Identifying and Categorizing Offensive Language in Social Media

IIIT - DELHI

Ankit Agarwal
M. Tech CSE
MT19021
ankit19021@iiitd.ac.in

Rahul Maheshwari
M. Tech CSE
MT19027
rahul19027@iiitd.ac.in

Diksha Solanki
M. Tech CSE
MT19078
diksha19078@iiitd.ac.in

Abstract

This document contains the report of the project "Identifying and Categorizing Offensive language in Social Media" which was a SemEval 2019 Task 6. There are 3 subtasks under the project which helps to categorize the category of offensive language expressed. For the task, we have performed various classification approaches to find best possible accuracy. Detailed explanation about the project is stated under the report subsections. The best accuracy score obtained for subtask A, subtask B, subtask C are 82.09%, 90.42%, 69.08%.

1 Credits

This document has been adapted from the official ACL 2019 website [link] and template adapted from link. This report contains a total of 8 pages according to the ACL format standards. The SemEval 2019 Task 6 OffensEval was organized by Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, Ritesh Kumar.

2 Introduction

One of the integral parts of Natural language processing is text data processing which includes classification of some sentence into some category. For classification, various methods can be used which can be supervised and unsupervised. For the purpose of classification in our project we have used various supervised classification methods such as Support Vector Machine, Decision Tree, K Nearest Neighbor, Logistic Regression and a semi supervised method which is Long Short-Term Memory model.

The dataset used for the purpose of classification and categorization is OLID (Offensive Language Identification Dataset) provided officially by Codalabs and is available here. The training dataset contains tweets with their unique id and their labels for all 3 subtasks. In this project, only English language tweets are taken into account. Offensive language classification can be done for various other languages also, which is proposed in SemEval 2020 [SemEval 2020].

3 Motivation

Offensive language has become an unsought part of social media such as twitter. People not only tweet offensive tweets but also spread hate and toxicity in society by taking advantage of anonymity provided by the social media platforms. Many researches have been done to prevent it and one of the best methods observed is computational methods such as natural language text processing. The high accuracy of NLP models for classification tasks make NLP classification methods most suitable for such a task.

4 Classification and Categorization

For this task of categorization and classification of offensive tweet, there are 3 sub categories.

These subcategories are defined in the official SemEval Task page.

Subtask A – Offensive Language Identification
[OFF (Offensive) or NOT (Not Offensive)]
To identify whether the given tweet is an offensive tweet or not.

Subtask B – Automatic Categorization of Offense Type
[UNT (Untargeted) or TIN (Targeted insult)]
To identify whether the offensive tweet is an untargeted insult or a targeted insult tweet.

Subtask C – Offense Target Identification
[IND (Individual) or GRP (Group) or OTH (Others)]
To identify whether the targeted insult tweet is targeted to an individual, group or others.

All the subtasks are done in the project and their corresponding results are shown in the report.

5 Classification and Categorization

The dataset used is OLID dataset which is provided officially on SemEval 2019 Task 6 page. OLID contains 13240 tweets each labelled with their id with which they were tweeted and their corresponding labels for subtask A, subtask B & subtask C.
A sample tweet is shown below

id	tweet	subtask_a	subtask_b	subtask_c
86426	@USER She should ask a few native Americans what their take on this is.	OFF	UNT	NULL
90194	@USER @USER Go home you’re drunk!!! @USER #MAGA #Trump2020 [illegible] URL	OFF	TIN	IND

Fig 1 Sample tweet

OLID contains 3 test set data files for subtask A, subtask B and subtask C which contains the test tweets for each subtask respectively. Test set for subtask A contains a total of 860 tweets, test set for subtask B contains a total of 240 tweets and test set for subtask C contains a total of 213 tweets with their tweet handle id and tweet text. OLID also contains gold labels for each test set with which confusion matrix and accuracy for the task can be calculated.

6 System Description

We used Naïve Bayes as our baseline model. For the project we have used Support Vector Machine, Decision Tree, K Nearest Neighbor, Logistic Regression and LSTM (Long Short-Term Memory model). We compared their accuracy, precision & recall for all the subtasks.

