MIDAS at SemEval-2019 Task 9: Suggestion Mining from Online Reviews using ULMFiT

Sarthak Anand,[3] Debanjan Mahata,[1] Kartik Aggarwal, [3] Laiba Mehnaz,[2] Simra Shahid,[2]

Haimin Zhang,[1] Yaman Kumar,[5] Rajiv Ratn Shah,[4] Karan Uppal[1]

[1]Bloomberg, USA, [2]DTU-Delhi, India, [3]NSIT-Delhi, India, [4]IIIT-Delhi, India, [5]Adobe, India,
sarthaka.ic@nsit.net.in, dmahata@bloomberg.net, kartik.mp.16@nsit.net.in,
laibamehnaz@dtu.ac.in, simrashahid_bt2k16@dtu.ac.in, hzhang449@bloomberg.net,
ykumar@adobe.com, rajivratn@iiitd.ac.in, kuppal8@bloomberg.net

Abstract

In this paper we present our approach and the system description for Sub Task A of SemEval 2019 Task 9: Suggestion Mining from Online Reviews and Forums. Given a sentence, the task asks to predict whether the sentence consists of a suggestion or not. Our model is based on Universal Language Model Fine-tuning for Text Classification. We apply various pre-processing techniques before training the language and the classification model. We further provide detailed analysis of the results obtained using the trained model. Our team ranked 10th out of 34 participants, achieving an F1 score of 0.7011. We publicly share our implementation[1].

1 Introduction and Background

Suggestion mining can be defined as the process of identifying and extracting sentences from unstructured text that contain suggestion (Negi et al., 2018). Suggestions in the form of unstructured text could be found in various social media platforms, discussion forums, review websites and blogs. They are often expressed in the form of advice, tips, recommendations, warnings, things to do, and various other forms in an explicit as well as an implicit way.

Identifying and retrieving suggestions from text can be useful in an industrial setting for enhancing a product, summarizing opinions of the consumers, giving recommendations and as an aid in decision making process (Jijkoun et al., 2010). For normal users of online platforms it could help in seeking advice related to general topics of interest like travel, health, food, shopping, education, and many more. Given the abundance of textual information in the Internet about a variety of topics, suggestion mining is certainly an useful task interesting to researchers working in academia as well as industry.

Most of the previous efforts in the direction of understanding online opinions and reactions have been limited to developing methods for areas like sentiment analysis and opinion mining (Medhat et al., 2014; Baghel et al., 2018; Kapoor et al., 2018; Mahata et al., 2018a,b; Jangid et al., 2018; Meghawat et al., 2018; Shah and Zimmermann, 2017). Mining and understanding suggestions can open new areas to study consumer behavior and tapping nuggets of information that could be directly linked with the development and enhancement of products (Brun and Hagege, 2013; Dong et al., 2013; Ramanand et al., 2010), improve customer experiences (Negi and Buitelaar, 2015), and aid in understanding the linguistic nuances of giving advice (Wicaksono and Myaeng, 2013).

Suggestion mining is a relatively new domain and is challenged by problems such as *ambiguity in task formulation and manual annotation*, *understanding sentence level semantics*, *figurative expressions*, *handling long and complex sentences*, *context dependency*, and *highly imbalanced class distribution*, as already mentioned by (Negi et al., 2018). Similar problems are also observed in the dataset shared by the organizers for the SemEval task, as it is obtained from a real-world application comprising of suggestions embedded in unstructured textual content.

Problem Definition - The problem of suggestion mining as presented in the SemEval 2019 Task 9 (Negi et al., 2019), is posed as a binary classification problem and could be formally stated as:

[1] https://github.com/isarth/SemEval9_MIDAS

Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019), pages 1213–1217
Minneapolis, Minnesota, USA, June 6–7, 2019.

Given a labeled dataset D of sentences, the objective of the task is to learn a classification/prediction function that can predict a label l for a sentence s, where $l \in \{suggestion, nonsuggestion\}$.

Our Contributions - Some of the contributions that we make by participating in this task are:

• To our knowledge we are the first one to use Universal Language Model Fine-tuning for Text Classification (ULMFiT) (Howard and Ruder, 2018), for the task of suggestion mining and show the effectiveness of transfer learning.

• We perform an error analysis of the provided dataset for Sub Task A, as well as the predictions made by our trained model.

Next, we give a detailed description of our system and the experiments performed by us along with explaining our results.

