number of sentences per post	1

Table 2: Dataset statistics

3.3 Annotators

We created Google forms to collect annotations from annotators. Gender, education background, medium of schooling was collected to know the diversity of the annotator. The annotators were warned that comments might have offensive language and abusive text. The annotator was given a choice to stop annotation if they find it disturbing or could not handle. Annotators were asked not to be biased to a particular person, situation or event during the annotation of comments. Each form was annotated by a minimum of three annotators and maximum of 5 annotators. From the Table 1, we can see that majority of the annotators' medium of schooling is English even though their mother tongue is Kannada and they were from Karnataka state in India where Kannada is the official language of the state. Krippendorff's alpha for sentiment analysis annotation was 0.73, and offensive language identification was 0.78.

Class	Kannada-English
Positive	3,518
Negative	1,484
Mixed feelings	691
Neutral	842
Other language	1,136
Total	7,671

Table 3: Sentiment Analysis Dataset Distribution

4 Data Statistics and Analysis

After performing annotations using google forms for both of the tasks, sentiment analysis and offensive language detection, all of the responses were converted into .csv format and then combined to a single dataset containing all the annotations. Our goal is to analyse the multitask dataset and perform experiments with several machine learning algorithms to establish benchmark results.

Table 3 and Table 4 shows the dataset statistics of the Kannada-English code-mixed dataset. As shown on the table, this huge dataset has 64,997 tokens, where the vocabulary size is 20,667. There are 7,671 comments and 8,472 distinct sentences in our code-mixed dataset. On average, there are eight tokens

Class	Kannada-English
Not Offensive	4,336
Offensive Untargeted	278
Offensive Targeted Individual	626
Offensive Targeted Group	416
Offensive Targeted Others	152
Other language	1,863
Total	7,671

Table 4: Offensive language Identification Dataset Distribution.

per sentence, and there is at least one sentence per post. As described earlier, the whole dataset was categorised into two tasks, sentiment analysis and offensive language detection.

The first task performed on the dataset was sentiment analysis. It was categorised into five groups, such as positive, negative, neutral, mixed-feelings, other languages. As mentioned in table 3, the distribution is as follows. Out of 7,671 posts, 3,518 have a positive polarity, being the most frequent category in this task of sentiment analysis. The second-largest category was a negative state, accounting to 1,484 comments of the whole code-mixed dataset. The absence of a speaker's emotional state relating to the subject in a post was considered as a neutral state. We split the dataset retaining ten percentage of the dataset, that is, 768 for the test, ten percentage for validation, being 767 and the remaining for training.

The second task performed on the dataset was offensive language detection. It was categorised into six groups such as not offensive, offensive untargeted, offensive targeted individual, offensive targeted group, offensive targeted other, other languages. Since the same code-mixed dataset was used for this task, the dataset statistics would be the same here. Out of 7,671 posts, 4338 of them were not considered to be offensive, which was the most frequent category in this task. We similarly split the dataset to what was done for the task of sentiment analysis.

5 Benchmark Systems

In order to provide a simple baseline, we applied several traditional machine learning algorithms such as Logistic Regression (LR), Support Vector Machine (SVM), Multinomial Naive Bayes (MNB), K-Nearest Neigbours (KNN), Decision Trees (DT), Random Forest (RF) separately, for both of the tasks, sentiment analysis and offensive language detection on KanCMD, the code-mixed Kannada-English dataset.

5.1 Experiments Setup

5.1.1 Logistic Regression (LR):

We evaluate the Logistic Regression model with L2 regularization. The input features are the Term Frequency Inverse Document Frequency (TF-IDF) values of up to 3 grams. This approach results in the model being trained only on this dataset without taking any pre-trained embeddings.

5.1.2 Support Vector Machine (SVM):

We evaluate the SVM model with L2 regularization. The features are the same as in LR. The main objective of SVM classifier is to find a hyperplane in an N-dimensional space that distinctly classifies the data points.

5.1.3 Multinomial Naive Bayes (MNB):

We evaluate a Naive Bayes classifier for multinomially distributed data, which is derived from Bayes Theorem that finds the probability of a future event to the given occurred event. Laplace smoothing is performed using $\alpha = 1$ to solve the problem of zero probability and then evaluate the MNB model with TF-IDF vectors.

5.1.4 K-Nearest Neighbour (KNN):

We use KNN for classification with 3,4,5, and 9 neighbours by applying uniform weights.

