# Towards the Extraction of Customer-to-Customer Suggestions from Reviews

**Sapna Negi**
Insight Centre for Data Analytics
National University of Ireland Galway

**Paul Buitelaar**
Insight Centre for Data Analytics
National University of Ireland Galway

{firstname.lastname}@insight-centre.org

## Abstract

State of the art in opinion mining mainly focuses on positive and negative sentiment summarisation of online customer reviews. We observe that reviewers tend to provide advice, recommendations and tips to the fellow customers on a variety of points of interest. In this work, we target the automatic detection of suggestion expressing sentences in customer reviews. This is a novel problem, and therefore to begin with, requires a well formed problem definition and benchmark dataset. This work provides a 3- fold contribution, namely, problem definition, benchmark dataset, and an approach for detection of suggestions for the customers. The problem is framed as a sentence classification problem, and a set of linguistically motivated features are proposed. Analysis of the nature of suggestions, and classification errors, highlight challenges and research opportunities associated with this problem.

## 1 Introduction

Opinion mining mainly deals with the summarisation of opinions on the basis of their sentiment polarities (Liu, 2012). However, on a closer observation of such opinionated text, we can discover other facets of opinions. For example, a hotel review sentence like, *Make sure you bring plenty of sun tan lotion-very expensive in the gift shop*, would be labeled as a neutral sentiment in current opinion mining methodologies, since it would only be interested in collecting opinions about the hotel. In the case of aspect based sentiment analysis, the sentence does not comprise of hotel related aspects, and thus will again be labeled as neutral/objective. While such sentences are generally ignored in sentiment based opinion summarisation, these can be very useful information to extract from the reviews. In hotel reviews, such suggestions range from tips and advice on the reviewed entity, to suggestions and recommendations about the neighbourhood, transportation, and things to do. Similarly, in product reviews suggestions can be about how to make a better use the product, accessories which go with them, or availability of better deals. We refer to such sentences as customer-to-customer suggestions (CTC).

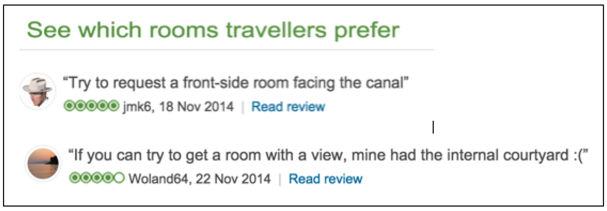
Another type of suggestions, which can appear in the reviews, are the ones aiming at manufacturers or service providers, suggesting new features and improvements in products or services. For example, *An electric kettle in the room would have been a useful addition*. Recently, there have been some works on extracting the suggestions for improvements from reviews (section 3), but they did not focus on CTC suggestions. Also, suggestions for improvement discuss only about the reviewed entity and its aspects, unlike suggestions to customers, which might also include other topics of interest.

Suggestion mining and retrieval can be a potential new research area emerging from this kind of research. Industrial importance of suggestions to customers can be validated from the sections like 'Room Tips' (Figure 1) on TripAdvisor[1]. Similarly, Yelp also features 'tips' [2] (Figure 2) related to a business, and defines it as 'key information'. Such tips are often suggestions, or some important information, a user wants to convey to others. These tips are manually entered by the users, in addition to the reviews. We note that using suggestion mining, such information can be automatically extracted from a large number of already ex-

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

[2] http://www.yelp-support.com/article/What-are-tips

*Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2159–2167,
Lisbon, Portugal, 17-21 September 2015. ©2015 Association for Computational Linguistics.



See which rooms travellers prefer

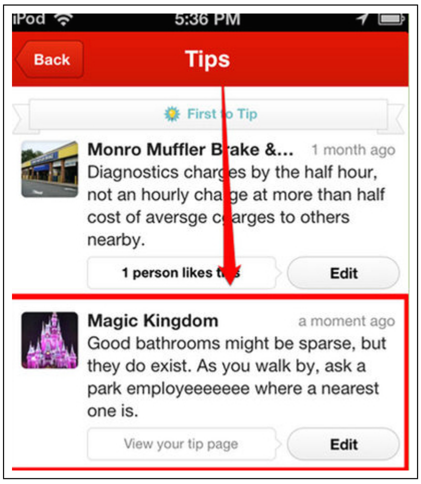
"Try to request a front-side room facing the canal"

jmk6, 18 Nov 2014

"If you can try to get a room with a view, mine had the internal courtyard :("

Woland64, 22 Nov 2014

Figure 1: Room Tips on TripAdvisor



Monro Muffler Brake &...

