****

CSI 4900 Honour Project Report

**Detecting and Categorizing Offensive Language in Twitter messages**

**University of Ottawa**

Supervisor: Diana Inkpen

Team members:

He, Xinyi 300072163

Chen,Wenwen 300070021

Wu, Haozhe 300010064

Submission Date: 2021/04/26

**1 Introduction**

In recent years, offensive language on various social media platforms has surged. Due to the anonymity of network communication, many netizens take advantage of this feature to say something in social media that would not be considered in real life. Especially during the epidemic, people need to be isolated at home, which can easily lead to emotional loss of control, venting their emotions on social platforms, making offensive language appear more frequently.

Major technology companies have invested heavily in this problem to find effective solutions to prevent language violence in social media. Since manual screening is very time-consuming and it may bring symptoms of post-traumatic stress disorder to workers, many studies have been devoted to automating this process.

One of the most effective strategies to solve this problem is to use computational methods to identify offenses, attacks, and hate speech in user-posted content. The task is usually modeled as a supervised classification project, where systems are trained on posts annotated with respect to the presence of some form of abusive or offensive content. A group of researchers organized an Offensive Language Detection challenge (Zampieri et al., 2019b) as part of the SemEval 2019 competition. This report covers the details of our submission to this project.

This Project has three subtasks:

**Subtask A**: Offensive language identification

**Subtask B**: Automatic categorization of offense

**Subtask C**: Offense target identification

The remainder of this paper is organized as follows: Section 2 presents the shared task description and the subtasks included in OffensEval. Section 3 discusses prior work, including shared tasks related to OffensEval. Section 4 includes a brief description of OLID based on (Zampieri et al., 2019). Section 5 discusses three methods used in this project. Section 6 discusses and compares the results for each of the three subtasks using the above methods. Finally, Section 7 improves and suggests directions for future work.

**2 Task Description and Evaluation .**

**Subtask A:** The goal is to discriminate between offensive and non-offensive posts. Offensive posts include insults, threats, and posts containing any form of

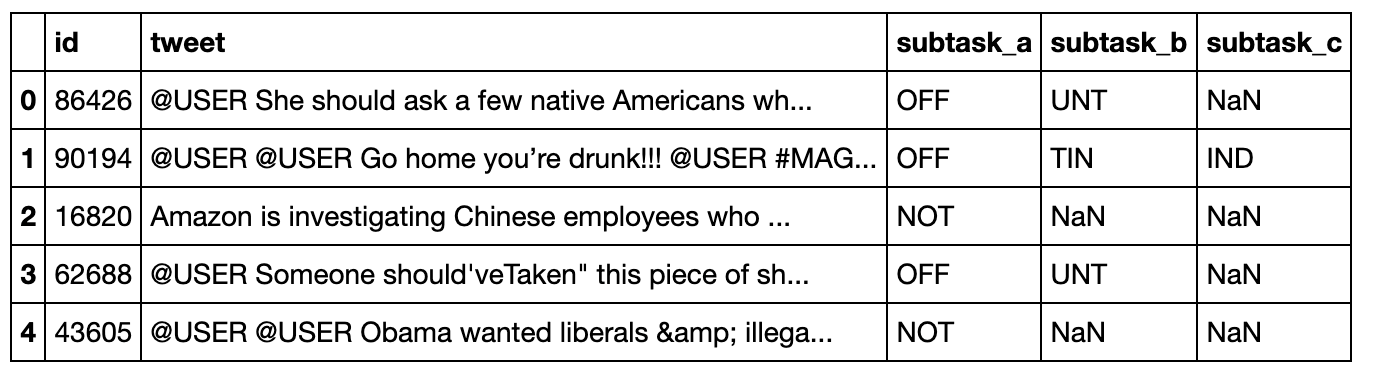


Table 1 OLID data in olid-training-v1.0.tsv

untargeted profanity. Each instance is assigned one of the following two labels.