All the models are implemented in Python and makes use of Tensorflow (Abadi et al., 2015), Sklearn (Pedregosa et al., 2011) and Keras for training the classifiers. We used TF-IDF Matrix for words and used Label encoder to label the tags. We picked the epochs according to the best accuracy score for each subtask done with LSTM. The dataset contained data in form of .tsv files (tab delimited file) which was imported and processed using Pandas library. For LSTM, CUDNNLSTM is used which is gpu accelerated and gives 300% time faster results.

6.1 Preprocessing

Further Preprocessing was not useful for the training of our models as we have used TF-IDF matrix which eliminates and give very less weight to very frequent words such as stop words. We tested the accuracy after applying preprocessing such as lower casing and stop words removal but there was no such noticeable improvement in accuracy.

6.2 Pipeline for Subtask A

1. Extracted Tweet text and converting to tokens for TF-IDF processing.

2. TF-IDF matrix is given to classifier as input and it performs binary classification prediction for each tweet as OFF or NOT.

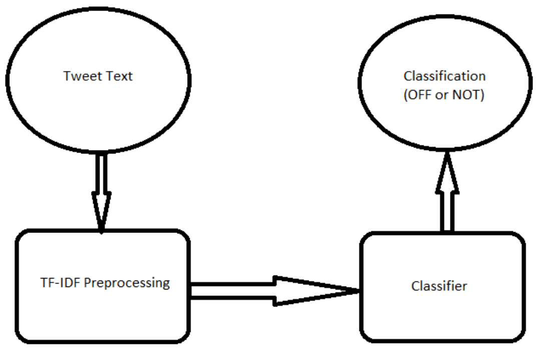

Fig 2 Pipeline for Subtask A

6.3 Pipeline for Subtask B

1. Extracted Tweet text and converting to tokens for TF-IDF processing.

2. TF-IDF matrix is given to classifier as input and it performs binary classification prediction for each offensive tweet as UNT or TIN.

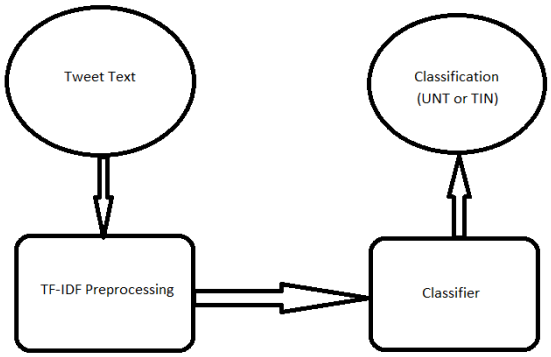

Fig 3 Pipeline for Subtask B

6.4 Pipeline for Subtask C

1. Extracted Tweet text and converting to tokens for TF-IDF processing.

2. TF-IDF matrix is given to classifier as input and it performs classification prediction for each targeted insult tweet as IND, GRP or OTH.

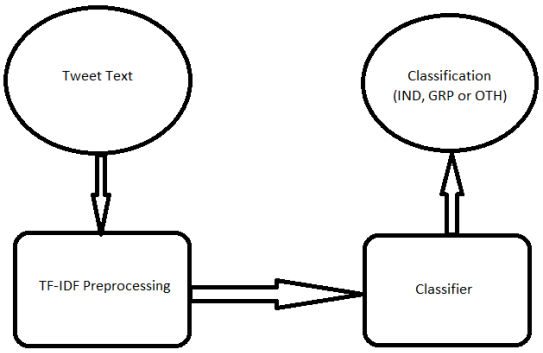

Fig 4 Pipeline for Subtask C

7 Results

7.1 Subtask A Results

For subtask A, we found out that all classifiers gave accuracy higher than 76% and highest of 82.09% using SVM (Support Vector Machine).