Diagnostics charges by the half hour, not an hourly charge at more than half cost of aversge charges to others nearby.

Magic Kingdom

Good bathrooms might be sparse, but they do exist. As you walk by, ask a park employeeeeeee where a nearest one is.

Figure 2: Tips on Yelp

isting reviews. The recommendation type of suggestions are of great importance in the case, when there is no dedicated reviews available for small shops, cafe, restaurants etc. in the vicinity of a hotel/business. Suggestions extracted from a large number of reviews, can also be seen as a kind of summarisation, an alternative or complementary to sentiment summarisation over the reviews.

In order to perform suggestion mining, expressions of suggestions should to be detected in a given text. The presence of a variety of linguistic strategies in the suggestion expressing sentences makes this task interesting from a computational linguistics perspective. Section 2 discusses this in more detail. The detection of suggestions in text goes beyond the scope of sentiment polarity detection, and opens up new problems and challenges in the areas of subjectivity analysis, social media analysis, and extra-propositional semantics.

Our main contributions through this work can be listed as:

- Proposition of the task of detection of CTC suggestions.
- A well formed problem definition and scope.
- Preparation of benchmark dataset for two domains of reviews, hotel and electronics.
- An approach to automatically detect CTC suggestions from review texts.

## 2 Task Definition

Since suggestion mining is a young problem, there is a need for problem analysis and definition. As indicated in the previous section, the tasks under suggestion mining may vary. Below, we propose three parameters whose value would help define such tasks, and the values for these parameters in the context of the current task of detection of CTC suggestions.

- **Who is the suggestion aimed at?**
  As we explained in section 1, suggestions can be aimed at one of the two kinds of stakeholders, customers and service providers. In this work, we perform the detection of suggestions for customers only.

- **What should be the textual unit of suggestion?**
  The previous works on extraction of suggestions for product improvement considered sentences as the unit of suggestions. In this work, we also consider sentences as units of suggestion. However, we observe that sentences might miss the context, or refer to something mentioned in the previous sentence. Furthermore, punctuation marks are often erroneously used in social media text, so automatic sentence split does not work well with such text. The approach used in this work aims at the detection of expressions of suggestions, regardless of the presence or absence of context in the same sentence. Therefore, we currently ignore the problems associated with using sentences as a unit for suggestions. The datasets used in this work are sentiment analysis review datasets from other works, in which reviews are already split into sentences. For future works, we assume that context can be determined by the neighbouring text once the expression of suggestion is successfully detected.

- **What kind of text should be considered as a suggestion?**
  Oxford dictionary defines suggestion as, *An idea or plan put forward for consideration*. Some of the listed synonyms of suggestions are *proposal, proposition, recommendation, advice, hint, tip, clue*. In a general scenario, this defini-

| Sentence | Category | $\%$ Annotators | Suggestion Expression |
|---|---|---|---|
| Concierge is not available 24 hours | Inform | 70 | Implicit |
| We did not have breakfast at the hotel but opted to grab breads/pastries/coffee from the many food-outlets in the main stations. | Tell | 80 | Implicit |
| Room was very quiet despite being close to the elevator. | Remark | 50 | Implicit |
| The staff was nice and friendly | Praise | 60 | Implicit |
| Double bed quite narrow and not as comfortable as expected. | Disappoint | 60 | Implicit |
| The view from the 7th floor was amazing. | Fascinate | 50 | Implicit |
| If you do end up here, be sure to specify a room at the back of the hotel. | Suggest | 100 | Explicit |
| I recommend going for a Trabi Safari | Recommend | 90 | Explicit |