Decision Tree (DT)									
Class	Precision	Recall	F1-score	Support	Class	Precision	Recall	F1-score	Support
Positive	0.59	0.73	0.66	363	NO	0.64	0.78	0.70	417
Negative	0.61	0.48	0.54	162	OU	0.21	0.09	0.13	33
Mixed	0.21	0.19	0.20	57	OTI	0.57	0.51	0.54	75
Neutral	0.39	0.14	0.21	83	OTG	0.29	0.18	0.22	44
Other	0.45	0.47	0.46	103	OTO	0.25	0.07	0.11	14
					OL	0.56	0.45	0.50	185
accuracy			0.54	768	accuracy			0.60	768
M-Avg	0.45	0.40	0.41	768	M-Avg	0.42	0.35	0.37	768
W-Avg	0.53	0.54	0.52	768	W-Avg	0.57	0.60	0.58	768
Random Forest (RF)									
Class	Precision	Recall	F1-score	Support	Class	Precision	Recall	F1-score	Support
Positive	0.59	0.87	0.70	363	NO	0.65	0.89	0.75	417
Negative	0.70	0.48	0.57	162	OU	0.00	0.00	0.00	33
Mixed	0.45	0.06	0.11	57	OTI	0.71	0.35	0.47	75
Neutral	0.48	0.18	0.27	83	OTG	0.43	0.08	0.14	44
Other	0.53	0.50	0.52	103	OTO	1.00	0.06	0.11	14
					OL	0.67	0.54	0.60	185
accuracy			0.59	768	accuracy			0.66	768
M-Avg	0.55	0.42	0.43	768	M-Avg	0.58	0.32	0.34	768
W-Avg	0.58	0.59	0.55	768	W-Avg	0.63	0.66	0.61	768
Logistic Regression (LR)									
Sentiment Analysis					**Offensive Language Detection**				
Class	Precision	Recall	F1-score	Support	Class	Precision	Recall	F1-score	Support
Positive	0.70	0.69	0.70	363	NO	0.77	0.76	0.77	417
Negative	0.60	0.51	0.55	162	OU	0.04	0.03	0.04	33
Mixed	0.24	0.26	0.25	57	OTI	0.63	0.59	0.61	75
Neutral	0.38	0.36	0.37	83	OTG	0.25	0.23	0.24	44
Other	0.45	0.55	0.50	103	OTO	0.22	0.29	0.25	14
					OL	0.64	0.71	0.68	185
accuracy			0.57	768	accuracy			0.66	768
M-Avg	0.47	0.48	0.47	768	M-Avg	0.43	0.43	0.43	768
W-Avg	0.58	0.57	0.57	768	W-Avg	0.66	0.66	0.66	768

Table 5: Tasks: Sentiment Analysis and Offensive language detection.Precision,Recall,F1-score and support for DT and RF. Class : NO(Not Offensive), OU(Offensive Untargeted), OTI(Offensive Targeted Individual), OTG(Offensive Targeted Group), OTO(Offensive Targeted Others), OL(Other Language), M-Avg (Macro Average), W-Avg (Weighted Average)

5.1.5 Decision Tree (DT):

Decision tree classification works by generating a tree structure, where each node corresponds to a feature name, and the branches correspond to the feature values. The leaves of the tree represent the classification labels. After sequentially choosing alternative decisions, each node is recursively split again, and finally, the classifier defines some rules to predict the result. We used it to classify for the two tasks as shown Table 5 for baseline. Maximum depth was 800, and minimum sample splits were 5 for DT. The criteria were Gini and entropy.

5.1.6 Random Forest (RF):

Random Forest is an ensemble classifier that makes its prediction based on the combination of different decision trees. We evaluate the RF model with the same features as DT.