SVM (Support Vector Machine)

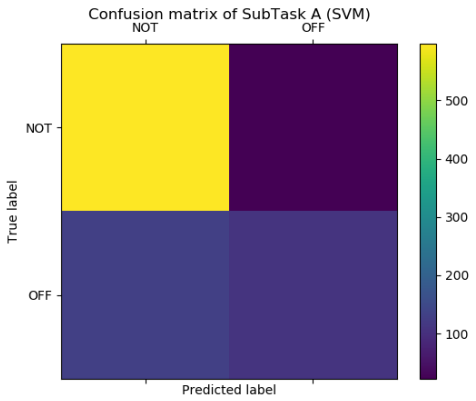

Fig 5 Task A Confusion Matrix SVM

Classification report for SVM

	True NOT	True OFF
System NOT	598	22
System OFF	132	108

	Precision	Recall	F1 Score	Support
NOT	0.82	0.96	0.89	620
OFF	0.83	0.45	0.58	240

Accuracy=82.09%

DT (Decision Tree)

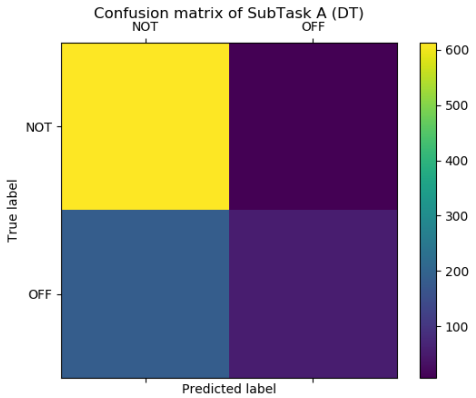

Fig 6 Task A Confusion Matrix DT

Classification report for Decision Tree

	True NOT	True OFF
System NOT	613	7
System OFF	185	55

	Precision	Recall	F1 Score	Support
NOT	0.77	0.99	0.86	620
OFF	0.89	0.23	0.36	240

Accuracy=77.67%

KNN (K Nearest Neighbor)

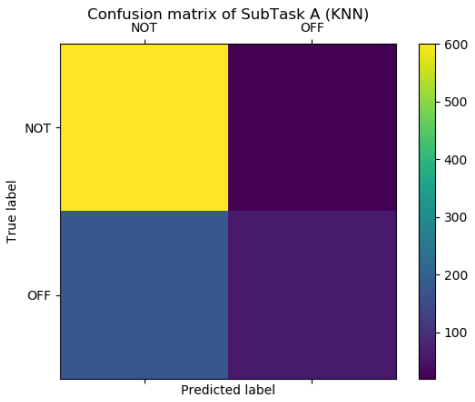

Fig 7 Task A Confusion Matrix KNN

Classification report for K Nearest Neighbor

	True NOT	True OFF
System NOT	601	19
System OFF	179	61

	Precision	Recall	F1 Score	Support
NOT	0.77	0.97	0.86	620
OFF	0.76	0.25	0.38	240

Accuracy=76.98%

LR (Logistic Regression)

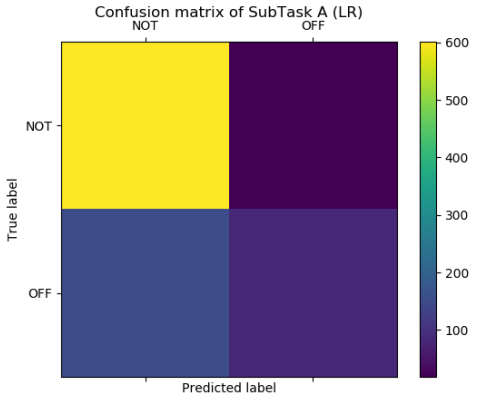

Fig 8 Task A Confusion Matrix LR

Classification report for Logistic Regression

	True NOT	True OFF
System NOT	602	18
System OFF	156	84

	Precision	Recall	F1 Score	Support
NOT	0.79	0.97	0.87	620
OFF	0.82	0.35	0.49	240

Accuracy=79.77%

LSTM (Long Short-Term Memory)

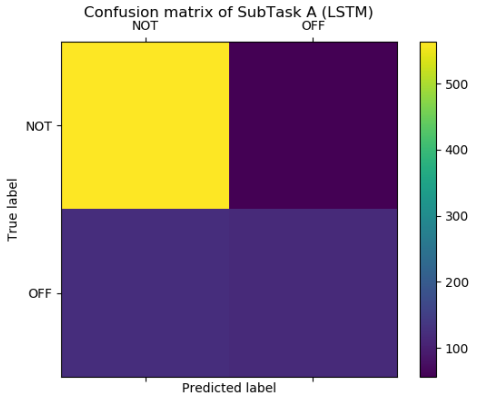

Fig 9 Task A Confusion Matrix LSTM

Classification report for LSTM

	True NOT	True OFF
System NOT	564	56
System OFF	123	117

	Precision	Recall	F1 Score	Support
NOT	0.82	0.91	0.86	620
OFF	0.68	0.49	0.57	240

Accuracy=79.19%

7.2 Subtask B Results

For subtask B, we found out that all classifiers gave accuracy higher than 88% and highest of 90.42% using KNN (K Nearest Neighbor).

