SemEval-2019 Task 9: Suggestion Mining from Online Reviews and Forums

Sapna Negi
Genesys Telecommunication Laboratories
Galway, Ireland
sapna.negi1@gmail.com

Tobias Daudert
National University of Ireland
Galway, Ireland
tobias.daudert@insight-centre.org

Paul Buitelaar
National University of Ireland
Galway, Ireland
paul.buitelaar@insight-centre.org

Abstract

We present the pilot SemEval task on Suggestion Mining. The task consists of subtasks A and B, where we created labeled data from feedback forum and hotel reviews respectively. Subtask A provides training and test data from the same domain, while Subtask B evaluates the system on a test dataset from a different domain than the available training data. 33 teams participated in the shared task, with a total of 50 members. We summarize the problem definition, benchmark dataset preparation, and methods used by the participating teams, providing details of the methods used by the top ranked systems. The dataset is made freely available to help advance the research in suggestion mining, and reproduce the systems submitted under this task.

1 Introduction

State of the art opinion mining systems provide numerical summaries of sentiments and tend to overlook additional descriptive and potentially useful content present in the opinionated text. We stress that such content also encompass information like suggestions, tips, and advice, which is otherwise explicitly sought by the stakeholders. For example, hotel reviews often contain room tips, i.e., which room should be preferred in a hotel. Likewise, tips on restaurants, shops, sightseeing, etc. are also present within the hotel reviews. On the other hand, platforms like Tripadvisor[1], which collect hotel and restaurant related opinions, request the reviewers to fill up the room tips section in addition to the hotel review. Likewise, sentences expressing advice, tips, and recommendations relating to a target entity can often be present in text available from different types of data sources, like blogs, microblogs, discussions, etc. Such sentences can be collectively referred to as suggestions. With the increasing availability of opinionated text, methods for automatic detection of suggestions can be employed for different use cases. Some example use cases are the extraction of suggestions for brand improvement, the extraction of tips and advice for customers, the extraction of the expressions of recommendations from unstructured data in order to aid recommender systems, or the summarisation of suggestion forums where suggestion providers often tend to provide context in their responses (Figure 1) which gets repetitive over a large number of responses relating to the same entity. The task of automatic identification of suggestions in a given text is referred to as *suggestion mining* (Brun and Hagege, 2013).

Studies performed on suggestion mining have defined it as a sentence classification task, where class prediction has to be made on each sentence of a given text, classes being *suggestion* and *non suggestion* (Negi, 2016). State of the art opinion mining systems have mostly focused on identifying sentiment polarity of the text. Therefore, suggestion mining remains a very less explored problem as compared to sentiment analysis, specially in the context of recent advancements in neural network based approaches for feature learning and transfer learning.

As suggestion mining is still an emerging research area, it lacks benchmark datasets and well defined annotation guidelines. A few early works were mostly rule based methods, mainly targeted towards the use case of extracting suggestions for product improvements (Brun and Hagege, 2013; Ramanand et al., 2010; Moghaddam, 2015). In our prior work, we performed early investigations on the problem definition and datasets, aiming for the statistical methods which also require benchmark train datasets in addition to the evaluation

[1] https://www.tripadvisor.com

Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019), pages 877–887
Minneapolis, Minnesota, USA, June 6–7, 2019.

