Helping each Other: A Framework for Customer-to-Customer Suggestion Mining using a Semi-supervised Deep Neural Network

Hitesh Golchha, Deepak Gupta, Asif Ekbal, Pushpak Bhattacharyya
Indian Institute of Technology Patna
Patna, India
{hitesh.me14, deepak.pcs16, asif, pb}@iitp.ac.in

Abstract

Suggestion mining is increasingly becoming an important task along with sentiment analysis. In today's cyberspace world, people not only express their sentiments and dispositions towards some entities or services, but they also spend considerable time sharing their experiences and advice to fellow customers and the product/service providers with two-fold agenda: helping fellow customers who are likely to share a similar experience, and motivating the producer to bring specific changes in their offerings which would be more appreciated by the customers. In our current work, we propose a hybrid deep learning model to identify whether a review text contains any suggestion. The model employs semi-supervised learning to leverage the useful information from the large amount of unlabeled data. We evaluate the performance of our proposed model on a benchmark customer review dataset, comprising of the reviews of *Hotel* and *Electronics* domains. Our proposed approach shows the F-scores of 65.6% and 65.5% for the Hotel and Electronics review datasets, respectively. These performances are significantly better compared to the existing state-of-the-art system.

1 Introduction

The online platforms like social media websites, e-commerce sites of products and services, blogs, online forums and discussion forums etc. are very much attached today with our day-to-day lives.The availability of the these information sharing platforms has fueled the humans' desires to share one's opinions, emotions and sentiments

with respect to the entities of all kinds: be it people, events, places, organizations, institutions, products, services, hobbies, games, movies, politics, technology etc. Generally people express their opinions in three ways: **(1)** through an independent piece of content writing **(2)** writing disposed towards a theme (such as a question in a community based question answering platform, or a topic in a discussion forum, or an entity in a product reviewing website/ e-commerce website) and **(3)** conversational writings in the form of exchange of utterances in dialog systems/chats or comments for a post in social media/online forums.

Such opinions which exist in different forms and places, have often hidden in them the experiences of people, their subjective emotions and sentiments towards different aspects of different entities, as well as the intentions of advices and suggestions proposing some action in a prescribed way. Suggestion mining can be thought of as a subproblem of opinion mining, entrusted with the task of extracting mentions of suggestions from the unstructured texts. Suggestions in the domain of reviews can be generally of two kinds:

1. **Customer to Companies:** These suggestions are directed from customers to the producers/service providers. Customers provide companies with feedbacks, often expressing their contentment or complaining about their dissatisfaction with certain product features, services, processes or amenities. They provide detailed reasons and personal experiences for the same and offer alternative ideas for implementation. These kinds of suggestions are not only important as a tool for the companies to review their current offerings, but they are also a great source of ideas for new directions.

183

2. **Customer to Customer:** These suggestions are provided from customers/users to the fellow customers/users. Customers share their experiences in reviews, and provide tips and recommendations to the other customers. This is sometimes more than merely the information whether they like some specific attributes of the products or services.

1.1 Motivation and Contributions

There are several use cases of automated retrieval and natural language understanding for suggestion mining. Apart from their own experiences, understanding and knowledge, people depend on the online community to form their own opinions and readily look for suggestions and tips from the other customers. The extracted suggestions and tips are equivalent to a set of effective guidelines for the other customers before they make their own decisions. The fellow users can avail more information, and hence the decision taken would be better. This is often beyond the sense conveyed by aspect based sentiment analysis (Thet et al., 2010; Gupta et al., 2015; Gupta and Ekbal, 2014).

Suggestions and feedbacks are also an important component of the market survey performed by the companies to drive innovation, change and improvements. This task is a prerequisite to other nuanced tasks which include classifying the domain of the suggestion, identifying the other arguments of the suggestions (finding the entity towards whom the suggestion is directed, identifying the aspects regarding which a suggestion has been made, finding the word boundaries of the suggestive expressions), and aggregation of such suggestions from multiple sources to comprehend a customer friendly summary.