SVM (Support Vector Machine)

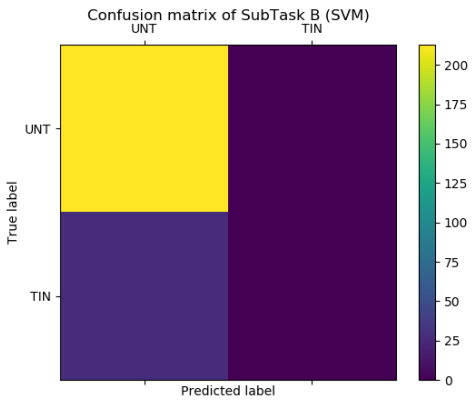

Fig 10 Task B Confusion Matrix SVM

Classification report for SVM

	True TIN	True UNT
System TIN	213	0
System UNT	27	0

	Precision	Recall	F1 Score	Support
TIN	0.89	1.00	0.94	213
UNT	0.00	0.00	0.00	27

Accuracy=88.75%

DT (Decision Tree)

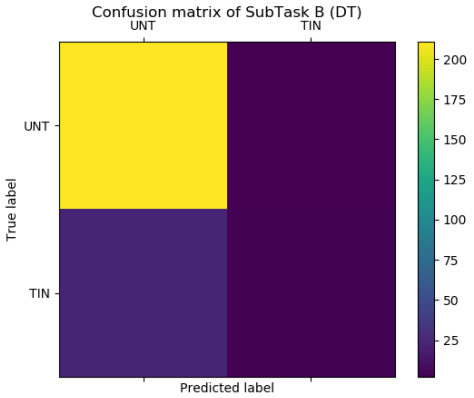

Fig 11 Task B Confusion Matrix DT

Classification report for Decision Tree

	True TIN	True UNT
System TIN	211	2
System UNT	24	3

	Precision	Recall	F1 Score	Support
TIN	0.90	0.99	0.94	213
UNT	0.60	0.11	0.19	27

Accuracy=89.17%

KNN (K Nearest Neighbor)

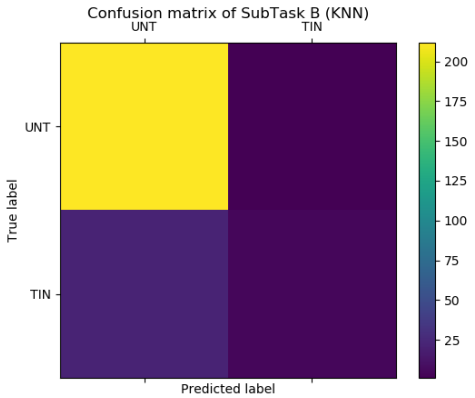

Fig 12 Task B Confusion Matrix KNN

Classification report for K Nearest Neighbor

	True TIN	True UNT
System TIN	212	1
System UNT	22	5

	Precision	Recall	F1 Score	Support
TIN	0.91	1.00	0.95	213
UNT	0.83	0.19	0.30	27

Accuracy=90.42%

LR (Logistic Regression)

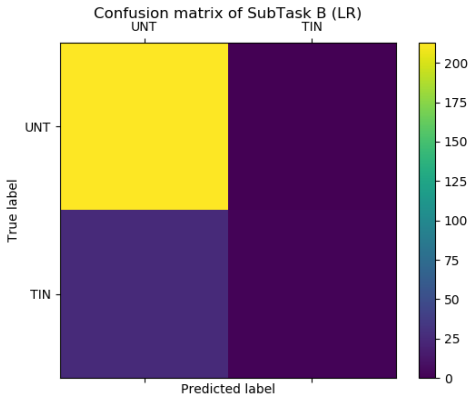

Fig 13 Task B Confusion Matrix LR

Classification report for Logistic Regression

	True TIN	True UNT
System TIN	213	0
System UNT	25	2

	Precision	Recall	F1 Score	Support
TIN	0.89	1.00	0.94	213
UNT	1.00	0.07	0.14	27

Accuracy=89.58%

LSTM (Long Short-Term Memory)