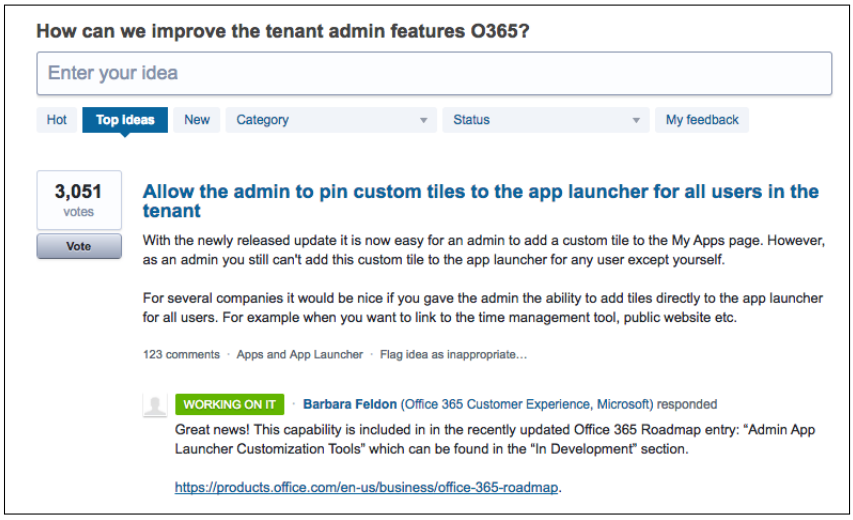

Figure 1: A post from the suggestion forum for Microsoft developers

datasets(Negi and Buitelaar, 2015; Negi et al., 2016). A few other works also evaluated statistical classifiers (Wicaksono and Myaeng, 2012; Dong et al., 2013), which employed mostly manually identified features, however only two other works (Wicaksono and Myaeng, 2012; Dong et al., 2013) provided their datasets. Suggestion mining still lacks well defined annotation guidelines, a multi-domain and cross-domain approach to the problem and benchmark datasets, which we address in our recent work (Negi et al., 2018). Therefore, we introduce this pilot shared task to disseminate suggestion mining benchmarks and evaluate state of the art methods for text classification on domain specific and cross domain training scenarios. The datasets released as a part of the shared task include the domains hotel reviews and software developers suggestion forum (see Table 1).

Suggestion mining faces similar text processing challenges as other sentence or short text classification tasks related to opinion mining and subjectivity analysis, such as stance detection (Mohammad et al., 2016), or tweet sentiment classification (Rosenthal et al., 2015). Some of the observed challenges in suggestion mining are elaborated below:

- **Class imbalance:** Usually, suggestions tend to appear sparsely among opinionated text, which leads to higher data annotation costs and results in a class distribution bias in the trained models.

- **Figurative expressions:** Text from social media and other sources usually contains figurative use of language, which demands pragmatic understanding from the models. For example, 'Try asking for extra juice at breakfast - its 22 euros!!!!!' is more of a sarcasm than a suggestion. Therefore, a sentence framed as a typical suggestions may not always be a suggestion and vice versa. A variety of linguistic strategies used in suggestions also make this task interesting from a computational linguistics perspective and labeled datasets can be leveraged for linguistic studies as well.

- **Context dependency:** In some cases, context plays a major role in determining whether a sentence is a suggestion or not. For example, 'There is a parking garage on the corner of the Forbes showroom.' can be labeled as a suggestion (for parking space) when it appears in a restaurant review and a human annotator gets to read the full review. However, the same sentence would not be labeled as a suggestion if the text is aimed to describe the surroundings of the Forbes showroom.

- **Long and complex sentences:** Often, a suggestion is expressed in either one part of a sentence, or it is elaborated as a long sentence, like, *'I think that there should be a nice feature where you can be able to slide the status bar down and view all the push notifications that you got but you didn't view, just like*

Source	Sentence	Label
Hotel reviews	Be sure to specify a room at the back of the hotel.	suggestion
Hotel reviews	The point is, don't advertise the service if there are caveats that go with it.	non-suggestion
Suggestion forum	Why not let us have several pages that we can put tiles on and name whatever we want to	suggestion
Suggestion forum	It fails with a uninformative message indicating deployment failed.	non-suggestion

Table 1: Examples of suggestions found in reviews and the labels assigned to the suggestion sentences

android and IOS, but the best part is that it fixes many problems like when people wanted a short cut to turn WiFi on and off and data on and off so that would be a nice feature to have 2'. This poses challenges to the training of algorithms in terms of learning effective features, as well as for certain pre-processing steps like part of speech tagging.

Investigating the development of high performance suggestion mining systems could drive the engagement of both, commercial entities (like brand owners) as well as the research communities, working on problems such as opinion mining, supervised learning, or representation learning. A suggestion mining component can empower both, public and private sectors, to extract and leverage suggestions which are constantly expressed on various online platforms like Twitter[2] , TripAdvisor, or Reddit[3] for developing innovative services and products.

2 Task Definition

The early rule based approaches towards suggestion mining assumed that suggestions are always expressed using standard expressions like 'I recommend', 'I suggest that', 'You should', and created small evaluation datasets which were labeled in-house (Brun and Hagege, 2013; Ramanand et al., 2010). Only two of the previous studies released training datasets, which cover travel discussion forums (Wicaksono and Myaeng, 2013) and microblogs (Dong et al., 2013), while the review datasets from the previous works remain proprietary (Ramanand et al., 2010; Moghaddam, 2015). In our recent work, we perform a qualitative analysis of datasets from different sources, which includes investigation of linguistic properties of suggestions, relationship between sentiments and suggestions, and a laymans perception of suggestions (Negi, 2019). We also observed a low inter-annotator agreement in labeling sentences as suggestions and non-suggestions, and formulate a typology for sentences in context to suggestion detection, and design an annotation procedure based on this typology (Negi et al., 2018; Negi, 2019). For this shared task, we extend datasets from our previous studies, following the same task description and annotation method.