Support Vector Machine (SVM)									
Class	Precision	Recall	F1-score	Support	Class	Precision	Recall	F1-score	Support
Positive	0.47	1.00	0.64	363	NO	0.55	1.00	0.71	417
Negative	0.00	0.00	0.00	162	OU	0.00	0.00	0.00	33
Mixed	0.00	0.00	0.00	57	OTI	0.00	0.00	0.00	75
Neutral	0.00	0.00	0.00	83	OTG	0.00	0.00	0.00	44
Other	0.00	0.00	0.00	103	OTO	0.00	0.00	0.00	14
					OL	0.00	0.00	0.00	185
accuracy			0.47	768	accuracy			0.55	768
M-Avg	0.09	0.20	0.13	768	M-Avg	0.09	0.17	0.12	768
W-Avg	0.22	0.47	0.30	768	W-Avg	0.30	0.55	0.39	768
Multinomial Naive Bayes (MNB)									
Class	Precision	Recall	F1-score	Support	Class	Precision	Recall	F1-score	Support
Positive	0.54	0.99	0.70	363	NO	0.60	0.98	0.74	417
Negative	0.82	0.36	0.50	162	OU	0.00	0.00	0.00	33
Mixed	1.00	0.02	0.03	57	OTI	0.86	0.33	0.48	75
Neutral	0.75	0.04	0.07	83	OTG	0.00	0.00	0.00	44
Other	0.74	0.14	0.23	103	OTO	0.00	0.00	0.00	14
					OL	0.78	0.22	0.34	185
accuracy			0.57	768	accuracy			0.62	768
M-Avg	0.77	0.31	0.31	768	M-Avg	0.37	0.26	0.26	768
W-Avg	0.68	0.57	0.48	768	W-Avg	0.60	0.62	0.54	768
K-Nearest Neighbour (KNN)									
Class	Precision	Recall	F1-score	Support	Class	Precision	Recall	F1-score	Support
Positive	0.51	0.91	0.65	363	NO	0.61	0.93	0.73	417
Negative	0.67	0.10	0.17	162	OU	0.00	0.00	0.00	33
Mixed	0.44	0.07	0.12	57	OTI	0.78	0.19	0.30	75
Neutral	0.50	0.05	0.09	83	OTG	0.67	0.09	0.16	44
Other	0.55	0.41	0.47	103	OTO	0.00	0.00	0.00	14
					OL	0.66	0.34	0.45	185
accuracy			0.52	768	accuracy			0.61	768
M-Avg	0.53	0.31	0.30	768	M-Avg	0.45	0.26	0.27	768
W-Avg	0.54	0.52	0.43	768	W-Avg	0.60	0.61	0.55	768

Table 6: Tasks: Sentiment Analysis and Offensive language detection.Precision,Recall,F1-score and support for LR, SVM, MNB and KNN. Class : NO(Not Offensive), OU(Offensive Untargeted), OTI(Offensive Targeted Individual), OTG(Offensive Targeted Group), OTO(Offensive Targeted Others), OL(Other Language). M-Avg (Macro Average), W-Avg (Weighted Average)

5.2 Experiment Results

The results of the experiments performed for both of the tasks of sentiment analysis and Offensive language detection using different methods are shown in terms of Precision, Recall, F1-score and support in Table 6 and Table 5. We used sklearn[2] to develop the models. A macro-average will compute the metrics (precision, recall, F1-score) independently for each of the classes and then take the average. Thus this metric treats all classes equally, and it does not take the attribute of class imbalance into account. A weighted average takes the metrics from each class just like a macro average, but the contribution of each class to the average is weighted by the number of examples available for it. The value counts of different classes for both the tasks areas listed in support in Table 6 and Table 5.

For sentiment analysis, all the classification algorithms perform inadequately to average on the code-

[2] https://scikit-learn.org/stable/

mixed dataset. Logistic regression, random forest classifiers and decision trees were the ones that fared comparatively better across all sentiment classes. To our surprise, we see that SVM performs very bad, having a bad heterogeneity than the other methods. The precision, recall and F1-score are higher for the "Positive" class followed by the "Negative" class. All of the other classes performed very poorly. One of the reasons being the nature of the dataset as the classes "Mixed feelings" and "Neutral" are challenging to annotate for the annotators due to several factors behind its reasoning.

For offensive language detection, all the classification algorithms perform poorly. We see that logistic regression and random forest are the ones that performed relatively better than the others. The precision, recall and F1-score are higher for the "Not Offensive" class followed by the "Offensive Targeted Individual" and "OL" classes. The reasons for the poor performance of other classes are as same as sentiment analysis. From Table 6 and Table 5, we see that the classification algorithms have performed better on the task of sentiment analysis in comparison to their performance on the task of offensive language detection. One of the main reasons could be the differences in the distributions of the classes among the two different tasks. Out of the total of 7,671 sentences, 46% and 19 % belong to the "Positive" and the "Negative" classes respectively while the other classes share 9%,11% and 15% respectively for sentiment analysis. This distribution is relatively better when compared to offensive language detection task where 56% belong to "Not Offensive", while the other class share a low distribution of 4%,8%,6%,2%,24%.