* **Not Offensive (NOT):** Posts that do not contain offense or profanity;
* **Offensive (OFF):** label a post as offensive if it contains any form of non-acceptable language (profanity) or a targeted offense, which can be veiled or direct.

**Subtask B**: The goal is to predict the type of offense. Only posts labeled as Offensive (OFF) in subtask A are included in subtask B. The two categories in subtask B are the following:

* **Targeted Insult (TIN):** Posts containing an insult to an individual, group, or others.
* **Untargeted (UNT):** Posts containing non-targeted profanity and swearing. Posts with general profanity are not targeted, but they contain non-acceptable language.

**Subtask C**: Focuses on the target of offenses. Only posts that are either insults or threats (TIN) are considered in this third layer of annotation. The three labels in subtask C are the following:

* **Individual (IND):** Posts targeting an individual.
* **Group (GRP):** The target of these offensive posts is a group of people considered as a unity due to the same ethnicity, gender or other common characteristic.
* **Other (OTH):** The target of these offensive posts does not belong to any of the previous two categories, like a situation, an event.

**3 Pre-Processing**

First, we removed the unlabeled tweets for each task. Then we proceeded onto preprocessing the data. The idea was to put all the tweet into lower case, and delete useless punctuation, to make them as uniform and standardized as possible so that the model could compare them easily.

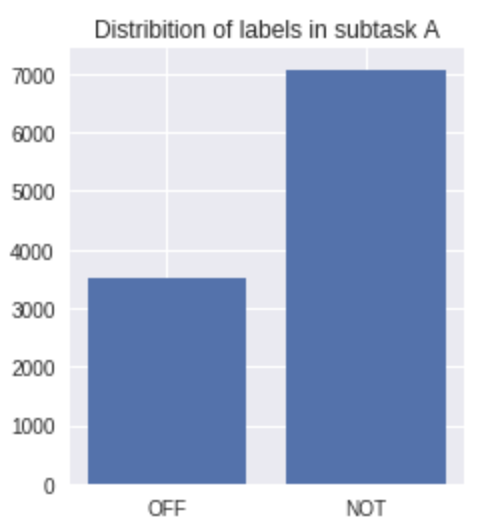
This means we had to remove ’#’ and ’@’, remove repetitive ’@USER’, and add space between words and punctuation (such as ’!’ or ’?’).

Additionally, we tried creating an extra column to store the number of ’@USER’ for each tweet. The initial thought was that there could be several advantages of doing so: we thought it would be an important feature, which should be captured by the model, and we didn’t want to lose this information but still wanted to remove redundant ’@USER’ during our cleaning process, as these were playing and overly emphasized role in the model. After testing, we determined those changes either did not contribute positively or decreased the performance of our classifiers.

**4 Data**

To train and evaluate our models, we will be using the provided training and trial dataset. The task was based on a new dataset, the Offensive Language Identification Dataset (OLID), which contains over 14,000 English tweets. This dataset takes both the target and the type of offensive content into account. To train our model, we split the combined dataset randomly into 90% train-val and 10% test set. Table1 shows the label distribution of all the datasets.

For each subtask:

subtask A:

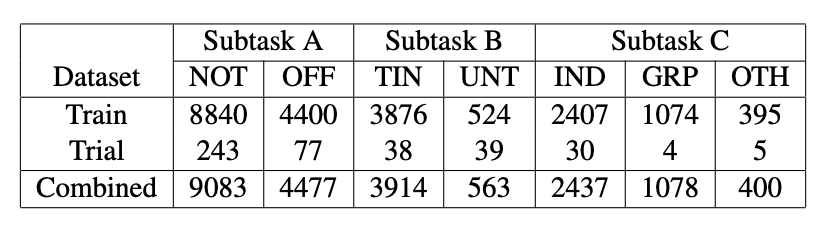
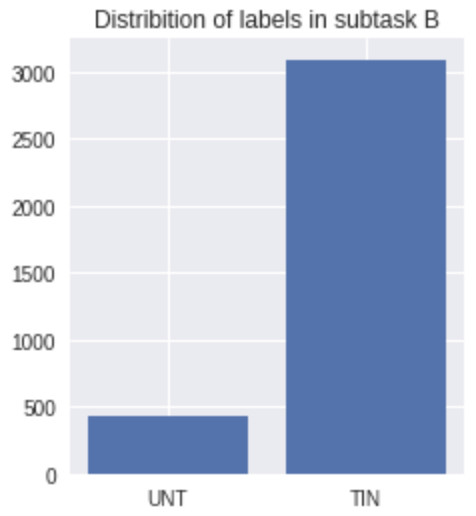
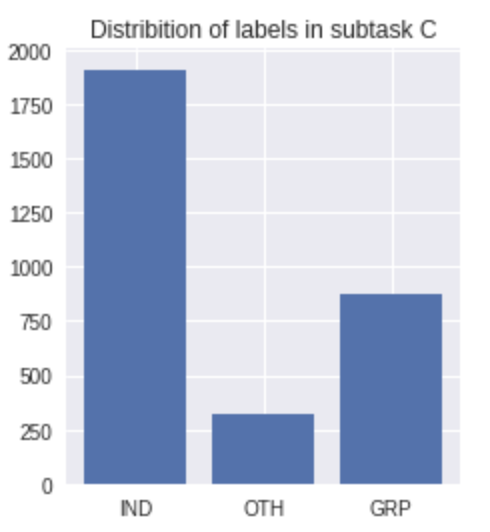


Table 1: Benchmark dataset label distribution

Subtask B:



Subtask C:



Since the distribution is not equal, we just choose 4000 from both sample in task A, 500 for taskB, 390 for task C. To prevent undersampling or Oversampling, we choose equal size of results.

**5 Method Implementation**

5.1 Naive Bayes classifier

Naive Bayes classifiers are a family of simple "probabilistic classifiers".The probabilistic classifier is used based on the given inputs and output the most probable class it belongs to. The Naive Bayes classifiers using Bayes' theorem to predict the input, The Bayes’ Formula is:

Where A and B are events where p(B)≠0

P(A|B) is a conditional probability: the probability of event A occurring given that B is true. P(B|A) is also a conditional probability: the probability of event B occurring given that A is true. P(A) and P(B) are the probabilities of observing A and B respectively without any given conditions; they are known as the marginal probability or prior probability. This mathematical theory is used to predict the result in this project. After the model is trained, The Naive Bayes model can be used for classification in different classes.The different tweet messages will be vectorized first, and converted into a numerical matrix, That numerical matrix will be calculated based on that Bayes mathematical equation. Comparing the result of test data and based on a pre-trained model, the class will be easily classified.by comparing advantage and disadvantage used in this method, the advantage is that,

1.The Naive Bayes model has stable classification efficiency.

2. The Naive Bayes performs well on small-scale data, can handle multi-classification tasks, and is suitable for incremental training, especially when the amount of data exceeds the memory, batches of incremental training can be performed.

However there are also some disadvantages which are:

1.Since we use a priori and data to determine the posterior probability to determine the classification, there is a certain error rate in the classification decision.

2.In theory, the Naive Bayes model has the smallest error rate compared with other classification methods.However, this is not always the case. This is because the Naive Bayes model assumes that the attributes are independent of each other. This assumption is often invalid in practical applications.