Table 1: Sentences Which Were Perceived as CTC Suggestions by Laymen

tion of suggestion easily distinguishes suggestions from other kind of text. However, when suggestions are required to be identified in the space of customer reviews, there seems to be a tendency to consider most of the sentences as suggestions to other customers. This observation is based on a preliminary data annotation task, which was meant for data analysis and development of annotation guidelines. Asher et. al (2009) define 20 types of opinion expressions, including suggestions and recommendations, which appear in opinionated text. In order to form a representative sample, we chose 20 sentences from reviews, corresponding to each of these types, and asked 10 people (layman) to decide if these sentences are CTC suggestions or not. Table 1 shows sentences for which 5 or more people agreed of CTC Suggestion label. Asher et. al. (2009) observe that these types are not uniformly distributed in the reviews. According to them, suggestions and recommendations constitute about 10$\%$ of the statements, while judgements (blame, praise) and sentiments (love, fascinate, hate, disappoint, sad) constitute about 80$\%$. Later, when we annotate the review datasets for CTC sugggestions, they also shows a smaller percentage of suggestions (see table 2).

Except the *suggest* and *recommend* categories, suggestions in rest of the categories are in an implicit form and need to be inferred. Since human beings can inherently infer suggestions, the layman annotators considered both implicit and explicit form of suggestions as suggestions. However, in a real case scenario, humans cannot go through the large amount of reviews and infer suggestions from all of them. In this work, we aim to automatically detect and extract the explicit expressions of suggestions, rather than inferring them.

For the ease of defining the scope of our work, we propose two form of suggestions:
Explicit: Directly suggests/recommends an entity or action,
Implicit: Only provides the information from which a suggestion can be inferred, but do not authoritatively suggest anything.

Lastly, we frame the problem of CTC suggestion detection problem as a sentence classification problem:
*Given a set S of statements* $\{s_1, s_2, s_3, ..., s_n\}$, *predict a label* $l_i$ *for each of statement in S, where* $l_i \in \{$*CTC suggestion, non CTC suggestion*$\}$*, where CTC suggestion should be explicitly expressed.*

## 3 Related Work

Only a few attempts have been made to study suggestion mining, and there is an unavailability of benchmark datasets. Therefore, suggestion mining still remains a young area of study.

- **Suggestion Mining from Customer Reviews**
As mentioned in section 1, there have been some attempts to extract suggestions for improvements in products from customer reviews. Ramanand et. al. (2010) used manually formulated patterns to extract wishes regarding improvements in products. Brun and Hagege (2013) also used manually formulated rules to extract suggestions for improvements from the product reviews. Negi and Buitelaar (2015) studied linguistic nature of suggestions and wishes for improvements and performed experiments in order to assert that these contain subjunctive mood. These works do not acknowledge the fact that reviews can also contain suggestions for other customers. One major drawback of previous works on customer reviews is the public unavailability of evaluation datasets.

- **Other Domains**
  Two other lines of work extracted suggestions from domains other than reviews. Dong et. al. (2013) performed detection of suggestions for product improvement using tweets. They used a statistical classifier, with features comprising of bag of words, automatically extracted suggestion templates using sequential pattern mining and hashtags. (Wicaksono and Myaeng, 2012; Wicaksono and Myaeng, 2013) extracted advice from discussion threads. They also used a supervised classification approach, where some of the features were domain dependent, like the similarity between original query post and a given sentence. They do not make any distinction between implicit and explicit advice, since there is less ambiguity in the domain of discussion forum.

None of the previous works study the complex and interesting nature of suggestions in opinionated text, and the relationship between suggestions and sentiments. Also, there is an unavailability of benchmark datasets of suggestions customer reviews.

## 4 Data

Since there is no available dataset of suggestions for customers, we prepare new datasets for this task. We consider two kinds of reviews for this task, hotel reviews and electronic product reviews.
**Hotel**: Wachsmuth et al. (2014) provide a large dataset of TripAdvisor hotel reviews, where reviews are segmented into statement so that each statement has only one sentiment label. Statements have been manually tagged with positive, negative, conflict and neutral sentiments. We take a smaller subset of these reviews, where each statement is an instance of our dataset. Each statement also bears a unique identity no., which is constituted of hotel identity no., statement number and review identity no. Therefore, the reviews belonging to same hotel, and the statements of the same review can be identified.
**Electronics**: Hu et. al.(2004) provide a dataset consisting of reviews of electronic products, which is also already split into sentences, and the corresponding sentiment for each sentence is manually tagged.
In the next section, we further annotate the sentences from these two datasets, for the current task.