We summarize the contributions of our proposed work as follows:

- We develop a linguistically motivated hybrid neural architecture to identify the review sentences that carry an intention of suggestion.
- We employ semi-supervised learning (self-training) along with a deep learning based supervised classification approach. This gives us the opportunity to harness the treasure of huge (unlabeled) data available in the form of customer reviews. To the best of our knowledge, this is the very first attempt in this direction to handle the target problem.
- Outperforming the current state-of-the-art

customer-to- customer suggestion mining techniques and setting up a new state-of-the-art.

2 Related works

The field of suggestion classification and customer feedback analysis are relatively new in the area of Natural Language Processing (NLP) and Text Mining. Our work is most closely related to the prior research as reported in (Negi and Buitelaar, 2015; Negi et al., 2016). In (Negi and Buitelaar, 2015) authors defined the annotation guidelines for customer-to-customer suggestion mining. They trained a support vector machine (SVM) classifier over the features relevant for classification in the domains of hotels and electronics reviews. They used heuristic features, features extracted from the Part-of-Speech (PoS) tags, sequential pattern mining features, sentiment features and the features extracted from the dependency relations.

In their subsequent work, (Negi et al., 2016) demonstrated the improved performance using Convolutional neural networks (CNN) (Kim, 2014) and Long short term memory (LSTM) (Hochreiter and Schmidhuber, 1997) based deep learning architectures to solve this problem. They experimented with both in-domain and cross-domain training data, and also compared their performance with a SVM based classifier trained with the same set of features similar to (Negi and Buitelaar, 2015).

There are some other existing works for suggestion mining, beyond customer-to-customer suggestions. (Ngo et al., 2017) developed a binary classification model based on Maximum Entropy and CNN for filtering suggestion intents in Vietnamese conversational texts like posts, comments, reviews, messages chat and spoken texts.

Brun and Hagege (Brun and Hagege, 2013) developed a feature-based suggestion mining system for the domain of product reviews. (Dong et al., 2013) performed suggestion mining on tweets of the customers regarding Microsoft Windows' phone. A model is proposed in (Wicaksono and Myaeng, 2013) which focused on extracting advices for the domains of travel using Hidden Markov Model (HMM) and Conditional Random Field (CRF). The work as reported in (Gupta et al., 2017) focused on classifying the customer feedback sentences of users into six classes using deep

learning based models.

Our proposed model differs from these existing works with respect to the problem addressed and the model developed. We have presented a very detailed comparison (in the experiments section) to the state-of-the-art system as reported in (Negi and Buitelaar, 2015; Negi et al., 2016).

3 Methodology

In this section at first we discuss the various deep learning models and then semi-supervised model.

3.1 Problem Definition

Given a multi-sentence review R having N sentences $\{s_1, s_2, \ldots, s_N\}$ the task is to categorize each sentence s_i into one of the classes $c \in C$, where C = {*"suggestive"*, *"non-suggestive"*}. For a sentence s with a sequence $w_1, w_2, \ldots, w_{n-1}, w_n$ of n words, the associated suggestion class c can be computed as:

$$c = argmax_y p(y|s) \tag{1}$$

where $y \in C$

The following example review sentence, *"Tip if you want a beach chair at the beach or pool, go there before 9 am or so and put your magazine or towel on your chair."* is a "suggestion" intent directed towards a fellow customer. Here the expression of the intent is explicitly conveyed in the form of a review sentence with imperative mood[1]. The "non-suggestive" sentences instead contain statements and facts (e.g. **(1)** *"We stayed in the Westin Grand Berlin in July 2007."*) or expressions of one's sentiments (e.g. **(2)** *"But the rooms are small and not very functional.")*. An interesting thing to note is that the second example has implicit suggestions for the fellow customers as well as the service provider (hotel owner). The other visiting customers are implicitly advised against renting the rooms of the hotel as they are small and have less utility. Moreover, this review sentence also consists of an implicit suggestion to the hotel owner to offer larger rooms to their customers, and also improve the functionalities that they provide. However in our work, we only deal with the suggestions which are very explicitly mentioned, and that too directed specifically to the fellow customers.

3.2 Proposed Deep Learning Model

The customer-to-customer suggestion mining task requires recognizing specific syntactic and semantic constructions represented in texts. It should be able to capture the constructions representing imperative moods, and identify the patterns or phrases which are highly correlated with suggestive sentences in a review. It should also have a way for deep semantic understanding of text in order to disambiguate suggestions from the sentences which appear like suggestions on the surface.