The Oxford dictionary defines suggestion as, *An idea or plan put forward for consideration*, and some of the listed synonyms of suggestions are *proposal, proposition, recommendation, advice, hint, tip, clue*. Many linguistic studies define how suggestions should be expressed in a standard use of language. However, in the context of text mining, we are dealing with user generated text on the web, which can be associated with multiple contexts, like the end user, domain etc. We observed in our layman annotation study, context may affect an annotator's judgment. In the absence of context, different annotators associate different contexts to a candidate sentence. We observed that the following concepts form an integral part of defining a suggestion in the context of suggestion mining and proposed an empirically driven and context-based definition of suggestions.

- **Surface structure:** Different surface structures (Chomsky, 1957; Crystal, 2011) can be used to express the underlying intention of giving the same suggestion. For example, *The nearby food outlets serve fresh local breakfast and are also cheaper* and *You can also have breakfast at the nearby food outlets which are cheaper and equally good.*

- **Context:** When dealing with specific use cases, context plays an important role in distinguishing a suggestion from a non-suggestion. Context may be present within

[2] https://www.twitter.com
[3] https://www.reddit.com

a given sentence. It can be a set of values corresponding to different variables that are provided explicitly and in addition to a given sentence. One or more of the following variables can constitute the context:

Domain: In this work, we refer to the source of a text as *domain*, which should not be considered in-line with the standard definition of domain. For example, in this shared task, we used hotel and suggestion forum domains.

Source text: The text in the entire source document to which a sentence belongs may also serve as a context, giving an insight into the discourse where the suggestion appeared.

Application or use case: Suggestions may sometimes be sought only around a specific topic, for example, room tips from hotel reviews. Suggestions can also be selectively mined for a certain class of users, for example, suggestions for future customers. All non-relevant suggestions in the data may be regarded as non-suggestions in this case.

Given that

- s denotes the surface structure of a sentence,
- C denotes additional context provided with s, where the context can be a set of values corresponding to certain variables, and
- $a(s, C)$ denotes the annotation agreement for the sentence, and t denotes a threshold value for the annotation agreement,

we write $S(s, C)$ to denote the *suggestion function*, which is defined as

$$S(s, C) = \begin{cases} Suggestion, & \text{if } a(s, C) \geq t \\ Non\text{-}suggestion, & \text{if } a(s, C) < t. \end{cases} \quad (1)$$

Depending on the choice of C, and, hence, on the value of $a(s, C)$, we identify four categories of sentences that a suggestion mining system is likely to encounter.

Explicit suggestions. *Explicit suggestions* are sentences for which S always outputs *Suggestion*, whether C is the empty set or not. They are like the *direct* and *conventionalised* forms of suggestions as defined by (Martínez Flor, 2005). It may also be the case that such sentences have a strong presence of context within their surface form, as in illustrated by *If you do end up here, be sure to specify a room at the back of the hotel*.

Explicit non-suggestions. These are the sentences for which S always outputs *Non-suggestion*, whether C is the empty set or not. For example, *Just returned from a 3 night stay*.

Implicit suggestions. These are sentences for which S outputs *Non-suggestion* only when C is the empty set. Typically, implicit suggestions do not posses the surface form of suggestions but the additional context helps the readers to identify them as suggestions. For example, *There is a parking garage on the corner of Forbes, so its pretty convenient* is labeled as a suggestion by the annotators when the context is revealed as that of a restaurant review. A sentence such as *Malahide is a pleasant village-turned-dormitory-town near the airport* can be considered as a suggestion given that it is obtained from a travel discussion thread for Dublin. These kind of sentences are observed to have a lower inter annotator agreement than the above two categories.

Implicit non-suggestions. These are sentences for which S outputs *Suggestion* only when C is an empty set. Typically, an implicit non-suggestion posses the surface form of suggestions but the context leads readers to identify them as non-suggestions. Such sentences may contain sarcasm. Examples include *Do not advertise if you don't know how to cook* appearing in a restaurant review and *The iPod is a very easy to use MP3 player, and if you can't figure this out, you shouldn't even own one* appearing in a MP3 player review.