| **Agreement** | **Hotel(8050 sentences)** | **Electronics(3782 sentences)** |
|---|---|---|
| Phase 1 | | |
| Confidence | CTC Sugg. | CTC Sugg. |
| >= 0.6 | 3220 | 1488 |
| >= 0.7 | 1046 | 604 |
| >= 0.8 | 1024 | 562 |
| >= 0.9 | 1020 | 558 |
| 1 | 1019 | 553 |
| Phase 2 | | |
| kappa | CTC Sugg.(Explicit) | CTC Sugg.(Explicit) |
| 0.86 | 407 | 273 |

Table 2: Statistics of Phase 1 and Phase 2 Annotations

### 4.1 Dataset Preparation

We performed a two phase annotation using both crowdsourced and expert annotations. This reduced the number of statements to be annotated by experts.
**Phase 1 - Crowdsourced Annotations:** The Crowdflower[3] platform was used for crowdsourced annotations. The platform provides a set of management and analytics tools for quality management, as well as for interaction with the annotators. In order to qualify for the annotation task, a Crowdflower worker was required to obtain a score of $7/10$ out of 10 test statements. The annotators were asked to choose one label out of 'Suggestion to Customers' or 'Other Statement' for each sentence. The definition of suggestion was left entirely to the understanding of the annotators. For each sentence, Crowdflower selects the answer with the highest confidence score[4]. We set the system not to seek more than 3 annotations for a statement if one of the labels attained a confidence score of 0.6 or more. At least 3, and at most 5 annotators labeled each statement. Confidence score for each label is the weighted sum of the trust scores of annotators who chose that label[5]. Trust score is determined by annotator's score in the test questions. Table 2 shows the variation in the number of statements tagged as suggestions with the corresponding confidence scores. These suggestions are a mixture of both implicit and explicit types, since the definition of suggestions was not restricted for the annotators. We observed that with the increase of confidence score, the ratio of explicit suggestions among the tagged suggestions increases.

[3] http://www.crowdflower.com/

[4] https://success.crowdflower.com/hc/en-us/articles/201855939-Get-Results-How-to-Calculate-a-Confidence-Score

[5] https://success.crowdflower.com/hc/en-us/articles/202703305-Getting-Started-Glossary-of-Terms

**Phase 2 - Expert Annotations:** Since we target the extraction of explicitly expressed suggestions, two expert annotators further classified the sentences which were finally labelled as 'Suggestion to Customers' by the Crowdflower platform, as explicit CTC suggestions and implicit CTC suggestions. Therefore, the number of sentences which experts annotated was much smaller than the total number of sentences in the two datasets. The key points of annotation guidelines for identifying the explicit suggestions are:

1. The intention of giving a suggestion and the suggested action or recommended entity should be explicitly stated in the sentence. For example:
   $[[\text{Try}]_{intention}[\text{the cup cakes at the bakery next door}]_{entity}]_{action}$.
   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*. Implicit form could be, *The cup cakes from the bakery next door were delicious.*

2. The suggestion should have the intention of benefiting the customers, and should not be mere sarcasm or joke. For example, *If the player doesn't work now, you can run it over with your car.*

A kappa value of 0.86 was calculated between the annotations performed by the expert annotators. The datasets from both phase 1 and phase 2 annotations are freely available for research. The final dataset has 3 kinds of labels, Implicit CTC, Explicit CTC, and others. Therefore, this dataset is also usable by the works which intend to extract implicit suggestions as well.

## 5 Suggestion Detection

We frame the task of CTC suggestion detection as a text classification problem. Our approach performs binary classification in a supervised fashion. Explicit CTC suggestions belong to the positive class, and rest of the sentences to the negative class.

**Heuristic Features:** A general notion about the task is that suggestions contain some distinctive keywords like, *suggest, recommend* etc, and should be easily detectable using them. Therefore, we use a set of manually selected features in order to test this notion.

