A Multi-Task Approach to Open Domain Suggestion Mining (Student Abstract)

Minni Jain,[*†] **Maitree Leekha,**[*] **Mononito Goswami**[*]
Department of Computer Science & Engineering,
Delhi Technological University, New Delhi, India
{minnijain, maitreeleekha_bt2k16, mononito_bt2k16}@dtu.ac.in

Abstract

Consumer reviews online may contain suggestions useful for improving the target products and services. Mining suggestions is challenging because the field lacks large labelled and balanced datasets. Furthermore, most prior studies have only focused on mining suggestions in a single domain. In this work, we introduce a novel up-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 up-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.

1 Introduction

Consumers often express their opinions towards products and services through online reviews and discussion forums. These reviews may include useful suggestions which 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 which can automatically mine suggestions is essential. The recent *SemEval* challenge on Suggestion Mining saw many researchers using different techniques to tackle the domain specific task. However, *open-domain suggestion mining*, which obviates the need for developing separate suggestion mining systems for different domains, is still an emerging research problem. Building on the work of (Negi 2019), we design a framework to detect suggestions from multiple domains. Specifically, 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 up-sampling approach to address the class imbalance problem.

[*]All the authors contributed equally, and wish that they be regarded as joint First Authors.

[†]Mobile Number: +91 8587898334

2 Methodology

Dataset & Pre-processing

We use the first publicly available and annotated dataset for suggestion mining from multiple domains created by (Negi 2019). 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 up-sampling.

Up-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. To this end, studies frequently utilize *Synthetic Minority Over-sampling Technique* (SMOTE) (Chawla et al. 2002) to up-sample the minority class samples using the text embeddings as features. However, SMOTE works in 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, **L**anguage **M**odel-based **O**ver-sampling **Te**chnique (LMOTE), exclusively for text data and note comparable (and even slightly better sometimes) performance to SMOTE. We use LMOTE to up-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 η=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 ($w_{t-5}, ..., w_{t-1}$), *i.e.*, the model learns to predict the probability distribution $P(w_i \mid w_{t-5}\ w_{t-4}\ w_{t-3}\ w_{t-2}\ w_{t-1})\ \forall i \in (V \cup \mathbf{E})$, such that $w_t = \arg\max_{w_i} P(w_i \mid w_{t-5}\ w_{t-4}\ w_{t-3}\ w_{t-2}\ w_{t-1})$.

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 till the end symbol **E** is predicted.

Mining Suggestion using Multi-task Learning

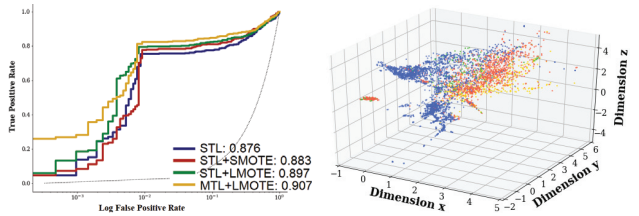
Multi-task learning (MTL) has been successful in many applications of machine learning since sharing representations between auxiliary tasks allows models to generalize better on the primary task. Figure 1B illustrates 3-dimensional Uniform Manifold Approximation and Projection (UMAP) visualization of *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, D, we aim to identify both the domain of the review, as well as its nature.

Classification Model

We use an ensemble of three architectures namely, CNN 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 model to capture the context of a text sequence. 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 singe softmax layer to predict the nature of the review. We use ELMo (Peters et al. 2018) word embeddings trained on the dataset, as input to the models.

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**
Pooled AUC		0.876 ±0.014	0.883 ±0.013	0.897 ±0.012	**0.907 ±0.012**

Table 1: Performance evaluation using F-1 score & pooled Area under ROC curve (AUC) with 95% confidence intervals. Multi-task Learning with LMOTE outperforms other alternatives in open-domain suggestion mining.



(A) *(B)*

Figure 1: ***(A)*** ROC (TPR vs. Log FPR) curves pooled across all domains for all models used in this work. ***(B)*** 3-dimensional UMAP visualization of text embeddings of *suggestions* coloured by domain.

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 (Negi 2019) (refer *Baseline* in Table 1). For more insightful evaluation, we also find the Pooled Area Under the Receiver Operating Characteristic (ROC) curves for all models used in this work. Table 1 summarizes 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* 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 the following sample *created* by LMOTE: *"It would be good if oversight bixby developed bug feels wide back content zoom should be an option."* While the suggestion may not be grammatically correct, its constituent phrases are nevertheless semantically sensible.

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.

References

Chawla, N. V.; Bowyer, K. W.; Hall, L. O.; and Kegelmeyer, W. P. 2002. Smote: Synthetic minority over-sampling technique. *Journal of artificial intelligence research* 16:321–357.

Negi, S. 2019. *Suggestion Mining from Text*. Ph.D. Dissertation, National University of Ireland Galway (NUIG).

Peters, M. E.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; and Zettlemoyer, L. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.