




A Multi-task Approach to Open Domain Suggestion Mining Using Language Model for Text Over-Sampling

Maitree Leekha^(✉) , Mononito Goswami , and Minni Jain

Delhi Technological University, New Delhi, India
{maitreeleekha_bt2k16,mononito_bt2k16,minnijain}@dtu.ac.in

Abstract. Consumer reviews online may contain suggestions useful for improving commercial products and services. Mining suggestions is challenging due to the absence of large labeled and balanced datasets. Furthermore, most prior studies attempting to mine suggestions, have focused on a single domain such as *Hotel* or *Travel* only. In this work, we introduce a novel over-sampling technique to address the problem of class imbalance, and propose a multi-task deep learning approach for mining suggestions from multiple domains. Experimental results on a publicly available dataset show that our over-sampling technique, coupled with the multi-task framework outperforms state-of-the-art open domain suggestion mining models in terms of the F-1 measure and AUC.

Keywords: Open domain suggestion mining · Multi-task learning · Over-sampling techniques · Deep learning

1 Introduction

Consumers often express their opinions towards products and services through online reviews and discussion forums. These reviews may include useful suggestions that can help companies better understand consumer needs and improve their products and services. However, manually mining *suggestions* amid vast numbers of *non-suggestions* can be cumbersome, and equated to finding needles in a haystack. Therefore, designing systems that can automatically mine suggestions is essential. The recent *SemEval* [6] challenge on Suggestion Mining saw many researchers using different techniques to tackle the domain-specific task (*in-domain Suggestion Mining*). However, *open-domain suggestion mining*, which obviates the need for developing separate suggestion mining systems for different domains, is still an emerging research problem. We formally define the problem of open-domain suggestion mining as follows:

Definition 1 (Open-domain Suggestion Mining). *Given a set of reviews $\mathcal{R} = \{r_1, r_2 \dots r_n\}$ from multiple domains in $\mathcal{D} = d_1 \cup d_2 \cup \dots d_m$, train a*

M. Leekha, M. Goswami and M. Jain—Contributed equally and would like to be considered as joint first authors.

classifier C using \mathcal{D} to predict the nature $n_i \in \{\text{'suggestion'}, \text{'non-suggestion'}\}$ of each review r_i .

Building on the work of [5], we design a framework to detect suggestions from multiple domains. We formulate a multitask classification problem to identify both the domain and nature (*suggestion* or *non-suggestion*) of reviews. Furthermore, we also propose a novel language model-based text over-sampling approach to address the class imbalance problem.

2 Methodology

2.1 Dataset and Pre-processing

We use the first publicly available and annotated dataset for suggestion mining from multiple domains created by [5]. It comprises of reviews from four domains namely, **hotel**, **electronics**, **travel** and **software**. During pre-processing, we remove all URLs (eg. *https:// ...*) and punctuation marks, convert the reviews to lower case and lemmatize them. We also pad the text with start **S** and end **E** symbols for over-sampling.

2.2 Over-Sampling Using Language Model: LMOTE

One of the major challenges in mining suggestions is the imbalanced distribution of classes, *i.e.* the number of non-suggestions greatly outweigh the number of suggestions (refer Table 1). To this end, studies frequently utilize *Synthetic Minority Over-sampling Technique* (SMOTE) [1] to over-sample the minority class samples using the text embeddings as features. However, SMOTE works in

Table 1. Datasets and their sources used in our study [5]. The class ratio column highlights the extent of class imbalance in the datasets. The **travel** datasets have lower inter-annotator agreement than the rest, indicating that they may contain confusing reviews which are hard to confidently classify as suggestions or non-suggestions. This also reflects in our classification results.

Dataset	Source	Class ratio (suggestion: non-suggestion)	Inter-annotator agreement
Hotel train	Tripadvisor	448/7086 \approx 6:100	0.86
Hotel test	Tripadvisor	404/3000 \approx 13:100	0.86
Electronics train	Amazon	324/3458 \approx 9:100	0.83
Electronics test	Amazon	101/1070 \approx 9:100	0.83
Travel train	Insight vacations, Fodors	1314/3869 \approx 34:100	0.72
Travel test	Fodors	229/871 \approx 26:100	0.72
Software train	Uservice suggestion forum	1428/4296 \approx 33:100	0.81
Software test	Uservice suggestion forum	296/742 \approx 39:100	0.81

Table 2. Most frequent 5-grams and their corresponding suggestions sampled using LMOTE. While the suggestions as a whole may not be grammatically correct, their constituent phrases are nevertheless semantically sensible.

