Eatery – A Multi-Aspect Restaurant Rating System

Rrubaa Panchendrarajan, Nazick Ahamed, Prakhash Sivakumar, Brunthavan Murugaiah, Surangika Ranathunga and Akila Pemasiri
Department of Computer Science and Engineering
University of Moratuwa
Katubedda 10400, Sri Lanka
{ruba.11, nazick.11, prakhash.11, bruntha.11, surangika, akila.10}@cse.mrt.ac.lk

ABSTRACT

This paper presents Eatery, a multi-aspect restaurant rating system that identifies rating values for different aspects of a restaurant by means of aspect-level sentiment analysis. Eatery uses a hierarchical taxonomy that represents relationships between various aspects of the restaurant domain that enables finding the sentiment score of an aspect as a composite sentiment score of its sub-aspects. The system consists of a word co-occurrence based technique to identify multiple implicit aspects appearing in a sentence of a review. An improved version of Analytic Hierarchy Process (AHP) is used to obtain weights specific to a restaurant by utilizing the relationships between aspects, which allows finding the composite sentiment score for each aspect in the taxonomy. The system also has the ability to rate individual food items and food categories. An improved version of Single Pass Partition Method (SPPM) is used to categorise food names to obtain food categories.

KEYWORDS

Rating system, aspect-level opinion mining, implicit aspect detection, text categorisation.

1 INTRODUCTION

Entities in a restaurant refer to products (e.g. food), services, individuals (i.e. staff), events, etc. Aspects are the attributes or components of these entities [1]. For example, in the review "food tasted great", *food* is the entity, and *taste* is its aspect. When considering the relationships between different entities, an entity may become an aspect of another entity. For example, *food* is a main aspect of *restaurant* entity. Therefore here onwards we refer both entities and aspects with the term 'aspects'.

HT '17, July 04-07, 2017, Prague, Czech Republic
© 2017 Association for Computing Machinery.
ACM ISBN 978-1-4503-4708-2/17/07…$15.00
http://dx.doi.org/10.1145/3078714.3078737

In the modern era, customers rely on restaurant reviews to choose a better restaurant to dine in. However, reading a lot of reviews and making a conclusion is a tedious process. Therefore it is desirable to process customer reviews and automatically find rating values for restaurants. Nowadays, customers visit a restaurant with different intentions such as having meetings and parties. Therefore they are interested in the ratings for different aspects that are related to their intention of the visit. For example, a set of professionals who wish to select a restaurant for a meeting would be interested in the rating for the aspect *parking*. However, manually going through customer reviews to pick a restaurant based on few of these aspects is a daunting task. Aspect-level sentiment analysis (or opinion mining) has been proposed as a solution for this [2].

The process of aspect-level sentiment analysis includes the identification of different aspects mentioned in the reviews, and sentiment analysis to find the level of polarity of these aspects. An aspect in a review can be categorised as explicit or implicit. Aspects that are literally mentioned in the text are called explicit aspects, whereas the implicit aspects are implied by the review but are not literally mentioned [3]. For example, consider the sentences "Taste of food in that restaurant is great", and "Food is delicious in that restaurant". In the first sentence, aspect *taste* is explicitly mentioned. In the second one, we can infer that the review refers to the aspect *taste*, thus it is an implicit aspect.

In the restaurant domain, aspects exhibit hierarchical relationships. For example, *staff* with sub-aspects *appearance*, *behaviour* and *availability* can be considered as an aspect of *service*, which in turn is one of the major aspects of *restaurant*. Therefore, when calculating the sentiment score for a particular aspect, contribution of its sub-aspects should also be considered. However, this contribution is not uniform across all sub-aspects. For example, a composite sentiment score for a restaurant can be calculated using the rating values of its sub-aspects, *food*, *service*, *ambiance*, etc. However, some aspects can be considered more important than others. For example, if the aspect *food* is more important compared to other aspects, it should be given a higher weight when calculating the composite score for the restaurant. This gives rise to the need of capturing these hierarchical relationships among aspects and the identification of a proper weighting scheme to compute the composite score for each aspect in the hierarchy using the sentiment scores of corresponding sub-aspects.

Similar to the rating value for different aspects of a restaurant, food lovers are usually interested in the ratings for different types of food. However identifying rating values for individual food items may not always be useful. For example, a customer who would like to eat Pizza will be interested in the rating value for the food category Pizza. In contrast, a customer who had tasted a specific type of Pizza is more inclined to refer to that specific type of Pizza in his review. This leads to the problem of categorizing individual food items and identifying rating values for the generic food categories as well, similar to the rating value of other aspects of a restaurant.

This paper presents Eatery, a multi-aspect restaurant rating system as a continuous research of our previous work [4] - [6]. Eatery is based on a hierarchical taxonomy of aspects for the restaurant domain, the first of its kind, according to the best of our knowledge. An improved weighting model using the Analytic Hierarchy Process (AHP) [7] is used to find weights for the aspects in the upper levels of the taxonomy by utilizing the hierarchical relationships between aspects [4]. This allows calculating the ratings for aspects at different levels as a weighted composite score of their sub-aspects, whereas the previous research disregarded these multi-level relationships between aspects and focused only on calculating r1)atings for few aspects such as *food, service, ambience,* and *worthiness* [8] - [13]. Further, this paper discusses an improved method for identifying multiple implicit aspects appearing in a sentence of a review using co-occurrence of words [5], a capability not provided in the related research [14] - [16]. This is a domain-independent technique, which can be used to identify multiple implicit aspects appearing in a sentence that inclines any domain. Our system also includes a new approach for categorising food names in restaurant reviews using an improved version of single pass partitioning method (SPPM) [6]. This allows calculating ratings for individual food items and different food categories, instead of only calculating ratings for individual food items as done in previous research [17], [18].