- **Suggestion keywords:** These include the verbs: *suggestion, advise, request, ask, warn, recommend, do, do not*; their corresponding nouns: *suggestion, advice, request, warning, tip, recommendation*, and the synonyms of these words obtained from WordNet. Suggestion keywords constitute a single feature, which is binary in nature, and its value is determined by the presence of any of the suggestion keywords in the sentence.
- **POS tag *VB*:** Base form of Verb (VB) appears to be frequently used in the predicates of CTC suggestions.

**Generic Features:** Standard uni, bi, tri-grams, and uni, bi, tri-grams of Part of Speech tags (Penn Tree bank tagset). We consider the best performing set of Heuristics and generic feature types as the baseline for this task (see Table 3).

### 5.1 Special Features

We suggest a set of complementary features, which are motivated by the linguistic analysis of explicit CTC suggestions.

1. **Imperative Mood Sequential Patterns:** Imperative mood expressions are often present in explicit suggestions. Wicaksono et. al. (2013) also observed the presence of imperative mood in advice sentences. They used an imperative mood detector, based on a set of handmade rules to determine whether a sentence is imperative.
   In our case, the statements often contain more than one clause, and more than one mood expression. This feature aims to detect if any part of a given statement bears atleast one expression of imperative mood. We aim to identify sequential patterns of linguistic elements (POS tags in our case) which mark the imperative mood, and check if these patterns are present in a given statement. We automatically extract these features (patterns) using sequential pattern mining. We prepare a small dataset of 200 example sentences of imperative mood. These sentences were short sentence of lengths between 3-8 words, and are manually collected from websites related to English grammar, and linguistic research papers on mood and modality.
   **Sequential Pattern Mining - Background:** State of the art sequential pattern mining algorithms require the dataset to be converted into

| Features | Hotel | | | Electronics | | |
|---|---|---|---|---|---|---|
| | Precision | Recall | F score | Precision | Recall | F score |
| Heuristic | 0.260 | 0.484 | 0.338 | 0.216 | 0.505 | 0.303 |
| Unigrams | 0.492 | 0.528 | 0.509 | 0.527 | 0.565 | 0.540 |
| Uni,bi-grams | 0.513 | **0.577** | 0.543 | 0.571 | 0.602 | 0.586 |
| Uni, bi, tri-grams | 0.519 | 0.545 | 0.532 | 0.562 | **0.605** | 0.570 |
| **uni, bi-grams + unigrams POS tags** | 0.539 | 0.555 | **0.547** | 0.634 | 0.565 | **0.595** |
| uni, bi-grams + uni, bi-grams POS tags | 0.568 | 0.491 | 0.527 | **0.662** | 0.515 | 0.575 |
| uni, bi-grams + uni, bi, tri-grams POS tags | **0.593** | 0.469 | 0.524 | 0.625 | 0.518 | 0.556 |

Table 3: Performance of Heuristic and Generic Features in a 10-Fold Cross Validation

| Support | Sequence |
|---|---|
| 0.69 | $< VB >< NN >$ |
| 0.60 | $< VB >< PRP >$ |
| 0.54 | $< VB >< VB >< NN >$ |
| 0.51 | $< RB >< VB >$ |
| 0.70 | $< MD >< VB >$ |

Table 4: Sequential POS Patterns obtained from Imperative Mood Dataset

an ordered list of *events*. An event is a non-empty unordered collection of *items*, which in turn is the smallest unit of sequences. The output of these algorithms are sequential patterns, which comprise of sequences of items, and is defined to be frequent if its support (a measure of frequency) is equal or more than a user-defined threshold. $(a_1, a_2, ..., a_q)$ is a sequence, where $a_i$ is an event. A sequence $(a_1, a_2, ..., a_n)$, is a subsequence of another sequence $(b_1, b_2, ...b_m)$, if there exist integers $i_1 < i_2 < ... < i_n$ such that $a_1 \subseteq b_{i1}$, $a_2 \subseteq b_{i2}, ..., a_n \subseteq b_{in}$. For example, given a sequence (AB,E,ACD), where B is an item and AB, E and ACD are events. (B,AC) is a subsequence, since B $\subseteq$ AB and AC $\subseteq$ ACD and the order of events is preserved. (AB,E) is not a subsequence of the sequence (ABE). In our case, each POS tag is an event, and all the events are of length 1 item.