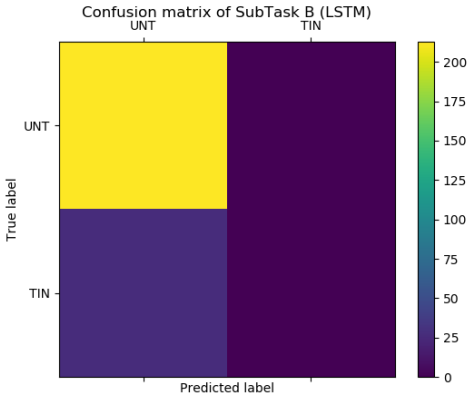

Fig 14 Task B Confusion Matrix LSTM

Classification report for LSTM

	True TIN	True UNT
System TIN	213	0
System UNT	27	0

	Precision	Recall	F1 Score	Support
TIN	0.89	1.00	0.94	213
UNT	0.00	0.00	0.00	27

Accuracy=88.75%

7.3 Subtask C Results

For subtask C, we found out that all classifiers gave accuracy higher than 61% and highest of 68.08% using LSTM (Long Short-Term Memory).

SVM (Support Vector Machine)

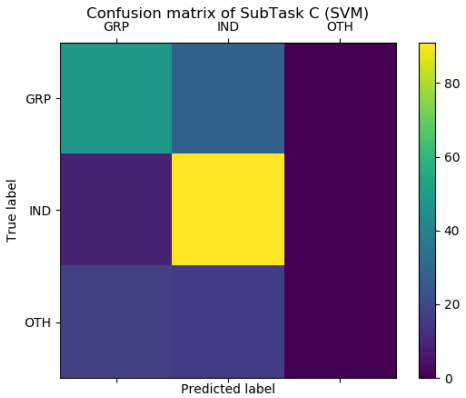

Fig 15 Task C Confusion Matrix SVM

Classification report for SVM

	True GRP	True IND	True OTH
System GRP	49	29	0
System IND	9	91	0
System OTH	18	17	0

	Precision	Recall	F1 Score	Support
GRP	0.64	0.63	0.64	78
IND	0.66	0.91	0.77	100
OTH	0.00	0.00	0.00	35

Accuracy=65.73%

DT (Decision Tree)

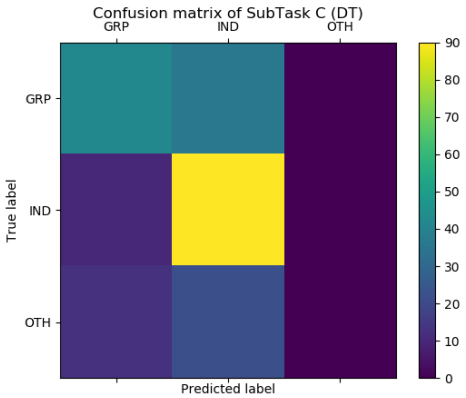

Fig 16 Task C Confusion Matrix DT

Classification report for DT

	True GRP	True IND	True OTH
System GRP	42	36	0
System IND	10	90	0
System OTH	13	22	0

	Precision	Recall	F1 Score	Support
GRP	0.65	0.54	0.59	78
IND	0.61	0.90	0.73	100
OTH	0.00	0.00	0.00	35

Accuracy=61.97%

KNN (K Nearest Neighbor)

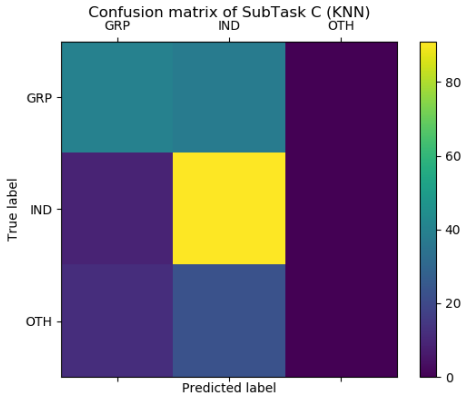

Fig 17 Task C Confusion Matrix KNN

Classification report for KNN

	True GRP	True IND	True OTH
System GRP	40	38	0
System IND	9	91	0
System OTH	12	23	0