The proposed categories provide the flexibility to change the scope of classes in a well defined manner, as well as to define context as per the application and use case. Based on the above four categoriese can define the scope of suggestion and non-suggestion classes for suggestion mining tasks. For open domain and cross domain suggestion mining, we proposed to limit the scope of suggestions to the *explicit suggestions*. Therefore, we set the definition of suggestion for this shared task as:

Let s be a sentence. If s is an explicit suggestion, assign the label *Suggestion*. Otherwise, assign the label *Non-suggestion*.

3 Dataset Annotation

A two phase annotation methodology, as proposed in our previous works (Negi et al., 2018; Negi, 2019) is followed.

3.1 Phase 1: Crowdsourced Annotations

This phase is performed using paid crowdsourcing, where each sentence is annotated by multiple layman annotators, and the set of annotators do not necessarily remain the same for all the sentences. We used Figure Eight[4] to collect layman annotations.

Annotators were also provided with the context, i.e. source text from where the sentence is extracted. They were simply asked to choose to label a sentence as suggestions if it contained expressions of suggestion, advice, tip, and recommendation. We aimed to collect implicit and explicit suggestions in this phase.

For quality control, before being allowed to perform a job, the annotators were presented with a set of test sentences which are similar to the actual questions except that their answers have already been provided by us to the system. We also submitted the explanation behind the correct answer. This way the test questions serve two purposes: test the annotators competency and understanding of the job, and train the annotator for the job. Crowdflower recommends certain best practices to prepare effective test questions.[5]. We submitted 30 test questions for each dataset. Each starting annotator was presented with 10 test questions, and only the annotators achieving an accuracy of 70% or more were allowed to proceed with the job. If an annotator passed the test and started the job, the remaining unseen test questions were presented to them in between the regular sentences without being notified. One sentence out of every 8 was a hidden test question. The accuracy score of a contributor on test questions is referred to as *Trust score* in a job. If an annotator's trust score dropped below a certain threshold during the course of the annotation, the system did not allow them to proceed further with the job. This threshold score was set to 70% in our case.

In addition to the hidden test questions, a minimum time for each annotator to stay on one page of the job was set. We set this time to 40 seconds (5 seconds on average for each sentence). If annotators appeared to be faster than that, they were automatically removed from the job. We restricted access to annotators from countries where English is a popular language and that are also likely to have a large crowdsourcing workforce. Most of the annotators came from Australia, Canada, Germany, India, Ireland, the United Kingdom, and the USA.

Annotation agreement: Crowdflower's *confidence* score describes the level of agreement between multiple contributors and the confidence in the validity of the result at the same time, we used a threshold confidence score of 0.6. However, it can be the case that a sentence is very ambiguous and cannot achieve the confidence score even after a large number of workers answered it. A maximum limit to the number of annotators is set in such case, and no further judgements are collected even if the threshold confidence is not reached. We set this limit to 5 annotators. Sentences that do not pass the confidence threshold of 0.6 are not included in the dataset.

3.2 Phase 2: Expert Annotations

This phase is performed by two in-house expert annotators, who are provided with the detailed annotation guidelines as compared to the phase 1 annotation guidelines, and the annotators are familiar with the problem definition and the task at hand. However, the annotators are not provided with the source text in this case. Phase 2 of the annotation is only applied to sentences that were labeled as suggestions in Phase 1, which drastically reduces the number of annotations to be performed in Phase 2.

Annotation Agreement: The inter-annotator agreement for Phase 2 was calculated by having two annotators label a subset of sentences for each domain (50 sentences). Cohen's kappa coefficient was used to measure the inter-annotator agreement. The remainder of the data instances were annotated by only one annotator. The fol-

[4]Earlier known as Crowdflower. https://www.figure-eight.com/

[5]https://success.crowdflower.com/hc/en-us/articles/213078963-Test-Question-Best-Practices

Subtask	Domain	Suggestion/Non-suggestion			IA agreement (phase 2)
		Training	Trial Test	Test	
A	Software developer suggestion forums (Uservoice)	1428 / 4296	296 / 742	87/746	0.81
B	Hotel reviews (Trip Advisor)	448 / 7086	404 / 3000	348/476	0.86

Table 2: Details of released datasets

lowing guidelines were provided to the annotators in Phase 2 :

- The intent of giving a suggestion and the suggested action or recommended entity should be explicitly stated in the sentence. *Try the cup cakes at the bakery next door* is a positive example. Other explicit forms of this suggestion could be: *I recommend the cup cakes at the bakery next door* or *You should definitely taste the cup cakes from the bakery next door*. An implicit way of expressing the suggestion could be *The cup cakes from the bakery next door were delicious*.