Domain	Most frequent 5-gram	A suggestion sampled using LMOTE
Hotel	I would definitely recommend hotel	I would definitely recommend hotel good value full ocean view great food worth
Electronics	Suggestion get lens protector help	Suggestion get lens protector help protect long lens coating uv 52 lens last long must try
Travel	Tipping remember shape luggage concerned	Tipping remember shape luggage concerned heavy luggage rough advised wheeled duffle wont heavy
Software	It would be good if oversight	It would be good if oversight bixby developed bug feels wide back content zoom should be an option

the euclidean space and therefore does not allow an intuitive understanding and representation of the over-sampled data, which is essential for qualitative and error analysis of the classification models. We introduce a novel over-sampling technique, **Language Model-based Over-sampling Technique (LMOTE)**, exclusively for text data and note comparable (and even slightly better sometimes) performance to SMOTE. We use LMOTE to over-sample the number of suggestions before training our classification model. For each domain, LMOTE uses the following procedure to over-sample suggestions:

Find Top η n-Grams: From all reviews labelled as suggestions (positive samples), sample the top $\eta = 100$ most frequently occurring n-grams ($n = 5$). For example, the phrase “*nice to be able to*” occurred frequently in many domains.

Train Language Model on Positive Samples: Train a BiLSTM language model on the positive samples (suggestions). The BiLSTM model predicts the probability distribution of the next word (w_t) over the whole vocabulary ($V \cup E$) based on the last $n = 5$ words (w_{t-5}, \dots, w_{t-1}), *i.e.*, the model learns to predict the probability distribution $P(w_i | w_{t-5} w_{t-4} w_{t-3} w_{t-2} w_{t-1}) \forall i \in (V \cup E)$, such that $w_t = \arg \max_{w_i} P(w_i | w_{t-5} w_{t-4} w_{t-3} w_{t-2} w_{t-1})$.

Generate Synthetic Text Using Language Model and Frequent n-Grams: Using the language model and a randomly chosen frequent 5-gram as the seed, we generate text by repeatedly predicting the most probable next word (w_t), until the end symbol **E** is predicted.

Table 2 comprises of the most frequent 5-grams and their corresponding suggestions ‘sampled’ using LMOTE. In our study, we generate synthetic positive reviews till the number of suggestion and non-suggestion class samples becomes equal in the training set.

Algorithm 1. Language Model-based Over-sampling Technique (LMOTE)

Input: $\mathcal{D}_{sugg} = \{r_i \in \mathcal{D} \mid n_i = \text{'suggestion'}\}$ Suggestions from a particular domain \mathcal{D} ; η - Number n -grams to use in LMOTE; n - type of n -grams *e.g.* 2 for bi-grams, etc.; \mathcal{N} - Number of suggestion samples required.**Output:** $\mathcal{S} : \mathcal{N}$ over-sampled suggestions

```

1:  $n\_grams \leftarrow \mathbf{NGrams}(\mathcal{D}_{sugg}, \eta, n)$ 
2:  $language\_model \leftarrow \mathbf{TrainLanguageModel}(\mathcal{D}_{sugg}, n)$ 
3: Initialize  $\mathcal{S} \leftarrow \mathcal{D}_{sugg}$ 
4: while  $|\mathcal{S}| < \mathcal{N}$  do
5:    $seed \leftarrow \mathbf{random}(n\_grams)$ 
6:    $sample \leftarrow \mathbf{LMOTEGenerate}(language\_model, seed)$ 
7:    $\mathcal{S} \leftarrow \mathcal{S} \cup sample$ 
8: end while
9: return  $\mathcal{S}$ 

```

Algorithm 1 summarizes the LMOTE over-sampling methodology. Following is a brief description of the sub-procedures used in the algorithm:

- **NGrams**($\mathcal{D}_{sugg}, \eta, n$): It returns the top η n -grams from the set of suggestions, \mathcal{D}_{sugg} .
- **TrainLanguageModel**(\mathcal{D}_{sugg}, n): This procedure trains an n -gram BiLSTM Language Model on \mathcal{D}_{sugg} .
- **random**(n_grams)- Randomly selects an n -gram from the input set.
- **LMOTEGenerate**($language_model, seed$): The procedure takes as input the trained language model and a randomly chosen n -gram from the set of top η n -grams as $seed$, and starts generating a review till the end tag, \mathbf{E} is produced. The procedure is repeated until we have a total of \mathcal{N} suggestion reviews.