We propose a hybrid model consisting of two deep learning based encoders designed to integrate different views or representations of the review sentences, and a linguistically motivated feature set. The information from the encoders along with linguistic knowledge are effectively combined with the help of a multi-layer perception (MLP) network. This is done to achieve higher abstraction necessary for a complex task like identifying the suggestive review sentence. Specifically, we use two different encoders, namely Convolutional Neural Network (CNN) and attention based Recurrent Neural Network (RNN). The effectiveness of CNN and RNN based encoder has been proven in other NLP tasks (Gupta et al., 2018d; Maitra et al., 2018; Gupta et al., 2018a,c). The CNN encoder uses multiple fully-connected over the convolution layer while the RNN encoder uses a LSTM layer with the attention (Raffel and Ellis, 2015) followed by multiple fully-connected layers. An overview of the architecture for suggestion mining is shown in Figure 1.

3.2.1 Linguistic Features

We use the following set of linguistic features in our model. We use slightly modified subset of features from (Negi and Buitelaar, 2015) and similar to (Gupta et al., 2018b)

Suggestive keywords : The suggestive keywords are usually associated with the texts containing actual suggestions. We use the following small set of suggestive keywords:
advice, suggest, may, suggestion, ask, warn, recommend, do, advise, request, warning, tip, recommendation, not, should, can, would, will
A binary-valued feature is defined that checks

185

---

[1]Imperative mood is a category or form of a verb which expresses a request or a command. For example, *"Get ready"*
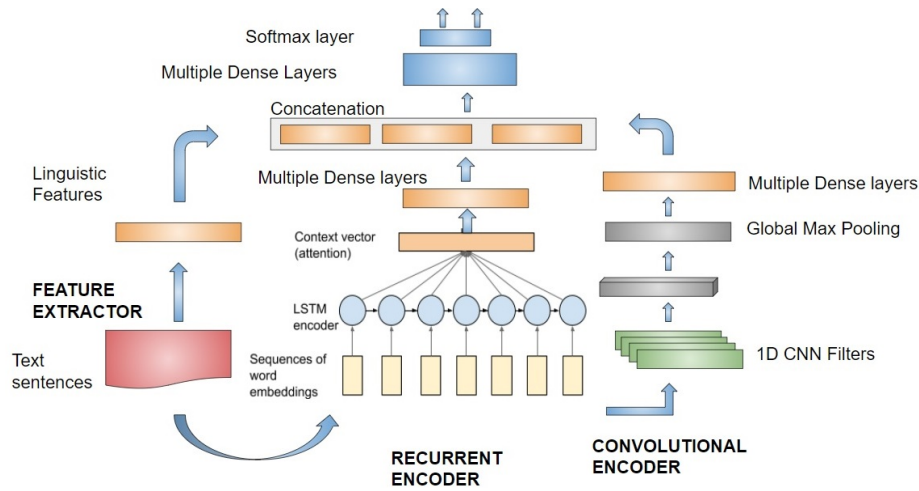
Figure 1: The proposed model architecture for customer suggestion mining

whether the current word is one of the keywords or not (1-presence, 0-absence).

N-gram features : We extract the most frequent 300 unigrams, 100 bigrams, and 100 trigrams from the training set. These are then used as a bag of n-gram features.

Part-of-Speech (PoS) N-gram features : We extract the most frequent PoS unigrams, bigrams and trigrams of size 50. These are then used as a bag of PoS n-grams features.

Imperative mood features : Most of the suggestions containing sentences have imperative mood. We try to capture this phenomenon by introducing the features obtained from the dependency trees[2]. We use the following imperative mood features:

1. Base verb (VB) at the beginning of sentence or without nsubj arc: In many imperative sentences, the subject (denoted by nsubj) is absent, i.e. it implies to be the second person. Moreover, the clause containing the suggestive expression begins with the base form of the verb (denoted by VB). Hence, this does not have any dependency relation with nsubj. This feature is illustrated in Figure 2.