- The suggestion should have the intent of benefiting a stakeholder and should not be mere sarcasm or a joke. For example, *If the player doesn't work now, you can run it over with your car* would not pass this test.

Following are some of the scenarios of conflicting judgments observed in this phase of annotation:

- In the case of suggestion forums for specific domains, like a software developer forum, domain knowledge is required to distinguish an implicit non-suggestion from an explicit suggestion. Consider, for example, the two sentences, *It needs to be an integrated part of the phones functionality, that is why I put it in Framework* and *Secondly, you need to limit the number of apps that a publisher can submit with a particular key word*. The first sentence is a description of already existing functionality and is a context sentence in the original post, while the second is suggestion for a new feature.

- No concrete mention of what is being advised such as in *It'd be great if you would work on a solution to improve the situation*.

- At times, there was a confusion between information (fact) or suggestion (opinion). For example, *You can get a ticket that covers 6 of the National Gallery sites for only about US$10*.

In the final dataset, the sentences that are labeled as suggestions in Phase 2 of the annotation process are labeled as suggestions, while all other sentences are labeled as non-suggestions.

4 SemEval 2019 Shared Task

This is the pilot shared task on suggestion mining, the task is set as a binary sentence classification task, where the classes are suggestion and non-suggestion. As explained previously, explicit suggestions are deemed as the suggestion class, and rest of the sentences are considered as non-suggestions. The task is further split into two subtasks, named as A and B. Participating teams were to participate in at-least one of the two subtasks.

Datasets: Table 2 lists the details of the currently datasets released under this task and the inter-annotator agreement in the phase 2 of annotations. The class distribution is retained as obtained from a random sample of the source dataset used for annotation.

Software suggestion forum: The sentences for this dataset were scraped from the Uservoice platform[6]. Uservoice provides customer engagement tools tobrands, and therefore hosts dedicated suggestion forums for certain prod-ucts. The Feedly mobile application forum and the Windows developer forum are openly accessible. A sample of posts were scraped and split into sentences using the Stanford CoreNLP toolkit. Many suggestions are in the form of requests, which is less frequent in other domains. The text contains highly technical vocabulary related to the software which is being discussed.

Hotel reviews: Wachsmuth et al. (2014) provide a large dataset of hotel reviews collected from the TripAdvisor website[7]. They segmented the

[6] https://www.uservoice.com/
[7] https://www.tripadvisor.com/

reviews into statements so that each statement has only one sentiment label and have manually labeled the sentiments. Statements are equivalent to sentences, and comprise of one or more clauses. We further annotated these segments as *suggestion* and *non-suggestion*.

Sub-Task A: Train and test dataset belong to the same domain. The provided domain is suggestion forum sentences for software developers. Title of the posts are excluded, which are at times summary of the suggestion.
Sub-Task B: No training dataset is provided, and the test dataset belongs to a different domain than the subtask A, i.e. hotel reviews. The participants could use the training dataset from subtask A. Participants were not allowed to use the trial test set for subtask B as a training dataset, however they were allowed to use trial test set as a validation dataset.
Additional resources: Participants were allowed to use additional language resources, with one exception. Participants will be prohibited from using additional hand labeled training datasets for any of the domain.
Evaluation Metrics: Classification performance of the submitted systems if evaluated on the basis of F-1 score for the positive class, i.e. the *suggestion* class, which ranges from 0 to 1.
Precision suggestion (P_{sugg}): The fraction of instances which are actually suggestions out of the ones which are predicted as suggestions.
P_{sugg} = True Positives / (True Positives + False Positives)

Recall suggestion (R_{sugg}): The fraction of suggestion class instances which are correctly identified out of the total number of suggestions.
R_{sugg} = True Positives / (True Positives + False Negatives)

F1 score for the suggestion class is:
$F1_{sugg} = 2 * (P_{sugg} * R_{sugg}) / (P_{sugg} + R_{sugg})$

Baseline System A rule based classifier is employed using the existing rules from some of the related works, the rules which were dependent on the domain specific variables were excluded from the baseline. Table 4 provides the rules used in the baseline system.

Trial vs Test phase: A trial test dataset was released prior to the final test/evaluation dataset. The class distribution in the trial set was deliberately balanced in order to not bias the participants towards a specific class distribution for the evaluation phase, and keep the class distribution of trial set different from that of the final test set. This was because the trial test dataset labels were released prior to the final evalaution phase, and it was used as a validation dataset by the participants.