	Precision	Recall	F1 Score	Support
GRP	0.66	0.51	0.58	78
IND	0.60	0.91	0.72	100
OTH	0.00	0.00	0.00	35

Accuracy=61.50%

LR (Logistic Regression)

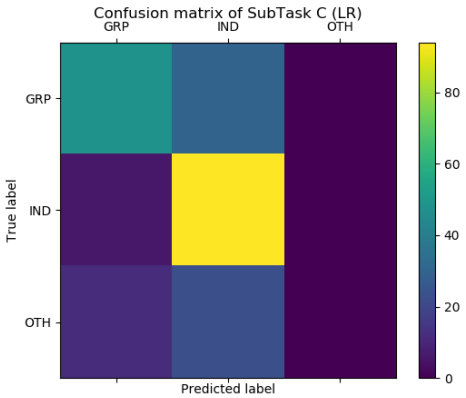

Fig 18 Task C Confusion Matrix LR

Classification report for LR

	True GRP	True IND	True OTH
System GRP	48	30	0
System IND	6	94	0
System OTH	12	23	0

	Precision	Recall	F1 Score	Support
GRP	0.73	0.62	0.67	78
IND	0.64	0.94	0.76	100
OTH	0.00	0.00	0.00	35

Accuracy=66.67%

LSTM (Long Short-Term Memory)

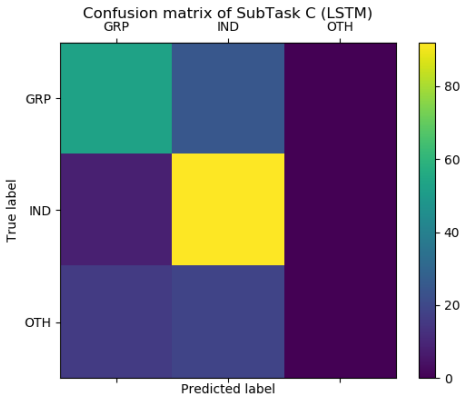

Fig 19 Task C Confusion Matrix LSTM

Classification report for LSTM

	True GRP	True IND	True OTH
System GRP	53	25	0
System IND	8	92	0
System OTH	16	19	0

	Precision	Recall	F1 Score	Support
GRP	0.69	0.68	0.68	78
IND	0.68	0.92	0.78	100
OTH	0.00	0.00	0.00	35

Accuracy=68.08%

8 Conclusion and Future Work

From the given comparison of accuracy, we can infer that for subtask A SVM model performed the best, for subtask B KNN model performed the best and for subtask C LSTM model performed the best. In future the same task can be done for multiple language tweets with more category classes as well.

9 References

Zampieri, Marcos and Malmasi, Shervin and Nakov, Preslav and Rosenthal, Sara and Farra, Noura and Kumar, Ritesh. *Predicting the Type and Target of Offensive Posts in Social Media, Proceedings of NAACL 2019.*

Himanshu Bansal, Daniel Nagel and Anita Soloveva. *HAD-Tübingen at SemEval-2019 Task 6: Deep Learning Analysis of Offensive Language on Twitter: Identification and Categorization.* University of Tübingen 2019.