2. 'nsubj' dependency relation features: The pair of PoS tags of the words connected by the dependency arc 'nsubj' is used as the bag of PoS feature. We describe the presence of this feature in Figure 3 and 4.
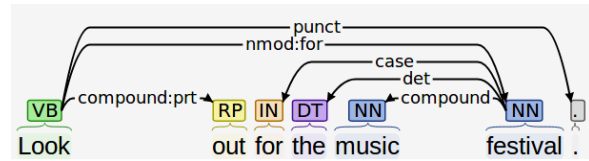

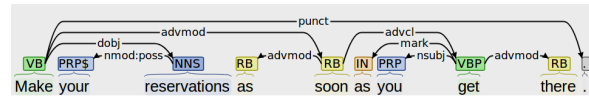Figure 2: Presence of 'VB' without nsubj arc

Figure 3: nsubj dependency arc relations. From this dependency tree the extracted features are (VBP, PRP).

This set of linguistic features are fed into a multi-layer perceptron having two hidden layers of size 150 and 25, respectively.

3.2.2 Recurrent and CNN Encoders

The words in the sequence $\{w_1, w_2 \ldots w_n\}$ from a given review sentence s are mapped to their corresponding word vectors $\{x_1, x_2 \ldots x_n\}$. The word embeddings are obtained through the publicly available[3] GloVe word embeddings(Pennington et al., 2014) of dimension 300 and trained on the Common Crawl.

The recurrent encoder uses a LSTM network (hidden size 64) over the embedded sequences and it then applies an internal attention over the hidden states.

The LSTM network is able to process the sentence as a sequence, with the ability to capture long term dependencies. Thus the hidden layers

---

[2]We use spaCy dependency parser. For visualization, we used Stanford dependency parser

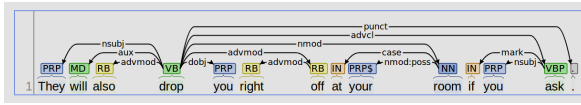[3]https://nlp.stanford.edu/projects/glove/

Figure 4: nsubj dependency arc relations. Here, (VB, PRP) and (VBP, PRP) features are active.

can efficiently perform composition over the local context, and help to identify patterns which are found in suggestive sentences. The attention mechanism then finds salient contexts and aggregates the important ones to build the context vector. The motivation for using attention stems from the fact that suggestive expressions can be identified in a short span of text within the sentence and the attention can effectively attend to those specific contexts encoded by LSTM. The Attention layer is followed by dense layers with 150, and 25 neurons, each having ReLU activations and a dropout value of 0.2.

The convolutional layer applies 250 one dimensional CNN filters of size 5 over the embeddings. The global max pooling is applied separately for the feature map obtained from each filter, and it helps to identify the presence of the n-gram feature corresponding to that feature in the sentence. The following dense layer with 250 neurons (ReLU activation and 0.75 dropout) helps to non-linearly compose multiple such features, thus giving itself an opportunity to learn a more diverse set of features.

3.2.3 Hybrid Model

The extracted linguistic features, the recurrent encoder representation and the convolutional encoder representation are concatenated (into a feature set p) and fed to a fully-connected layer with two neurons, followed by softmax activation. The softmax layer outputs the probability of the given review sentence being *suggestive* or *non-suggestive*. The probability that the output class $\hat{y}$ is i given the sentence s and parameters θ is computed as:

$$P(\hat{y} = i|s, \theta) = softmax_i(p^{\mathrm{T}}w_i + z_i)$$
$$= \frac{e^{p^T w_i + z_i}}{\sum_{k=1}^{K} e^{p^T w_k + z_k}} \quad (2)$$

where z_k and w_k are the bias and weight vector of the kth labels, p is the concatenated feature set, and K is the number of total classes (i.e. 2). θ is the

set of all the parameters of the model. The system predicts the most probable class.

3.3 Semi-supervised Model

Semi-supervised learning makes use of both labeled (small) and unlabeled (huge) data for designing a more efficient classifier, as compared to the traditional supervised learning. We utilize self-training algorithm (Zhu, 2006), also known as bootstrapping, which can be flexibly used as a wrapper over any supervised learning algorithm. We use our hybrid model for this semi-supervised learning.