We use Clospan (Yan et al., 2003) algorithm to obtain the patterns of POS tags appearing in imperative mood sentences. Closed pattern mining algorithms produce a significantly less number of sequences than the older methods, while preserving the same expressive power. We use 2-3 length sequences with a relative support greater than 0.5, where maximum relative support is 1 (see Table 4).

**Sequential patterns and n-grams:** Sequential patterns are different from continuous n-grams in the sense that these are similar to the templates with place-holders, or skip n-grams, where the items in a pattern are ordered but should not necessarily be immediate to each other.

The part of speech tags avoid sparse nature of word based patterns, and at the same time captures the language usage model of imperative sentences. However, the pos tags tend to repeat in longer texts, therefore we limit the number of placeholders which could appear between two items of a pattern to 2. Each pattern is treated as a separate feature, whose feature value is binary depending on its presence or absence in a given sentence.

2. **Sentiment Features:** Given that suggestions are being extracted from customer reviews, which are otherwise mostly used for sentiment analysis, a relation between sentiments and suggestions can be suspected. It can be observed From the Figure 3, 4 suggestions do not seem to always carry one particular sentiment, but different sentiments at different instances. We compare three types of sentiment related features:
a) Manually tagged sentiments: These annotations were provided with the used sentiment analysis datasets.
b) Sentiwordnet score summation: SentiWordNet (Esuli and Sebastiani, 2006) sentiment score summation of all the words in a sentence. No sense disambiguation is performed, and all synset scores are summed up for each word.
c) Normalised sentiwordnet score summation: These scores are the sum of all the sentiment scores of the words in a given sentence, normalised over the number of words carrying non-neutral sentiment score.

3. **Information about the subject/s of a statement:** This feature captures the presence of nsubj dependency (Marneffe and Manning,

| **Features** | **Hotel** | | | | **Electronics** | | | |
|---|---|---|---|---|---|---|---|---|
| | Precision | Recall | F score | AUC | Precision | Recall | F score | AUC |
| Baseline (best generic features) | 0.539 | 0.555 | 0.547 | 0.763 | 0.634 | 0.565 | 0.595 | 0.817 |
| + patterns | 0.542 | 0.511 | 0.526 | 0.743 | 0.607 | 0.604 | 0.616 | 0.790 |
| + sentiments (manual) | 0.529 | 0.536 | 0.532 | 0.754 | 0.581 | **0.630** | 0.605 | 0.797 |
| + sentiments (score) | 0.537 | 0.541 | 0.539 | 0.757 | 0.563 | 0.593 | 0.578 | 0.779 |
| + sentiments (normalised score) | 0.543 | **0.561** | 0.550 | 0.774 | 0.645 | 0.593 | 0.618 | 0.784 |
| + nsubj | 0.559 | 0.538 | 0.548 | 0.757 | 0.597 | 0.586 | 0.591 | 0.778 |
| **Baseline + special (all)** | **0.580** | 0.512 | **0.567** | **0.781** | **0.645** | 0.621 | **0.640** | **0.790** |

Table 5: Performance of Special Features in a 10 Fold Cross Validation

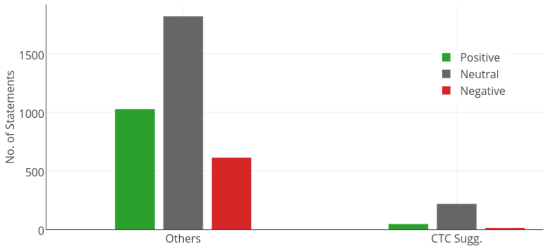


Figure 3: Manually labelled Sentiment Distribution of Electronics Dataset

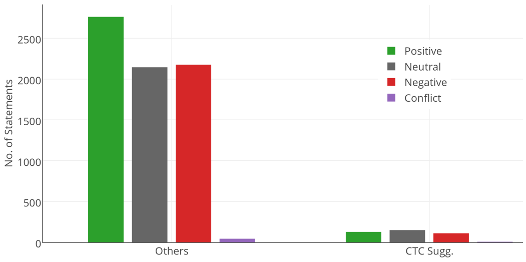


Figure 4: Manually labelled Sentiment Distribution of Hotel Dataset

2008), and if present, the pair of POS tags of the arguments of this dependency. Often a reviewer addresses the reader when giving a suggestion. For the sentence, *If you do end up here, be sure to specify a room at the back of the hotel*, the nsubj dependency is nsubj(do, you). The feature value would be *VBP-PRP* in this case. On the other hand, this suggestion could also have been, *Be sure to specify a room at the back of the hotel.*. In this case the feature value would be *null*. If more than one nsubj dependency is present, the POS pair of each of them will be included in the feature value.