2 Experiments

2.1 Dataset

The dataset used in all our experiments was provided by the organizers of the task and consists of sentences from a suggestion forum annotated by humans to be a *suggestion* or a *non-suggestion*. Suggestion forums are dedicated forums used for providing suggestions on a specific product, service, process or an entity of interest. The provided dataset is collected from uservoice.com[2], and consists of feedback posts on Universal Windows Platform. Only those sentences are present in the dataset that explicitly expresses suggestions, for example - *Do try the cupcakes from the bakery next door*, instead of those that contain implicit suggestions such as - *I loved the cup cakes from the bakery next door* (Negi et al., 2018).

Label	Train	Trial
Suggestion	2085	296
Non Suggestion	6415	296

Table 1: Dataset Distribution for Sub Task A - Task 9: Suggestion Mining from Online Reviews.

For Sub Task A, the organizers shared a training and a validation dataset whose label distribution (*suggestion* or a *non-suggestion*) is presented in Table 1. The unlabeled test data on which the performance of our model was evaluated was also from the same domain. As evident from Table 1, there is a significant imbalance in the distribution of training instances that are *suggestions* and *non-suggestions*, which mimics the distributions of these classes in the real-world datasets. Although the dataset was collected from a suggestion forum and is expected to have a high occurrence of suggestions, yet the imbalance is more prominent due to the avoidance of implicit suggestions.

2.2 Dataset Preparation

Before using the provided dataset for training a prediction model, we take steps to prepare it as an input to our machine learning models. We primarily use Ekphrasis[3] for implementing our pre-processing steps. Some of the steps that we take are presented in this section.

2.2.1 Tokenization

Tokenization is a fundamental pre-processing step and could be one of the important factors influencing the performance of a machine learning model that deals with text. As online suggestion forums include wide variation in vocabulary and expressions, the tokenization process could become a challenging task. Ekphrasis ships with custom tokenizers that understands expressions found in colloquial languages often used in forums and has the ability to handle hashtags, dates, times, emoticons, besides standard tokenization of English language sentences. We also had to tokenize certain misspellings and slangs (eg. "I'm", "r:are") after carefully inspecting the provided dataset.

2.2.2 Normalization

After tokenization, a range of transformations such as word-normalization, spell correction and segmentation are applied to the extracted tokens. During word-normalization, URLs, usernames, phone numbers, date, time, currencies and special type of tokens such as hashtags, emoticons, censored words etc. are recognized and replaced by masks (eg. <date>, <hashtag>, <url>). These steps results in a reduction in the vocabulary size without the loss of informative excerpts that has signals for expressing suggestions. This was validated manually by analyzing the text after applying the different processing steps. Table 2 shows an example text snippet and its form after the application of the pre-processing steps.

[2]https://www.uservoice.com/

[3]https://github.com/cbaziotis/ekphrasis

Text Snippet before Pre-processing	Text Snippet after Pre-processing
ie9mobile does not do this :(	ie mobile does not do this <emsad>
For example if you want a feed for every Tumblr feed containing the hashtags " "#retail #design " "; " "http://www.tumblr .com/tagged/retail+ design""; would be a feedly feed."	For example if you want a feed for every tumblr feed containing the hashtags <hashtag>retail <hashtag>design <url>would be a feedly feed

Table 2: Text snippet from the dataset before and after applying pre-processing steps.

2.2.3 Class Imbalance

As already pointed in Section 2.1, *class imbalance* is a prevalent challenge in this domain and is reflected in the provided dataset. We use oversampling technique in order to tackle this challenge. We duplicate the training instances labeled as *suggestions* and boost their number of occurrences exactly to double the amount present in the original dataset.

3 Model Architecture Training and Evaluation

We show the effectiveness of transfer learning for the task of suggestion mining by training Universal Language Model Fine-tuning for Text Classification (ULMFiT) (Howard and Ruder, 2018). One of the main advantages of training ULMFiT is that it works very well for a small dataset as provided in the Sub Task A and also avoids the process of training a classification model from scratch. This avoids overfitting. We use the fast.ai[4] implementation of this model.

The ULMFiT model has mainly two parts, the *language model* and the *classification model*. The language model is trained on a Wiki Text corpus to capture general features of the language in different layers. We fine tune the language model on the training, validation and the evaluation data. Also, we additionally scrap around two thousand reviews from the Universal Windows Platform for training our language model. After analysis of the performance we find optimal parameters to be:

- BPTT: 70, bs: 48.
- Embedding size: 400, hidden size: 1150, num of layers: 3

We also experiment with MultinomialNB, Logistic Regression, Support Vector Machines, LSTM. For LSTM we use fasttext word embeddings[5] having 300 dimensions trained on Wikipedia corpus, for representing words.