In self-training, we iteratively train a classifier enhancing each time the original training dataset with newly labeled instances. At the end of each iteration, the classifier is made to predict on the unlabeled dataset and 100 most confidently predicted instances of each class is added to the training data, with the predicted labels as the true labels. For self-training, a methodology similar to early stopping is applied, with a maximum of six iterations. We stop the iteration when the F1-Score on the validation data [4] does not improve over the existing best model in consecutive three iterations, saving only the best performing model for testing. For example in Fig. 5, the training terminates after the 6th iteration, and the model trained in the 3rd iteration is chosen for the final evaluation. Effect of adding unlabeled data to training for the electronics domain is depicted in Figure 6.
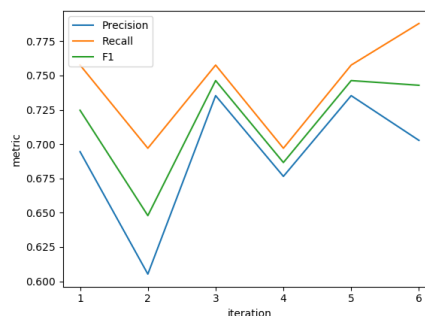

Figure 5: Scores on the validation set during self-training: Hotel domain.

For this semi-supervised setting, the cross-entropy error is minimized using the Adam Optimizer, and the training is stopped as the validation loss stops decreasing (early stopping). Because of the class imbalance (cf. Table 1), the loss function

---

187

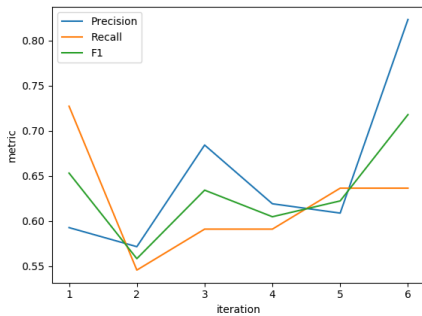[4]A part of the training set was used for validation.

Figure 6: Scores on the validation set during self-training: Electronics domain.

weighs the loss for the positive class instances 10 times more than the loss for the negative class instances. All the other configurations are similar to the supervised setting[5].

4 Dataset and Experiments

In this section we present the datasets, experimental results and the necessary analysis.

4.1 Dataset

We conduct experiments on the dataset created by (Negi and Buitelaar, 2015)). The dataset comprises of the sentences of reviews taken from two domains, *viz.* Hotels and Electronics. The dataset was annotated as 'suggestive' and 'non-suggestive'.

The hotel reviews in (Negi and Buitelaar, 2015) are a subset of the TripAdvisor reviews annotated by (Wachsmuth et al., 2014)), with the sentiment polarity classes of positive, negative, neutral and conflicting. The electronics dataset was originally annotated by (Hu and Liu, 2004)) with the sentiment labels, and (Negi and Buitelaar, 2015) extended it for suggestion mining. The dataset consists of 7534 sentences from the hotel reviews and 3782 sentences from the reviews of electronic items. For semi-supervised learning experiments, we obtain the complete dataset from (Wachsmuth et al., 2014) for hotels. We segment these reviews into 21328 sentences in total. For the electronics domain, we use the Amazon reviews obtained from the electronics segment of (He and McAuley, 2016) as the unlabeled data. The first 50,000 sentences extracted from the reviews were chosen for the experiments.

	Hotel reviews	Electronics reviews
Class = 1 *Suggestive*	407	273
Class = 0 *Non - Suggestive*	7127	3509
Total	7534	3782
Class1 : Class0	1:17.5	1:12.9

Table 1: Dataset statistics (on the sentence level)

Instances of suggestions and tips form a relatively small percentage of the total review sentences, and this is reflected in the class distribution of the labeled dataset. The number of instances is not enough for very deep architectures. Statistics of the datasets are presented in Table 1.

4.2 Results and Analysis

We re-implement the LSTM and CNN architectures proposed in (Negi et al., 2016) to construct our baselines. We re-implement this state-of-the-art system with the common training methodologies as ours. Detailed evaluation results are demonstrated in Table 2.