6 Conclusion

In this paper, we presented KanCMD, a multi-task learning dataset for sentiment analysis and offensive language identification in under-resourced Kannada language. The dataset consists of 7,671 YouTube comments annotated by a minimum of three annotators and had 0.73 for sentiment analysis annotation, and 0.78 for offensive language identification in terms of Kripendorffs alpha inter-annotator agreement. We believe this dataset will allow future work in under-resourced Kannada language to progress in multi-task learning of the code-mixed real-world data. We have created computational models to set the benchmark for this dataset. We aim to promote research in the Kannada language and to encourage future investigations into multi-task learning for under-resourced languages in general and how it can be used to improve performance for under-resourced languages.

7 Acknowledgments

The author Bharathi Raja Chakravarthi was supported in part by a research grant from Science Foundation Ireland (SFI) under Grant Number SFI/12/RC/2289_P2 (Insight_2), co-funded by the European Regional Development Fund as well as by the EU H2020 programme under grant agreement 825182 (Prêt-à-LLOD), and Irish Research Council grant IRCLA/2017/129 (CARDAMOM-Comparative Deep Models of Language for Minority and Historical Languages) for his postdoctoral period at National University of Ireland Galway.

References

Kalika Bali, Jatin Sharma, Monojit Choudhury, and Yogarshi Vyas. 2014. "I am borrowing ya mixing ?" an analysis of English-Hindi code mixing in Facebook. In *Proceedings of the First Workshop on Computational Approaches to Code Switching*, pages 116–126, Doha, Qatar, October. Association for Computational Linguistics.

Shubhanker Banerjee, Bharathi Raja Chakravarthi, and John Philip McCrae. 2020. Comparison of pretrained embeddings to identify hate speech in Indian code-mixed text. In *2nd IEEE International Conference on Advances in Computing, Communication Control and Networking –ICACCCN (ICAC3N-20)*.

Aditya Bohra, Deepanshu Vijay, Vinay Singh, Syed Sarfaraz Akhtar, and Manish Shrivastava. 2018. A dataset of Hindi-English code-mixed social media text for hate speech detection. In *Proceedings of the Second Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media*, pages 36–41, New Orleans, Louisiana, USA, June. Association for Computational Linguistics.

Bharathi Raja Chakravarthi, Mihael Arcan, and John P. McCrae. 2019a. Comparison of Different Orthographies for Machine Translation of Under-Resourced Dravidian Languages. In Maria Eskevich, Gerard de Melo, Christian Fäth, John P. McCrae, Paul Buitelaar, Christian Chiarcos, Bettina Klimek, and Milan Dojchinovski, editors, *2nd Conference on Language, Data and Knowledge (LDK 2019)*, volume 70 of *OpenAccess Series in Informatics (OASIcs)*, pages 6:1–6:14, Dagstuhl, Germany. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

Bharathi Raja Chakravarthi, Mihael Arcan, and John P. McCrae. 2019b. WordNet gloss translation for under-resourced languages using multilingual neural machine translation. In *Proceedings of the Second Workshop on Multilingualism at the Intersection of Knowledge Bases and Machine Translation*, pages 1–7, Dublin, Ireland, 19 August. European Association for Machine Translation.

Bharathi Raja Chakravarthi, Navya Jose, Shardul Suryawanshi, Elizabeth Sherly, and John Philip McCrae. 2020a. A sentiment analysis dataset for code-mixed Malayalam-English. In *Proceedings of the 1st Joint Workshop on Spoken Language Technologies for Under-resourced languages (SLTU) and Collaboration and Computing for Under-Resourced Languages (CCURL)*, pages 177–184, Marseille, France, May. European Language Resources association.

Bharathi Raja Chakravarthi, Vigneshwaran Muralidaran, Ruba Priyadharshini, and John Philip McCrae. 2020b. Corpus creation for sentiment analysis in code-mixed Tamil-English text. In *Proceedings of the 1st Joint Workshop on Spoken Language Technologies for Under-resourced languages (SLTU) and Collaboration and Computing for Under-Resourced Languages (CCURL)*, pages 202–210, Marseille, France, May. European Language Resources association.

Bharathi Raja Chakravarthi, Navaneethan Rajasekaran, Mihael Arcan, Kevin McGuinness, Noel E.O'Connor, and John P McCrae. 2020c. Bilingual lexicon induction across orthographically-distinct under-resourced Dravidian languages. In *Proceedings of the Seventh Workshop on NLP for Similar Languages, Varieties and Dialects*, Barcelona, Spain, December.