5 Participating Systems

A total of 33 teams participated in the evaluation phase, where all teams participated in the subtask A, and 16 of these also participated in subtask B. This number is lower than the trial phase submissions, where a total of 50 teams submitted their results on trial test dataset. Out of 33, 20 teams also submitted their system description papers. A summary of these 20 systems is provided in Table 3, listing results and corresponding methods. The highest F-score achieved was reasonably high i.e. 0.78 for subtask A, given a very low number of suggestion sentences in the test dataset 2. The highest F-score for subtask B was 0.858, where the ratio of suggestion and non-suggestion sentences in the test set was higher than subtask A.

Top 3 systems: BERT (Devlin et al., 2018) pre-trained language model remains the common method in the top three systems submitted in subtask A, which is one of the state of the art statistical language models. However, the most interesting results are provided by the best performing system in subtask B, which uses a rule based classifier, where rules comprise of both words and POS tags. The devised rule-based classifier (Potamias et al., 2019) assigns confidence scores to sentences on the basis of lexical patterns organised in pre-specified categories and lexical lists corresponding to each subtask. This rule based system also performed fairly well in subtask A, where it achieved rank 5.

Transfer and Unsupervised Learning: While a variety of pre-trained word embeddings and language models were employed, BERT remains the most popular means of transfer learning in the submitted systems, where 7 out of top 13 systems for subtask A used BERT.
For subtask B, only two systems used additional

Rank		Team Name	F-score		Method Used
Subtask A	Subtask B		Subtask A	Subtask B	
1	2	OleNet@Baidu (Jiaxiang et al., 2019)	0.7812	0.8579	Ensemble classifier (Logistic, GRU, FFA, CNN), with BERT
2	12	ThisIsCompetition (Park et al., 2019)	0.7778	0.6486	Ensemble classifier. Attention sentence encoder. BERT, CNN based word encoder.
3	5	m_y (Yamamoto and Sekiya, 2019)	0.7761	0.793	Distant supervision on unlabeled hotel reviews. BERT, ULMfit
4	NA	Yimmon (Zhuang, 2019)	0.7629	NA	Customised network, combination of convolution, self-attention and feed-forward layers. BERT
5	1	NTUA-ISLab (Potamias et al., 2019)	0.7488	0.858	Automatically learned rules
6	13	YNU-HPCC (Ping et al., 2019)	0.735	0.503	Ensemble classifier CNN, BILSTM and GRU. BERT
7	4	DS (Cabanski, 2019)	0.7273	0.8187	Ensemble classifier CNN and LSTM. BERT
9	NA	ZQM (Zhou et al., 2019)	0.715	NA	CNN. BERT
10	NA	MIDAS (Anand et al., 2019)	0.7011		Naive Bayes, Logistic Regression, SVM, LSTM. ULMFit
12	11	NL-FIIT (Pecar et al., 2019)	0.6816	0.685	Bi-LSTM. ELmO
13	3	Zoho (Prasanna and Seelan, 2019)	0.6807	0.8194	CNN. GloVe, BERT
14	NA	Lijunyi (Li and Ding, 2019)	0.6776	NA	Ensemble classifier, LSTM (attention-based), TextCNN, C-LSTM, Bi-LSTM. Word2Vec
19	7	WUT (Klimaszewski and Andruszkiewicz, 2019)	0.6293	0.7778	Domain-Adversarial Neural Networks (DANN). ELMo
23	6	Taurus (Oostdijk and Halteren, 2019)	0.5845	0.7925	Rules
27	NA	YNU_DYX (Ding et al., 2019)	0.5659	NA	BiLSTM, LSTM. Word2Vec, GloVe
28	9	INRIA (Markov and De la Clergerie, 2019)	0.5118	0.733	SVM, Logistic Regression. Hand crafted features.
29	17	SSN-SPARKS (S et al., 2019)	0.494	0.155	MultiLayer Perceptron, Random Forest and Convolutional Neural Network
30	14	DBMS-KU (Fatyanosa et al., 2019)	0.473	0.369	SVM, Linear Regression, Naive Bayes, CNN. GloVe
31	NA	UOL Artificial Intelligence Research Group (Ahmed et al., 2019)	0.3537	NA	Containment similarity, maximum common subgraph, Tree-based Pipeline Optimization Tool (TPOT)
NA	8	Hybrid RNN (Ezen-Can and F. Can, 2019)	NA	0.7449	Rule-based patterns, Glove, Bi-LSTM
32	10	Baseline	0.268	0.7329	Manually observed rules

Table 3: A summary of systems which are available as system description papers.