The LSTM is capable of handling long term dependencies and that may be attributed to its better performance against CNN for the domain of electronics where the average sentence length is relatively longer. The model based on LSTM achieves the F1 scores of 0.562 and 0.611 for the hotel and electronics datasets, respectively. CNN based model also demonstrates comparative performance with F1 scores of 0.598 and 0.600 for the two domains, respectively. Introducing attention to the LSTM model was found to be effective with reasonable performance improvement. Because of attention the system could attend to specific regions of the input sentence which had patterns similar to that of suggestive sentences, encoded by its query vector. The system with attention shows the best recall of 76.9% for the hotel reviews and 69.9% for the electronics reviews, establishing our claim about its ability. It achieves the F1 scores of 0.602 and 0.611 for the two domains, respectively.

Among the different architectures, the proposed hybrid model is found to be the best performing one with F1 scores of 0.643 and 0.621 for the two domains, respectively. Each of the encoders provides a different representation and features of the input, and the dense layers are able to combine them in an effective way. We also remove different encoders-one after other-from the proposed hybrid

---

[5]Models are optimized based on the validation set, a part of the training set

188

Model		Hotel			Electronics	
	Precision	Recall	F1	Precision	Recall	F1
CNN	0.560	0.641	0.598	0.586	0.615	0.600
LSTM	0.511	0.624	0.562	0.582	0.644	0.611
LSTM + Attention	0.494	**0.769**	0.602	0.543	**0.699**	0.611
Negi and Buitelaar (2015)	0.580	0.512	0.567	**0.645**	0.621	0.640
Proposed Hybrid	0.593	0.703	0.643	0.587	0.660	0.621
Proposed Hybrid + Self Training	**0.639**	0.673	**0.656**	0.634	0.677	**0.655**

Table 2: Macro average evaluation results on 5-fold cross validation. Results of CNN and LSTM are based on the reimplementation of (Negi et al., 2016)

Model		Hotel			Electronics	
	Precision	Recall	F1	Precision	Recall	F1
Hybrid	0.593	0.703	0.643	0.587	0.660	0.621
Hybrid - CNN encoder	0.585	0.696	0.636	0.542	0.721	0.618
Hybrid - RNN encoder	0.636	0.641	0.638	0.586	0.644	0.614
Hybrid - Linguistic encoder	0.554	0.626	0.588	0.615	0.633	0.624

Table 3: Macro average evaluation on 5-fold cross validation for the ablation study of different component models

system to analyze the importance of each. Ablation studies of these models are reported in Table 3. For hotel reviews the order of importance of feature encoders are: *Linguistic encoder >CNN Encoder >RNN Encoder*. For electronics reviews the importance of model encoders are: *RNN Encoder >CNN Encoder >Linguistic Encoder*. Effectively, different representations of the review sentences and the corresponding features are indeed important for the classification task.

The use of self-training further improves the precision of the proposed hybrid model because it conservatively adds high confidence predictions obtained from the unlabeled data to the training data in each iteration. Inclusion of the 'suggestion' class examples into training helped in reducing the class imbalance, which leads to the improved recall scores for the positive class. Augmentation of new data also added more lexical variability for the system to learn. This, in turn, helps for better classification with the improved F1 scores of 0.656 and 0.655 for the hotel and electronics domains, respectively. The self training runs for a mean of 3.2 (SD = 0.75) iterations for the hotel domain, and 3.6 iterations (SD = 1.62) for the electronics domain. Thus, the expected number of unlabeled sentences added are 640 and 720, respectively. Our proposed system clearly performs better than the state-of-the-art model (Negi and Buitelaar, 2015) with the increments of 8.9 and 1.5 F1 score points for the

hotel and electronics domain, respectively. Please note that the SVM based model was trained with a diverse and rich feature set. Statistical T-test show the performance improvement as significant.