**5.2 Random forests**

Random forests or random decision forests are an ensemble learning method for classification, regression.It constructs a multitude of decision trees at training time,The Random Forest combines different decision trees, built against a subset of the features, it is not pruned, and diversity in features/attributes utilised. The Random forest will use the average values or mean values for the decision trees outputting.In simplify, Random forest is using multiple decision trees synchronously. Then it is necessary to know about the decision tree, the decision tree is a decision support tool that uses a tree-like model of decisions and their possible consequences, including chance event outcomes, resource costs, and utility. The Internal node denotes a test on an attribute,branch represents an outcome of the test, All tuples in branch have the same value for the tested attribute and leaf node represents class label It is one way to display an algorithm that only contains conditional control statements. by comparing advantage and disadvantage used in this method, the advantage is that,

1. It can handle very high-dimensional (a lot of features) data, and there is no need to do feature selection

2. If a large part of the features are missing, the accuracy can still be maintained.

3. It can balance error,if there are some unbalanced data sets

4.When creating a random forest, unbiased estimation is used for the generalization error, and the model has strong generalization ability

but the disadvantage is that:

1.Random forests have been proven to be over-fitting in some noisy classification or regression problems

2.For data with attributes with different values, attributes with more value divisions will have a greater impact on the random forest, so the attribute weights produced by the random forest on this kind of data are unreliable.

**5.3 SVM**

support-vector machine**s** are supervised learning models with associated learning algorithms that analyze data for classification and regression analysis. In other words, it just needs to find a best-fit line that can simply divide two different sets.Normally svm will only perform with divided two labels, but there are also some improvements with this algorithm (SVMs)，or svc (support vector classifier) to classified for multi classes.  
comparing advantage and disadvantage used in this method, the advantage is that,

1.Non-linear mapping is the theoretical basis of SVM method, SVM uses inner product kernel function to replace nonlinear mapping to high-dimensional space;

2. The optimal hyperplane to divide the feature space is the goal of SVM, and the idea of ​​maximizing the classification margin is the core of the SVM method;

3.support vector is the training result of SVM, and it is the support vector that plays a decisive role in SVM classification decisions.

but the disadvantage is that:

1.SVMs are Relatively slower and using more space

2.The classic support vector machine algorithm only gives the two-class classification algorithm, but in the practical application of data mining, generally it is necessary to solve the multi-class classification problem.

There is a theory called the “No free lunch” theorem, which means No single algorithm is always the most accurate in all situations, so for the three tasks, it is beneficial for different algorithms for various tasks.The further analyse is given below.

**6. Result and Conclusion**

Finally, it is important to display the final result in a nice and tidy way. And then we make some comparisons and conclusions towards the result we have. We discuss why the model has the best performance for each task.

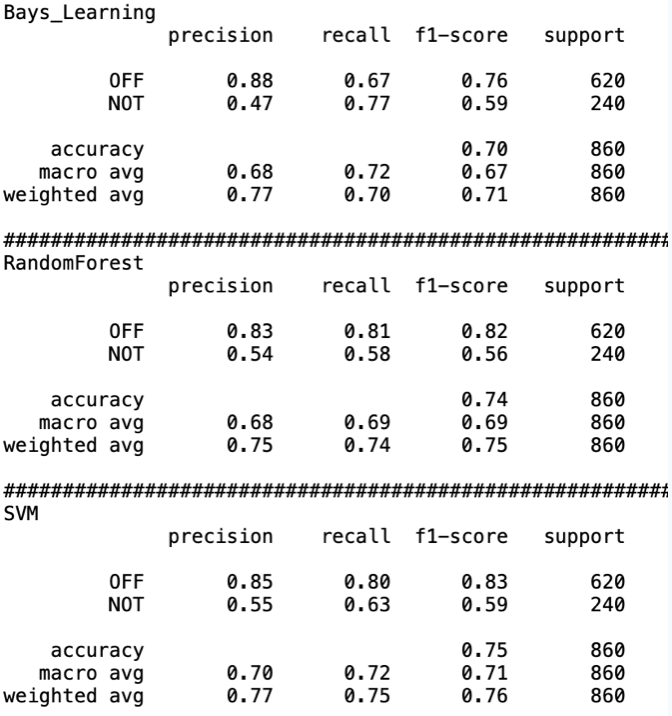
6.1 Display Result

For each part, we display the result by using the method clssification\_report from scikit learn.metrics. It is a built-in function that is used for getting precision, recall, f1-score and support value, where:

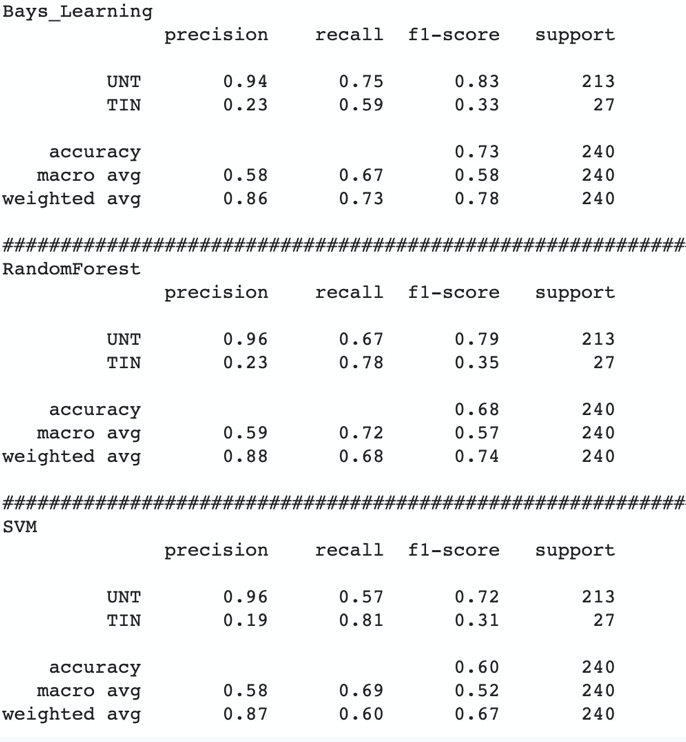
1. Precision is defined as the number of true positives over the number of true positives plus the number of false positives.
2. Recall is defined as the number of true positives over the number of true positives plus the number of false negatives.
3. The F1 score can be understood as a weighted average of the precision and recall, where its value is between [0,1] The formula for the F1 score is: F1 = 2 \* (precision \* recall) / (precision + recall)
4. Support number means the number of occurrences of each label in y\_true.

Here we have the report for each model in each task:

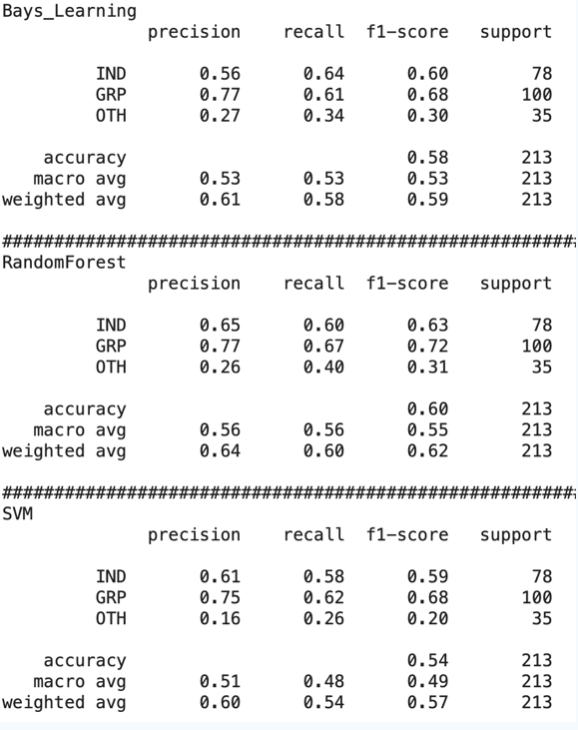
Task A:



Task B:



Task C:



6.2 Comparisons

From the results above, we can make comparisons to each model in each task.