Bharathi Raja Chakravarthi. 2020. *Leveraging orthographic information to improve machine translation of under-resourced languages*. Ph.D. thesis, NUI Galway.

Isobelle Clarke and Jack Grieve. 2017. Dimensions of abusive language on twitter. In *Proceedings of the First Workshop on Abusive Language Online*, pages 1–10, Vancouver, BC, Canada, August. Association for Computational Linguistics.

Y. Hegde and S. K. Padma. 2015. Sentiment analysis for ”kannada” using mobile product reviews: A case study. In *2015 IEEE International Advance Computing Conference (IACC)*, pages 822–827.

Navya Jose, Bharathi Raja Chakravarthi, Shardul Suryawanshi, Elizabeth Sherly, and John P. McCrae. 2020. A survey of current datasets for code-switching research. In *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)*.

K. M. Anil Kumar, N. Rajasimha, Manovikas Reddy, A. Rajanarayana, and Kewal Nadgir. 2015. Analysis of users' sentiments from kannada web documents. *Procedia Computer Science*, 54:247 – 256. Eleventh International Conference on Communication Networks, ICCN 2015, August 21-23, 2015, Bangalore, India Eleventh International Conference on Data Mining and Warehousing, ICDMW 2015, August 21-23, 2015, Bangalore, India Eleventh International Conference on Image and Signal Processing, ICISP 2015, August 21-23, 2015, Bangalore, India.

Ritesh Kumar, Atul Kr. Ojha, Shervin Malmasi, and Marcos Zampieri. 2018. Benchmarking aggression identification in social media. In *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018)*, pages 1–11, Santa Fe, New Mexico, USA, August. Association for Computational Linguistics.

Yash Kumar Lal, Vaibhav Kumar, Mrinal Dhar, Manish Shrivastava, and Philipp Koehn. 2019. De-mixing sentiment from code-mixed text. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 371–377, Florence, Italy, July. Association for Computational Linguistics.

Héctor Martínez Alonso and Barbara Plank. 2017. When is multitask learning effective? semantic sequence prediction under varying data conditions. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 44–53, Valencia, Spain, April. Association for Computational Linguistics.

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

Zesis Pitenis, Marcos Zampieri, and Tharindu Ranasinghe. 2020. Offensive language identification in Greek. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 5113–5119, Marseille, France, May. European Language Resources Association.

Suhan Prabhu, Ujwal Narayan, Alok Debnath, Sumukh S, and Manish Shrivastava. 2020. Detection and annotation of events in Kannada. In *16th Joint ACL - ISO Workshop on Interoperable Semantic Annotation PROCEEDINGS*, pages 88–93, Marseille, May. European Language Resources Association.

Adithya Pratapa, Gayatri Bhat, Monojit Choudhury, Sunayana Sitaram, Sandipan Dandapat, and Kalika Bali. 2018. Language modeling for code-mixing: The role of linguistic theory based synthetic data. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1543–1553, Melbourne, Australia, July. Association for Computational Linguistics.

Ruba Priyadharshini, Bharathi Raja Chakravarthi, Mani Vegupatti, and John P. McCrae. 2020. Named entity recognition for code-mixed Indian corpus using meta embedding. In *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)*.

Aliaksei Severyn, Alessandro Moschitti, Olga Uryupina, Barbara Plank, and Katja Filippova. 2014. Opinion mining on YouTube. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1252–1261, Baltimore, Maryland, June. Association for Computational Linguistics.

K. Shalini, H. B. Ganesh, M. A. Kumar, and K. P. Soman. 2018. Sentiment analysis for code-mixed indian social media text with distributed representation. In *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pages 1126–1131.

V. S. Skanda, M. A. Kumar, and K. P. Soman. 2017. Detecting stance in kannada social media code-mixed text using sentence embedding. In *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pages 964–969.

B. S. Sowmya Lakshmi and B. R. Shambhavi. 2017. An automatic language identification system for code-mixed english-kannada social media text. In *2017 2nd International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS)*, pages 1–5, Dec.

Ye Tian, Thiago Galery, Giulio Dulcinati, Emilia Molimpakis, and Chao Sun. 2017. Facebook sentiment: Reactions and emojis. In *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media*, pages 11–16, Valencia, Spain, April. Association for Computational Linguistics.

Peter Turney. 2002. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 417–424, Philadelphia, Pennsylvania, USA, July. Association for Computational Linguistics.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019. Predicting the type and target of offensive posts in social media. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1415–1420, Minneapolis, Minnesota, June. Association for Computational Linguistics.