**Experimental Setup:** We use the Stanford Parser (Toutanova et al., 2003) for obtaining part of speech and dependency information. Stemming did not effect the results. Stopwords were used using a customised stopword list. We employ the LibSVM library (Chang and Lin, 2011) for Support Vector Machine classifiers (SVM), as implemented in Weka machine learning toolkit (Hall et al., 2009). The parameter values of SVM classifiers are: SVM type = C-SVC, Kernel Function = Radial Basis Function. A 10 fold cross validation is performed in order to evaluate the classifier model. Features are ranked using the feature selection algorithm InfoGain (Mitchell, 1997). The other attribute selection algorithms provided with the Weka toolkit were also experimented with, but InfoGain consistently performed best in all the runs. Only the features which posses a positive information gain value are retained. The best performing run has a feature vector of size 300. The problem of having a imbalanced dataset is handled by using a higher class weight of 5:1 (Akbani et al., 2004) for positive class.

## 6 Results and Discussion

We evaluate the proposed features using 10-fold cross validation. As indicated in section 5, we consider best performing set of generic features as the baseline, which is: uni, bi grams and unigrams of pos tags. Table 5 summarises the classifier performance with the addition of special features, measured in Precision, Recall, F1 score, and ROC Area Under Curve. The results indicate that special features improve the F score in both the domains. However, some of the generic features produced better precision and recall values (Table 3). Imperative patterns improve the baseline results for electronics dataset, but not for the hotel dataset. Also, the special features produce better improvement over baseline for electronics dataset. We attribute this to the smaller size of electronics dataset.

Table 7 shows how the top ranked features change on addition of special features. Heuristic features tend to appear as some of the top ranked generic features. On the addition of special features, some of the special features replace the top general features. This validates the importance of proposed special features. Normalised sentiment

| # | Sentence | Classifier | Human |
|---|---|---|---|
| 1 | Buy this for the storage and price , avoid it if you know nothing about computers. | 1 | 1 |
| 2 | whichever camera you buy, add upto about 200 dollars for an additional memory card i bought a 256 mb card, usb card reader,camera bag and the warranty. | 0 | 1 |
| 3 | Looks sort of like picasa software (google it if you dont know ) in the interface and is as easy to install and operate as g2 | 0 | 1 |
| 4 | If you cant make an Italian meal don't advertise you can. | 1 | 0 |
| 5 | It would have been better to have some sort of window on the carrying case , so you could see the display without opening it | 1 | 0 |
| 6 | If you have a lot of money to waste, make sure you book this hotel | 1 | 0 |

Table 6: Error Analysis of the Best Performing Feature Set (0 = Negative Class, 1 = Positive Class)

| Type | Domain | Top Features |
|---|---|---|
| Baseline | Hotel | you, VB, if you, recommend, if, want, you are, highly recommend, I would |
| | Elec. | you, VB, recommend, if you, if, buy, recommend this, we, I highly recommend, don't, get |
| Baseline +Special | Hotel | you, VB, $< VB, NN >$, VBD, $< VB, VB, NN >$, recommend, if you, root(Root,VB), norm. sentiment score, suggest |
| | Elec. | you, VB, $< VB, NN >$, recommend, if you, nsubj, buy, suggest, $< VB, PRP >$, norm. sentiment score |

Table 7: Some of the Top Ranked Features in Different Runs

score prove to be better features than the other two types of sentiment features including the manually labelled sentiments. This indicates that this method of sentiment calculation is capturing some universally used phrasing for suggestions, where real sentiment of the sentence fails to capture it.

**Observed Challenges:** Table 6 shows some instances of Type 1 and Type 2 errors in the best performing feature set. Our error analysis reveals the challenges associated with this task.