Table 3, shows the performances of all the models that we trained on the provided training dataset. We also obtained the test dataset from the organizers and evaluated our trained models on the same. The ULMFiT model achieved the best results with a F1-score of 0.861 on the training dataset and a F1-score of 0.701 on the test dataset. Table 4 shows the performance of the top 5 models for Sub Task A of SemEval 2019 Task 9. Our team ranked 10th out of 34 participants.

Model	F1 (train)	F1 (test)
Multinomial Naive Bayes (using Count Vectorizer)	0.641	0.517
Logistic Regression (using Count Vectorizer)	0.679	0.572
SVM (Linear Kernel) (using TfIdf Vectorizer)	0.695	0.576
LSTM (128 LSTM Units)	0.731	0.591
Provided Baseline	0.720	0.267
ULMFit*	0.861	0.701

Table 3: Performance of different models on the provided train and test dataset for Sub Task A.

Ranking	Team Name	Performance (F1)
1	OleNet	0.7812
2	ThisIsCompetition	0.7778
3	m_y	0.7761
4	yimmon	0.7629
5	NTUA-ISLab	0.7488
10	**MIDAS (our team)**	**0.7011***

Table 4: Best performing models for SemEval Task 9: Sub Task A.

[4]https://docs.fast.ai/text.html

[5]https://fasttext.cc/docs/en/pretrained-vectors.html

4 Error Analysis

In this section, we analyse the performance of our best model (ULMFiT) on the training data as shown by the confusion matrix presented in Figure 1. We specially look at the predictions made by our model that falls into the categories of False Positive and False Negative, as that gives us insights into the instances which our model could not classify correctly. We also present some of the instances that we found to be wrongly labeled in the provided dataset.

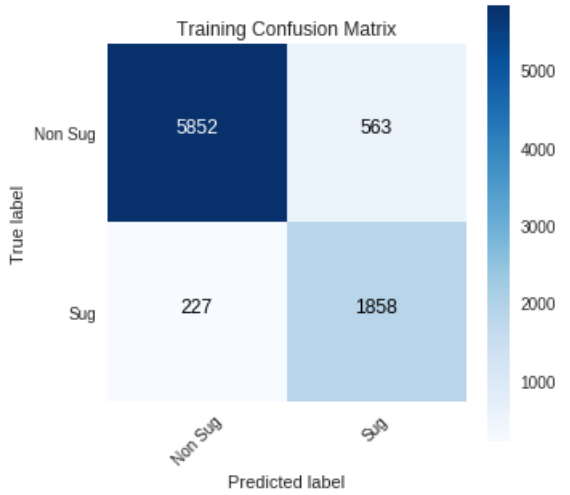

Figure 1: Confusion matrix training data

False Positives (Labeled or predicted wrongly as suggestion) Some examples that seems incorrectly labeled as suggestion in training data are given below:

- **Id 2602**: Current app extension only supports loading assets and scripts.
- **Id 3388**: One is TextCanvas for Display and Editing both Text and Inking.
- **Id 0-1747**: Unfortunately they only pull their feeds from google reader

Some examples that are incorrectly predicted by the model as suggestions are:

- **Id 1575**: That's why I'm suggesting a specialized textbox for numbers.
- **Id 1462**: If you have such limits publish them in the API docs.
- **Id 1360-2**: Adding this feature will help alot.

False Negatives (Labeled or predicted wrongly as Non Suggestion) Some examples that seems incorrectly labeled as non suggestion in the training data:

- **Id 0-1594**: Please consider adding this type of feature to feedly.
- **Id 3354**: Please support the passing of all selected files as command arguments.
- **Id 0-941**: Microsoft should provide a SDK for developers to intergate such feedback system in their Apps.

Some examples that are incorrectly predicted by the model as non-suggestions:

- **Id 0-757**: Create your own 3d library.
- **Id 834-15**: Please try again after a few minutes" in Firefox.
- **Id 4166**: I want my user to stay inside my app.

We also find that **77%** of the false positives have keywords (*want, please, add, support, would, could, should, need*), with **would** being highest i.e. around 30%.

5 Conclusion and Future Work

In this work we showed how transfer learning could be used for the task of classifying sentences extracted from unstructured text as suggestion and non-suggestions. Towards this end we train a ULMFiT model on the dataset (only Sub Task A) provided by the organizers of the SemEval 2019 Task 9 and rank 10th in the competition out of 34 participating teams.