2.3 Mining Suggestion Using Multi-task Learning

Multi-task learning (MTL) has been successful in many applications of machine learning since sharing representations between auxiliary tasks allows models to generalize better on the primary task. Figure 1B illustrates 3-dimensional UMAP [4] visualization of *text embeddings* of suggestions, coloured by their domain. These embeddings are outputs of the penultimate layer (dense layer before the final softmax layer) of the *Single task* (STL) ensemble baseline. It can be clearly seen that suggestions from different domains may have varying feature representations. Therefore, we hypothesize that we can identify suggestions better by leveraging domain-specific information using MTL. Therefore, in the MTL setting, given a review r_i in the dataset, \mathcal{D} , we aim to identify both the domain of the review, as well as its nature.

2.4 Classification Model

We use an ensemble of three architectures namely, CNN [2] to mirror the spatial perspective and preserve the n -gram representations; Attention Network to learn

the most important features automatically; and a BiLSTM-based text RCNN [3] model to capture the context of a text sequence (Fig. 2). In the MTL setting, the ensemble has two output softmax layers, to predict the domain and nature of a review. The STL baselines on the contrary, only have a single softmax layer to predict the nature of the review. We use ELMo [7] word embeddings trained on the dataset, as input to the models.

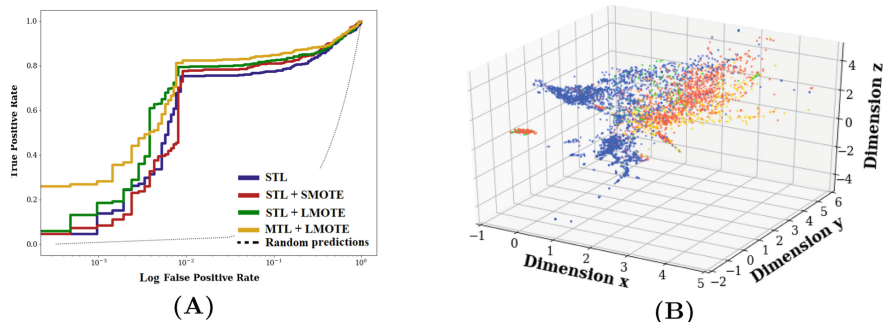


Fig. 1. (A) Receiver operating characteristics (TPR vs. Log FPR) curve pooled across all domains for all models used in this work demonstrates that LMOTE coupled with our multi-task model outperforms other considered alternatives across domains (B) 3-dimensional UMAP visualization of text embeddings of *suggestions* coloured by domain. Suggestions from different domains have distinct feature representations.

3 Results and Discussion

We conducted experiments to assess the impact of over-sampling, the performance of LMOTE and the multi-task model. We used the same train-test split as provided in the dataset for our experiments. All comparisons have been made in terms of the F-1 score of the suggestion class for a fair comparison with prior work on representational learning for open domain suggestion mining [5] (refer *Baseline* in Table 3). For a more insightful evaluation, we also compute the Area under Receiver Operating Characteristic (ROC) curves for all models used in this work. Tables 3, 4 and Figs. 3 and 1A summarize the results of our experiments, and there are several interesting findings:

Over-Sampling Improves Performance. To examine the impact of over-sampling, we compared the performance of our ensemble classifier with and without over-sampling *i.e.* we compared results under the *STL*, *STL + SMOTE* and *STL + LMOTE* columns. Our results confirm that in general, over-sampling suggestions to obtain a balanced dataset improves the performance (F-1 score & AUC) of our classifiers.

LMOTE Performs Comparably to SMOTE. We compared the performance of SMOTE and LMOTE in the single task settings (*STL + SMOTE*

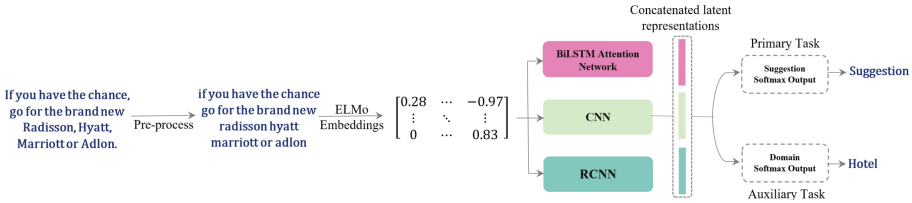


Fig. 2. Our multi-task classification model which consists of an ensemble of RCNN, CNN and BiLSTM attention network. The primary task is predicting the nature of a review (suggestion), while the auxiliary task involves predicting its domain (**hotel**).

and *STL + LMOTE*) and found that LMOTE performs comparably to SMOTE (and even outperforms it in the **electronics** and **software** domains). LMOTE also has the added advantage of resulting in intelligible samples which can be used to qualitatively analyze and troubleshoot deep learning based systems. For instance, consider suggestions *created* by LMOTE in Table 2. While the suggestions may not be grammatically correct, their constituent phrases are nevertheless semantically sensible.