Keywords and phrases
needs to, need to
suggest, recommend, if, i wish, go for,should have, would, could have been
i would like, i'd like, i would love, I'd love, love to see
there should be, I wish, allow us to
Syntactic clues
If a modal verb or a base form of verb is present in the sentence. Eg, *I would prefer the unit to have a simple on off* switch.

Table 4: Rules for the baseline system

domain specific unlabeled data, i.e. hotel reviews, and were ranked as 3 and 5. All other submissions for subtask B relied on pre-trained word embeddings and language models.

Class Imbalance: Given that there was a major difference in the class distribution between training, trial test, and final test datasets, a minority of the top ten systems explicitly handle class imbalance by methods like oversampling (team MIDAS) and assigning weights to the predicted probability which are in proportion to the class distribution of the training data (team Yimmon). Other top 10 systems performed fairly well without any additional configuration for class imbalance in their classifiers.

Types of Classifiers: All systems except two used statistical classifiers, with most of them using neural network classifiers. Classifier ensembles also remain a favoured approach among the top ten systems. The neural network classifiers clearly outperformed SVM, Naive Bayes and Logistic regression. For subtask B, rule based classifier seem to do fairly well. The state of the art deep learning classifiers achieved a similar performance without any manual feature engineering, as compared to the carefully hand crafted rules.

6 Summary

We organised the pilot shared task on suggestion mining, which was framed as a binary text classification task, with two subtasks representing domain dependent and cross-domain/open domain evaluation. The task achieved a high level of participation, and most importantly a wide coverage in terms of methods and algorithms. The approaches covered automatically learned rule, carefully crafted linguistic features and rules, SOTA neural network classifiers, and SOTA transfer learning approaches. This shared task acted as a catalyst in pushing forward the state of the art for Suggestion Mining which otherwise received We plan to extend the task in future years with larger datasets, and the problem framed as the extraction of suggestion sentences from source texts in place of sentence classification. The problem definition here a better availability of document level context as compared to the sentence level context.

References

Usman Ahmed, Humera Liaquat, Luqman Ahmed, and Syed Jawad Hussain. 2019. Suggestion miner at semeval-2019 task 9: Suggestion detection in online forum using word graph. In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*.

Sarthak Anand, Debanjan Mahata, Kartik Aggarwal, Laiba Mehnaz, Simra Shahid, Haimin Zhang, Yaman Kumar, Rajiv Ratn Shah, and Karan Uppal. 2019. Midas at semeval-2019 task 9: Suggestion mining from online reviews using ulmfit. In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*.

Caroline Brun and Caroline Hagege. 2013. Suggestion mining: Detecting suggestions for improvement in users comments. *Research in Computing Science*.

Tobias Cabanski. 2019. Ds at semeval-2019 task 9: From suggestion mining with neural networks to adversarial cross-domain classification. In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*, Minneapolis, Minnesota.

Noam Chomsky. 1957. *Syntactic Structures*. Mouton and Co., The Hague.

D. Crystal. 2011. *A Dictionary of Linguistics and Phonetics*. The Language Library. Wiley.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Yunxia Ding, Xiaobing Zhou, and Xuejie Zhang. 2019. Ynu_dyx at semeval-2019 task 9: A stacked bilstm model for suggestion mining classific. In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*.

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*. AAAI Press.

Aysu Ezen-Can and Ethem F. Can. 2019. Hybrid rnn at semeval-2019 task 9: Blending information sources

for domain-independent suggestion mining. In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*.

Tirana Noor Fatyanosa, Al Hafiz Akbar, Maulana Siagian, and Masayoshi Aritsugi. 2019. Dbms-ku at semeval-2019 task 9: Exploring machine learning approaches in classifying text as suggestion or non-suggestion. In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*.

Liu Jiaxiang, Wang Shuohuan, and Sun Yu. 2019. Olenet at semeval-2019 task 9: Bert based multi-perspective models for suggestion mining. In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*.

Mateusz Klimaszewski and Piotr Andruszkiewicz. 2019. Wut at semeval-2019 task 9: Domain-adversarial neural networks for domain adaptation in suggestion mining. In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*.