10 Accuracy Comparison

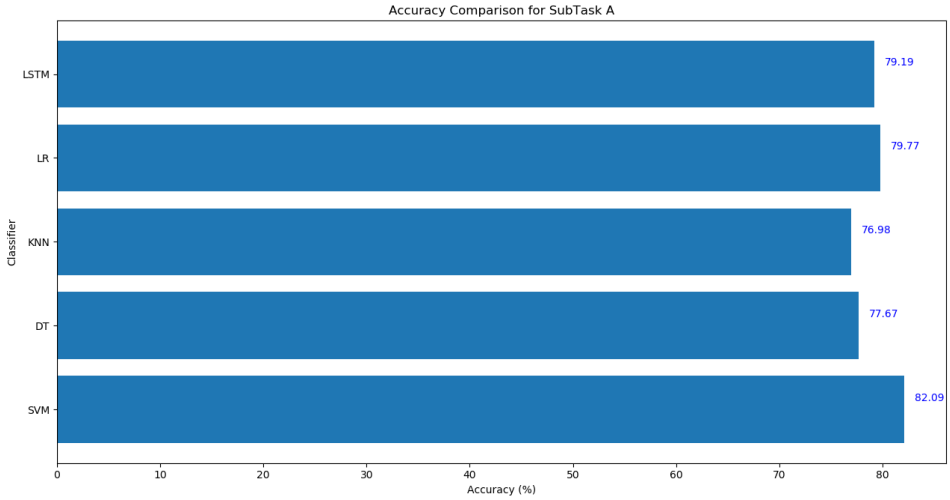

Fig 20 Accuracy Comparison for Subtask A

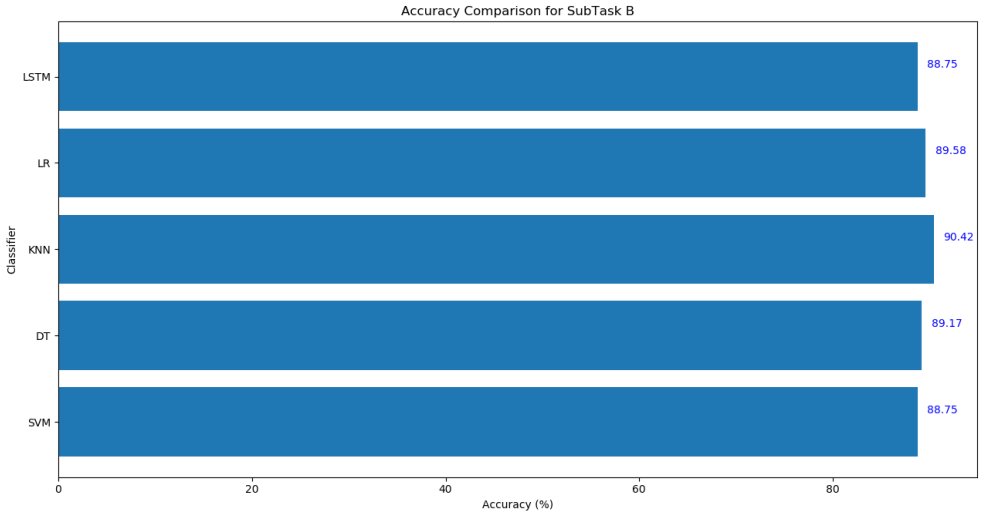

Fig 21 Accuracy Comparison for Subtask B

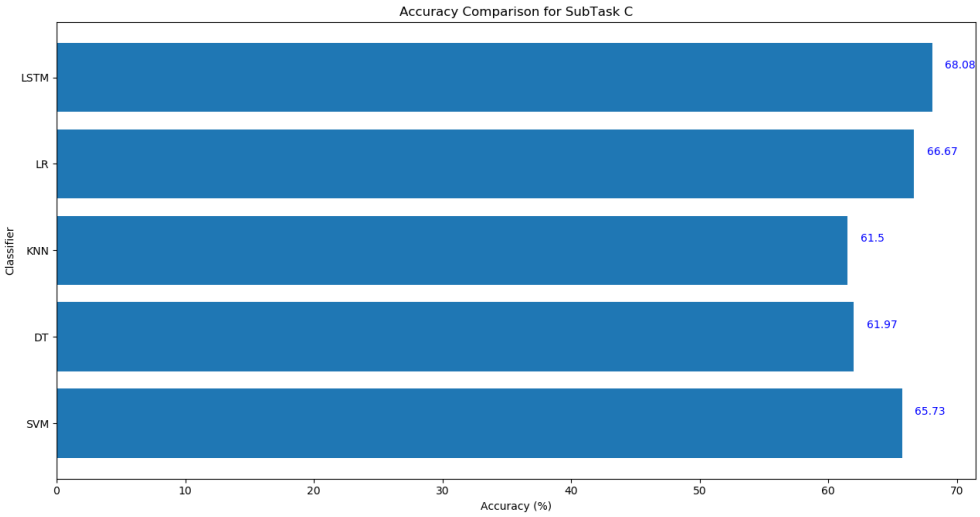

Fig 22 Accuracy Comparison for Subtask C