1. Non-CTC Suggestions: Example #4 gives a suggestion for the improvement of the hotel restaurant. Similarly, #5 is a suggestion to the product manufacturer instead of fellow customers. These kind of suggestions at times possess features similar to CTC suggestions.
2. Complex sentences: Often, suggestion is only expressed in one part of a very long sentence (#2,#3,). This might generate erroneous rank for features; also part of speech taggers tend to perform poorly for such sentences.
3. Sarcasm: The surface form of #6 is a suggestion, but it is a sarcasm.
4. Biased Datasets: Explicit CTC suggestion expressing sentences occur sparsely, which is unfavourable for supervised learning approaches because of increased cost of data annotation task, and imbalanced class representation.

A general observation is that the text in the form of suggestions may not always be a suggestion, and vice versa. Therefore, syntactic and lexical features seem to be ineffective in many cases.

## 7 Conclusion

This work serves as an introduction and analysis of the problem of the extraction of customer-to-customer suggestions from reviews. The task is useful for a number of practical applications. We observed that the layman perception of *suggestion* is very wide, especially in the case of reviews. Therefore, we defined and limited the scope of our work to explicit suggestions. Following this, we prepared a well-investigated benchmark dataset, which is freely available for research purposes.

A pilot approach to the problem is presented, which analyses the performance of standard text classification features, and tests a set of complementary/special features. The special features improved the baseline results for both service and product reviews.

Analysis of classification errors highlighted the challenges associated with the task. The classification results have scope for improvement, and therefore the task calls for advanced semantic features and dedicated models, which will be our future direction. Furthermore, the relation between sentiments and suggestions seem to be worth investigating.

## Acknowledgement

This work has been funded by the European Unions Horizon 2020 programme under grant agreement No 644632 MixedEmotions, and the Science Foundation Ireland under Grant Number SFI/12/RC/2289 (Insight Center).

## References

Rehan Akbani, Stephen Kwek, and Nathalie Japkowicz. 2004. Applying support vector machines to imbalanced datasets. In *Machine Learning: ECML 2004*, Lecture Notes in Computer Science, pages 39–50. Springer Berlin Heidelberg.

Nicholas Asher, Farah Benamara, and Yannick Mathieu. 2009. Appraisal of Opinion Expressions in Discourse. *Lingvistic Investigationes*, 31.2:279–292.

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

Chih-Chung Chang and Chih-Jen Lin. 2011. Libsvm: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, 2(3):27:1–27:27.

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

Andrea Esuli and Fabrizio Sebastiani. 2006. Sentiwordnet: A publicly available lexical resource for opinion mining. In *In Proceedings of the 5th Conference on Language Resources and Evaluation (LREC06*, pages 417–422.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. 2009. The weka data mining software, an update. *SIGKDD Explorations*, 11:10–18.

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*, KDD '04, pages 168–177, New York, NY, USA. ACM.

Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, 5(1):1–167.

Marie-Catherine De Marneffe and Christopher D. Manning. 2008. Stanford typed dependencies manual.

Thomas M. Mitchell. 1997. *Machine Learning*. McGraw-Hill, Inc., New York, NY, USA, 1 edition.

Sapna Negi and Paul Buitelaar. 2015. Curse or boon? presence of subjunctive mood in opinionated text. In *Proceedings of the 11th International Conference on Computational Semantics*, pages 101–106, London, UK, April. Association for Computational Linguistics.

Janardhanan Ramanand, Krishna Bhavsar, and Niranjan Pedanekar. 2010. Wishful thinking - finding suggestions and 'buy' wishes from product reviews. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 54–61, Los Angeles, CA, June. Association for Computational Linguistics.

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *In Proceedings of HLT-NAACL 2003*, pages 252–259.

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, New York, NY, USA. ACM.

Alfan Farizki Wicaksono and Sung-Hyon Myaeng. 2013. Automatic extraction of advice-revealing sentences foradvice mining from online forums. In *K-CAP*, pages 97–104. ACM.

Xifeng Yan, Jiawei Han, and Ramin Afshar. 2003. Clospan: Mining closed sequential patterns in large datasets. In *In SDM*, pages 166–177.