5 Error Analysis

In order to understand the behaviors of our proposed model, we perform error analysis-both quantitatively and qualitatively. For quantitative analysis we depict the confusion matrix in Figure 7. Our closer analysis reveals that a lot of electron-
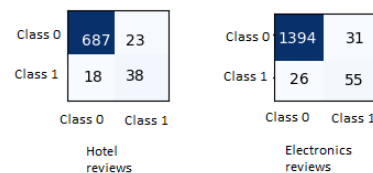

Figure 7: Confusion matrix on test set using Hybrid+Self- training. Here, 1: suggestive and 0: non-suggestive

ics reviews are slightly longer and more complex than the hotel reviews, making it slightly harder to predict despite having slightly more balanced class distribution. Moreover, the presence of only 273 reviews of class 1 in all (about 218 reviews in training), is too small for the architectures to effectively model.

We provide more detailed analysis with the actual examples. At first we describe the phenom-

ena where the instances are incorrectly predicted as suggestions (i.e. false positive cases):

- Many of the false positives are of imperative mood, but do not contain any suggestion towards any entity or product. For e.g.:
  Forget the fact that it will probably take me a year to figure out all the features this camera has to offer.
- Sometimes a user shares his/her own experience(s) but in the form of a second person, thus confusing the machine to predict as a suggestive sentence. For e.g.
  You book a top floor, you get first floor, you booked a suite, and got a room...you go out to your balcony to relax...and someone from a top floor....,which you reserved, has just spit on the back of your head.
- Many of the sentences consist of objects in the second person, but the sentences are not imperative in mood. Such errors are more common in the CNN based model, but are lesser in the proposed hybrid system that makes use of self-training. For e.g.
  If we find some great cheap places we will share it with you.
  "Sentence: very comfortable camera , easy to use , and the best digital photos you re going to get at this price"
- LSTM model sometimes incorrectly predicts those review texts where tokens with VB PoS tags appear. This happens because the sentence appears to be similar to a suggestive sentence that also starts from that particular word having VB PoS tag. For e.g.
  You need the storage to hold a decent amount of shots at 4 megapixel resolution
  might be confused with
  Hold a decent amount of ....
- Some of the false positives are actually suggestions which appear to be wrongly labeled in the original dataset. *We would definitely recommend this hotel to our friends.*
- Suggestions against a product/service which are sarcastic in nature have been annotated as non-suggestive but are difficult for our system to differentiate from the usual suggestions.
  I recommend this hotel only if you don't mind blithely throwing money around, and if you bring your own towels

We also show here few examples that contribute to the false negatives:

- When the sentences are very long, and only a clause of the text belongs to the imperative mood, it is missed by even the best system. For e.g. *"The battery lasts very long when playing music, but writing files to the player drains the battery fast , so you need to have it plugged into an outlet when sending files. "*

- Sometimes two sentences are clubbed together into one when the end marker is missing. In such scenarios, one of the sentences is suggestive and the other is not. In these cases the system predicts the sentence as non-suggestive. For e.g.
  "My only suggestion is to get a lens protector to help protect the shooting lens the lens coating will wear out after so many clean wipes and I m getting the those 52 mm adapter and uv lens filter at lensmateonline.com ."

  It becomes more tricky for the machine if the remaining part of the sentence contains multiple occurrences of first person pronoun. For e.g. *"You have to press the buttons hard and frequently I end up pressing enter when I meant to scroll ."*

From qualitative analysis we observe that systems have learned the ability to identify the sentences with suggestive terms and also the sentences which are imperative in nature. We believe that many of these errors can be reduced to a greater extent by increasing the size of the training data. With sufficient data, systems would be able to learn to better model the input, extract more relevant features, and be able to reason better about the differences between the suggestive sentences and sentences which look like suggestions.

6 Conclusion and Future Work

In this paper, we have proposed a hybrid deep learning model for the task of suggestion mining by incorporating richer and diverse representations of the inputs. We have also used self-training algorithm, which even improved the performance of the hybrid model, opening up more opportunities for the use of semi-supervised learning for this task. Experiments on benchmark datasets show that we obtain superior performance over the existing state-of-the-art system. In the future, we would like extend our work to other semi-supervised learning algorithms.

References

Caroline Brun and Caroline Hagege. 2013. Suggestion mining: Detecting suggestions for improvement in users' comments. *Research in Computing Science*, 70:199–209.

Li Dong, Furu Wei, Yajuan Duan, Xiaohua Liu, Ming Zhou, and Ke Xu. 2013. The automated acquisition of suggestions from tweets. In *AAAI*.