In the future we would like to experiment and show the effectiveness of our trained model in Sub Task B where the training dataset remains the same, but the test dataset consists of suggestions from a different domain. It would be interesting to see how our model performs in predicting out-of-domain suggestions and show the ability of the ULMFiT model to fine-tune itself to a completely new domain with the already existing pre-trained model. Another interesting area would be to explore Multi Task Learning models and see how the domain of suggestion mining could get benefited by borrowing weights from models trained on other related tasks and similar tasks across different domains.

References

Nupur Baghel, Yaman Kumar, Paavini Nanda, Rajiv Ratn Shah, Debanjan Mahata, and Roger Zimmermann. 2018. Kiki kills: Identifying dangerous challenge videos from social media. *arXiv preprint arXiv:1812.00399*.

Caroline Brun and Caroline Hagege. 2013. Suggestion mining: Detecting suggestions for improvement in users' comments. *Research in Computing Science*, 70(79.7179):5379–62.

Li Dong, Furu Wei, Yajuan Duan, Xiaohua Liu, Ming Zhou, and Ke Xu. 2013. The automated acquisition of suggestions from tweets. In *Twenty-Seventh AAAI Conference on Artificial Intelligence*.

Jeremy Howard and Sebastian Ruder. 2018. Fine-tuned language models for text classification. *CoRR*, abs/1801.06146.

Hitkul Jangid, Shivangi Singhal, Rajiv Ratn Shah, and Roger Zimmermann. 2018. Aspect-based financial sentiment analysis using deep learning. In *Companion of the The Web Conference 2018 on The Web Conference 2018*, pages 1961–1966. International World Wide Web Conferences Steering Committee.

Valentin Jijkoun, Wouter Weerkamp, Maarten de Rijke, Paul Ackermans, and Gijs Geleijnse. 2010. Mining user experiences from online forums: an exploration. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Linguistics in a World of Social Media*, pages 17–18.

Raghav Kapoor, Yaman Kumar, Kshitij Rajput, Rajiv Ratn Shah, Ponnurangam Kumaraguru, and Roger Zimmermann. 2018. Mind your language: Abuse and offense detection for code-switched languages. *arXiv preprint arXiv:1809.08652*.

Debanjan Mahata, Jasper Friedrichs, Rajiv Ratn Shah, and Jing Jiang. 2018a. Detecting personal intake of medicine from twitter. *IEEE Intelligent Systems*, 33(4):87–95.

Debanjan Mahata, Jasper Friedrichs, Rajiv Ratn Shah, et al. 2018b. # phramacovigilance-exploring deep learning techniques for identifying mentions of medication intake from twitter. *arXiv preprint arXiv:1805.06375*.

Walaa Medhat, Ahmed Hassan, and Hoda Korashy. 2014. Sentiment analysis algorithms and applications: A survey. *Ain Shams engineering journal*, 5(4):1093–1113.

Mayank Meghawat, Satyendra Yadav, Debanjan Mahata, Yifang Yin, Rajiv Ratn Shah, and Roger Zimmermann. 2018. A multimodal approach to predict social media popularity. In *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 190–195. IEEE.

Sapna Negi and Paul Buitelaar. 2015. Towards the extraction of customer-to-customer suggestions from reviews. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2159–2167.

Sapna Negi, Tobias Daudert, and Paul Buitelaar. 2019. Semeval-2019 task 9: Suggestion mining from online reviews and forums. In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*.

Sapna Negi, Maarten de Rijke, and Paul Buitelaar. 2018. Open domain suggestion mining: Problem definition and datasets. *arXiv preprint arXiv:1806.02179*.

Janardhanan Ramanand, Krishna Bhavsar, and Niranjan Pedanekar. 2010. Wishful thinking: finding suggestions and'buy'wishes from product reviews. In *Proceedings of the NAACL HLT 2010 workshop on computational approaches to analysis and generation of emotion in text*, pages 54–61. Association for Computational Linguistics.

Rajiv Shah and Roger Zimmermann. 2017. *Multimodal analysis of user-generated multimedia content*. Springer.

Alfan Farizki Wicaksono and Sung-Hyon Myaeng. 2013. Automatic extraction of advice-revealing sentences foradvice mining from online forums. In *Proceedings of the seventh international conference on Knowledge capture*, pages 97–104. ACM.