Junyi Li and Haiyan Ding. 2019. Lijunyi at semeval-2019 task 9: An attention-based lstm model and ensemble of different models for suggestion mining from online reviews and forums. In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*.

Ilia Markov and Eric Villemonte De la Clergerie. 2019. Inria at semeval-2019 task 9: Suggestion mining using svm with handcrafted features. In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*.

Alicia Martínez Flor. 2005. A theoretical review of the speech act of suggesting: Towards a taxonomy for its use in flt. *Revista alicantina de estudios ingleses, No. 18 (Nov. 2005); pp. 167-187*.

Samaneh Moghaddam. 2015. Beyond sentiment analysis: mining defects and improvements from customer feedback. In *European Conference on Information Retrieval*, pages 400–410. Springer.

Saif M Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. 2016. Semeval-2016 task 6: Detecting stance in tweets. pages 31–41.

Sapna Negi. 2016. Suggestion mining from opinionated text. In Alberto Pozzi, Elisabetta Fersini, Enza Messina, and Bing Liu, editors, *Sentiment Analysis in Social Networks*, chapter 8. Elsevier.

Sapna Negi. 2019. *Suggestion mining from text*. Ph.D. thesis, NUI Galway.

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. Association for Computational Linguistics.

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, Lisbon,Portugal. Association for Computational Linguistics.

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

Nelleke Oostdijk and Hans van Halteren. 2019. Team taurus at semeval-2019 task 9: Expert-informed pattern recognition for suggestion mining. In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*.

Cheoneum Park, Juae Kim, Hyeon-gu Lee, Reinald Kim Amplayo, Harksoo Kim, Jungyun Seo, and Changki Lee. 2019. Thisiscompetition at semeval-2019 task 9: Bert is unstable for out-of-domain samples. In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*.

Samuel Pecar, Marian Simko, and Maria Bielikova. 2019. Nl-fiit at semeval-2019 task 9: Neural model ensemble for suggestion mining. In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*.

Yue Ping, Jin Wang, and Xuejie Zhang. 2019. Ynuhpcc at semeval-2019 task 9: Using a bert and cnn-bilstm-gru model for suggestion mining. In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*.

Rolandos Alexandros Potamias, Alexandros Neofytou, and Georgios Siolas. 2019. Ntua-islab at semeval-2019 task 9: Mining suggestions in the wild. In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*, Minneapolis, Minnesota.

Sai Prasanna and Sri Ananda Seelan. 2019. Zoho at semeval-2019 task 9: Semi-supervised domain adaptation using tri-training for suggestion mining. In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*.

J 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.

Sara Rosenthal, Preslav Nakov, Svetlana Kiritchenko, Saif Mohammad, Alan Ritter, and Veselin Stoyanov. 2015. Semeval-2015 task 10: Sentiment analysis in twitter. In *Proceedings of the 9th International Workshop on Semantic Evaluation*, pages 451–463, Denver, Colorado. Association for Computational Linguistics.

Rajalakshmi S, Angel Deborah S, S Milton Rajendram, and Mirnalinee T T. 2019. Ssn-sparks at semeval-2019 task 9: Mining suggestions from online reviews using deep learning techniques on augmented data. In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*.

Henning Wachsmuth, Martin Trenkmann, Benno Stein, Gregor Engels, and Tsvetomira Palakarska. 2014. A review corpus for argumentation analysis. In *Proceedings of the 15th International Conference on Computational Linguistics and Intelligent Text Processing*, volume 8404 of *LNCS*, pages 115–127, Kathmandu, Nepal. Springer.

Alfan Farizki Wicaksono and Sung-Hyon Myaeng. 2012. Mining advices from weblogs. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, CIKM '12, pages 2347–2350. ACM.

Alfan Farizki Wicaksono and Sung-Hyon Myaeng. 2013. Automatic extraction of advice-revealing sentences for advice mining from online forums. In *Proceedings of the Seventh International Conference on Knowledge Capture*, K-CAP '13, pages 97–104. ACM.

Masahiro Yamamoto and Toshiyuki Sekiya. 2019. m_y at semeval2019 task 9: Exploring bert for suggestion mining. In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*.

Qimin Zhou, Zhengxin Zhang, Hao Wu, and Linmao Wang. 2019. Zqm at semeval-2019 task9: A single layer cnn based on pre-trained model for suggestion mining. In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*, Minneapolis, Minnesota.

Yimeng Zhuang. 2019. Yimmon at semeval-2019 task 9: Suggestion mining with hybrid augmented approaches. In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*.