Deepak Gupta, Asif Ekbal, and Pushpak Bhattacharyya. 2018a. A Deep Neural Network based Approach for Entity Extraction in Code-Mixed Indian Social Media Text. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Deepak Gupta, Sarah Kohail, and Pushpak Bhattacharyya. 2018b. Combining graph-based dependency features with convolutional neural network for answer triggering. *arXiv preprint arXiv:1808.01650*.

Deepak Gupta, Pabitra Lenka, Harsimran Bedi, Asif Ekbal, and Pushpak Bhattacharyya. 2017. Iitp at ijcnlp-2017 task 4: Auto analysis of customer feedback using cnn and gru network. In *Proceedings of the IJCNLP 2017, Shared Tasks*, pages 184–193. Asian Federation of Natural Language Processing.

Deepak Gupta, pabitra Lenka, Asif Ekbal, and Pushpak Bhattacharyya. 2018c. Uncovering code-mixed challenges: A framework for linguistically driven question generation and neural based question answering. In *Proceedings of 22nd International Conference on Computational Natural Language Learning . (CoNLL 2018)*. Association for Computational Linguistics (Accepted).

Deepak Gupta, Rajkumar Pujari, Asif Ekbal, Pushpak Bhattacharyya, Anutosh Maitra, Tom Jain, and Shubhashis Sengupta. 2018d. Can taxonomy help? improving semantic question matching using question taxonomy. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING 2018)*, pages 499–513. Association for Computational Linguistics.

Deepak Kumar Gupta and Asif Ekbal. 2014. IITP: Supervised Machine Learning for Aspect based Sentiment Analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 319–323, Dublin, Ireland. Association for Computational Linguistics and Dublin City University.

Deepak Kumar Gupta, Kandula Srikanth Reddy, Asif Ekbal, et al. 2015. PSO-ASent: Feature Selection using Particle Swarm Optimization for Aspect Based Sentiment analysis. In *International Conference on Applications of Natural Language to Information Systems (NLDB-2015)*, pages 220–233. Springer.

Ruining He and Julian McAuley. 2016. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *WWW*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177. ACM.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar. Association for Computational Linguistics.

Anutosh Maitra, Shubhashis Sengupta, Deepak Gupta, Rajkumar Pujari, Asif Ekbal, Pushpak Bhattacharyya, Anutosh Maitra, Mukhopadhyay Abhisek, and Tom Jain. 2018. Semantic question matching in data constrained environment. In *Proceedings of the 21st International Conference on Text, Speech and Dialogue (TSD-2018)*, pages 499–513.

Sapna Negi, Kartik Asooja, Shubham Mehrotra, and Paul Buitelaar. 2016. A study of suggestions in opinionated texts and their automatic detection. In *Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics*, pages 170–178.

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.

Thi-Lan Ngo, Khac Linh Pham, Hideaki Takeda, Son Bao Pham, and Xuan Hieu Phan. 2017. On the identification of suggestion intents from vietnamese conversational texts. In *Proceedings of the Eighth International Symposium on Information and Communication Technology*, pages 417–424. ACM.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Colin Raffel and Daniel PW Ellis. 2015. Feedforward networks with attention can solve some long-term memory problems. *arXiv preprint arXiv:1512.08756*.

Tun Thura Thet, Jin-Cheon Na, and Christopher SG Khoo. 2010. Aspect-based sentiment analysis of movie reviews on discussion boards. *Journal of information science*, 36(6):823–848.

Henning Wachsmuth, Martin Trenkmann, Benno Stein, Gregor Engels, and Tsvetomira Palakarska. 2014. A review corpus for argumentation analysis. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 115–127. Springer.

Alfan Farizki Wicaksono and Sung-Hyon Myaeng. 2013. Toward advice mining: Conditional random fields for extracting advice-revealing text units. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 2039–2048. ACM.

X Zhu. 2006. Semi-supervised learning literature survey, department of computer sciences, university of wisconsin at madison, madison. Technical report, WI, Technical Report 1530. http://pages. cs. wisc. edu/~ jerryzhu/pub/ssl_survey. pdf.