Table 3. Performance evaluation using F-1 score. Multi-task learning with LMOTE outperforms other alternatives in open-domain suggestion mining. Furthermore, owing to potentially confusing reviews in the travel domain (Table 1), its F-1 scores are significantly lower than the other domains.

Domain	Baseline	STL	STL+SMOTE	STL+LMOTE	MTL+LMOTE
Hotel	0.77 (LSTM)	0.79	0.83	0.83	0.86
Electronics	0.78 (SVM)	0.80	0.80	0.83	0.83
Travel	0.66 (SVM)	0.65	0.68	0.69	0.71
Software	0.80 (LSTM)	0.79	0.81	0.84	0.88

Table 4. Performance evaluation using area under ROC with 95% confidence intervals. Multi-task learning with LMOTE outperforms other alternatives in open-domain suggestion mining. Multi-task learning leads to a significant improvement in AUC over its single task counterpart. (AUCs for baseline models proposed by [5] were unavailable.)

Domain	STL	STL+SMOTE	STL+LMOTE	MTL+LMOTE
Hotel	0.878 ± 0.022	0.897 ± 0.021	0.828 ± 0.025	0.894 ± 0.012
Electronics	0.897 ± 0.041	0.92 ± 0.037	0.912 ± 0.037	0.944 ± 0.031
Travel	0.828 ± 0.034	0.848 ± 0.033	0.835 ± 0.025	0.852 ± 0.032
Software	0.894 ± 0.025	0.893 ± 0.025	0.919 ± 0.022	0.956 ± 0.015
Pooled AUC	0.876 ± 0.014	0.883 ± 0.013	0.897 ± 0.012	0.907 ± 0.012

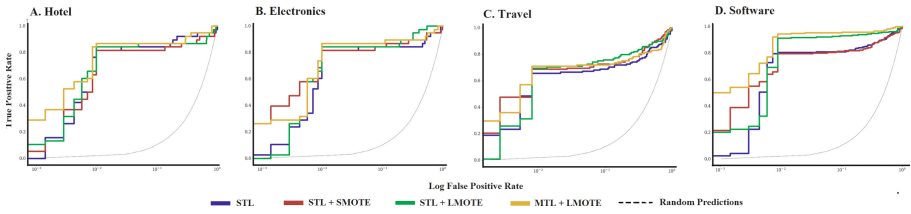


Fig. 3. Domain wise receiver operating characteristics (ROC) curves.

Multi-task Learning Outperforms Single-Task Learning. We compared the performance of our classifier in single and multi-task settings (*STL + LMOTE* and *MTL + LMOTE*) and found that by multi-task learning improves the performance of our classifier. We qualitatively analysed the single and multi task models, and found many instances where by leveraging domain-specific information the multi task model was able to accurately identify suggestions. For instance, consider the following review: “*Bring a Lan cable and charger for your laptop because house-keeping doesn’t provide it.*” While the review appears to be an assertion (*non-suggestion*), by predicting its domain (*hotel*), the multi-task model was able to accurately classify it as a suggestion.

4 Conclusion

In this work, we proposed a Multi-task learning framework for Open Domain Suggestion Mining along with a novel language model based over-sampling technique for text-LMOTE. Our experiments revealed that Multi-task learning combined with LMOTE over-sampling outperformed considered alternatives in terms of both the F1-score of the suggestion class and AUC.

References

1. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: SMOTE: synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **16**, 321–357 (2002)
2. Kim, Y.: Convolutional neural networks for sentence classification. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (2014). <https://doi.org/10.3115/v1/d14-1181>
3. Lai, S., Xu, L., Liu, K., Zhao, J.: Recurrent convolutional neural networks for text classification. In: *Twenty-Ninth AAAI Conference on Artificial Intelligence* (2015)
4. McInnes, L., Healy, J., Melville, J.: UMAP: uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426* (2018)
5. Negi, S.: Suggestion mining from text. Ph.D. thesis, National University of Ireland Galway (NUIG) (2019)
6. Negi, S., Daudert, T., Buitelaar, P.: SemEval-2019 task 9: suggestion mining from online reviews and forums. In: *SemEval@NAACL-HLT* (2019)
7. Peters, M.E., et al.: Deep contextualized word representations. *arXiv preprint arXiv:1802.05365* (2018)