1. For the models in task A, the best model is considered as SVM, because it has the highest f1-score of 0.75.
2. For the models in task B, we think that Bayes\_Learning has the best performance, because it has the f1-score of 0.73. The other two models, random forest has f1-score of 0.68 and svm has f1-score of 0.6.
3. For the models in task C, we conclude that Random Forest is the most suitable model, because it has the highest f1-score of 0.6. In the same task, Bayes\_learning has the f1-score of 0.58 and SVM has the lowest score of 0.54.

6.3 Conclusions

By comparing the results for each model in the three sub-tasks, we can make some conclusions towards the results.

Firstly, the results for the three models in task A have the overall highest f1-score, while the result in task C is overall the lowest no matter which model we choose. We think about the reason that may cause this situation and in conclusion we think that it may be because of the number of samples we choose. For task A, we choose 4000 samples for each label, while in task C we only choose 390. Because in task C there are only 395 samples that have the label “group”.

Secondly, the svm model in task A has the highest f1-score among all 3 sub tasks. This may because Task A is a Binary classification problem that is very suitable for us to use the SVM model.

**7. Ideas on Future Works**

We also have some improvements for future work. We could increase the size of the original data-set. And we can preprocess the original data with more methods.

For the main idea that we use in the project now is deep learning, we can use some other ideas to compare results such as convolutional neural networks and recurrent neural networks. Or we plan to research bidirectional recurrent models.

**References**

**1.R, 6., & Ray, S. (2017). Learn Naive Bayes Algorithm | Naive Bayes Classifier Examples. Retrieved 22 April 2021, from** [**https://www.analyticsvidhya.com/blog/2017/09/naive-bayes-explained/**](https://www.analyticsvidhya.com/blog/2017/09/naive-bayes-explained/)

**2.朴素贝叶斯算法优缺点\_zrh\_CSDN的博客-CSDN博客\_朴素贝叶斯优缺点. (2021). Retrieved 20 April 2021, from** [**https://blog.csdn.net/zrh\_CSDN/article/details/81007851#:~:text=%E4%BC%98%E7%82%B9%EF%BC%9A1.%E7%AE%80%E5%8D%95%E5%A5%BD%E7%94%A8,%E5%BE%88%E5%A4%A7%E7%9A%84%E6%97%B6%E5%80%99…**](https://blog.csdn.net/zrh_CSDN/article/details/81007851#:~:text=%E4%BC%98%E7%82%B9%EF%BC%9A1.%E7%AE%80%E5%8D%95%E5%A5%BD%E7%94%A8,%E5%BE%88%E5%A4%A7%E7%9A%84%E6%97%B6%E5%80%99%E2%80%A6)

**3.Bayes' theorem - Wikipedia. (2021). Retrieved 20 April 2021, from** [**https://en.wikipedia.org/wiki/Bayes%27\_theorem**](https://en.wikipedia.org/wiki/Bayes%27_theorem)

**4.Precision-Recall — scikit-learn 0.24.1 documentation. (2021). Retrieved 20 April 2021, from** [**https://scikit-learn.org/stable/auto\_examples/model\_selection/plot\_precision\_recall.html**](https://scikit-learn.org/stable/auto_examples/model_selection/plot_precision_recall.html)

**5.LIU, P. (2021). 朴素贝叶斯算法原理小结 - 刘建平Pinard - 博客园. Retrieved 20 April 2021, from** [**https://www.cnblogs.com/pinard/p/6069267.html**](https://www.cnblogs.com/pinard/p/6069267.html)

**6.Sharma, A. (2021). Decision Tree vs. Random Forest - Which Algorithm Should you Use?. Retrieved 20 April 2021, from** [**https://www.analyticsvidhya.com/blog/2020/05/decision-tree-vs-random-forest-algorithm/**](https://www.analyticsvidhya.com/blog/2020/05/decision-tree-vs-random-forest-algorithm/)

**7.sklearn.svm.SVC — scikit-learn 0.24.1 documentation. (2021). Retrieved 20 April 2021, from** [**https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html**](https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html)