Rest of the paper is organized as follows. Section 2 presents related work and section 3 describes the data collection process. Section 4 describes the implemented system. Evaluations are given in section 5 and finally, section 6 concludes the paper.

2 RELATED WORK

Currently, many well-known restaurant recommendation and rating systems such as Yelp [19] are available. Swant and Pai [20] have presented a recommendation system that is capable of calculating the rating for a restaurant based on the actual numerical rankings given by customers and recommending a suitable restaurant for a user using clustering algorithms.

A research by Kang et al. [21] introduced a system to predict the level of hygiene of restaurants using customer reviews. It provides a single rating per restaurant based only on the hygiene factor. Ahiladas et al. [17] and Trevisiol et al. [18] have presented approaches that rate individual food items based on the customer reviews. Gupta et al. [8] have focused on summarizing restaurant reviews by attaching the sentiment polarity of a review to three main aspects *food, service,* and *ambience.* An approach by Lu et al. [9] rates main aspects *food, service, ambience* and *prices* using the topic modeling technique. Another approach by Mittal et al. [10] finds ratings for a similar set of aspects. Snyder and Barzilay [11] have used the good grief algorithm to rate multiple aspects in restaurants. In their approach, they consider only the main aspects *food, service, ambience, value,* and ov*erall experience.* Similarly, Govindarajan [12] has focused on finding rating values for *food, service, ambience, deals/discounts* and *worthiness* using a hybrid classification method. A regression-based approach to finding sentiment polarities is introduced by Ganu et al. [13], which focuses on the categories *food, service, price, ambiance, anecdotes,* and *miscellaneous.* It identifies four overall sentiment polarity labels (positive, negative, conflict, neutral) for a given sentence and assigns one or more aspects together with a polarity label for each aspect. However, none of the above research has focused on identifying rating values for all the hierarchically related aspects of a restaurant.

Pontiki et al. [22] have experimented on identifying different aspects expressed in reviews towards a target entity and the sentiment expressed in each aspect. They have evaluated their system for restaurant reviews and laptop reviews. They have separately considered aspect-terms (lower level aspects such as waiter) and aspect categories (higher level aspects such as service) and do the sentiment analysis and find sentiment polarity independently. Therefore the relationship between aspect-term and aspect-category is not utilized here. Pavlopoulos [23] has focused on a similar research that identifies aspect-terms and aggregates them by clustering similar aspect-terms and identifying sentiment polarity for both cluster and individual aspect-term. For example, the aspects *money, price,* and *cost* are clustered together and sentiment polarity is identified for that cluster. Later, Pontiki et al. [24], [25] have extended their work to identify different entities and their aspects and carried out aspect-level sentiment analysis to find the sentiment polarity of each aspect. However, this research does not utilize entity-aspect relationships to identify composite sentiment score for an entity. Cena et al. [26] considered a hierarchy of aspects of a restaurant for extracting opinions on those aspects by means of a tagging framework, where tags are enriched with structure and expressivity. However they do not consider all the possible aspects of a restaurant.

In summary, all this research considers few high-level aspects or low-level aspects only, or both independently while performing aspect-level sentiment analysis in restaurant reviews. Very little research considers even a sub set of the hierarchy of aspects to process structured tags given by users to express opinions on social context. None has focused on utilizing the entity-aspect or entity-entity relationships that can be modeled as a hierarchy of aspects thus enabling sentiment score calculation of an aspect as a composite score of its sub-aspects by performing aspect-level sentiment analysis in restaurant reviews.

3 DATA COLLECTION

3.1 Food Names Collection

A list of more than 200,000 food names extracted from restaurant menus (under a different project) served as the main source. Apart from that, 1400 food names were collected from the A-Z of Food and Drink dictionary [27], and 1300 food names were collected from the Food timeline [28]. As described in section 4.1, these food names were used to obtain food categories.

3.2 Review Collection, Pre-processing, and Annotation

Review Collection & Pre-processing. 990627 restaurant reviews were extracted from the Yelp data challenge [29]. From this set, reviews written in languages other than English were removed with the help of a language detection library [30]. An automatic spell corrector [31] was used to correct language errors in the reviews. The spell corrector algorithm requires a dictionary file that contains the correctly spelled words that are taken as the reference to predict the correct words for the given inputs. This allows the inclusion of domain-specific words in this dictionary file to obtain high accuracy for a specific domain. In Eatery, the dictionary file contains words that are related to the restaurant domain and the words that are frequently cited in the restaurant reviews. We also included many food names that are used in the training data set. This allows correcting spelling mistakes in food names. Apart from these domain-specific words, we also added stop words, adjectives, and adverbs. Spam identification was not considered as the Yelp dataset has already been spam filtered.

Annotation. Previous research [32] suggests that a dataset to train a model to identify aspects in the text should contain at least 15000 sentences for that model to perform well. From the Yelp dataset, 1500 reviews were randomly picked to create the annotated dataset to be used for training and testing. In the training dataset, each review had an average of 10 sentences to ensure that the entire training data set has 15000 sentences. In these 1500 reviews, aspects (both explicit and implicit) were manually labeled. For example, in the sentence "Pizza was small in that big restaurant", *pizza* and *restaurant* are identified as explicit aspects and are labeled as *Food_item* and *Restaurant*, respectively. *small* and *big* are opinion words that identify implicit aspects. Therefore each opinion word that identifies an implicit aspect is labeled with the aspect it implies. For example, in the above sentence, *small* and *big* are labeled as *Food_item_size* and *Environment_size*, respectively. Finally, the annotated sentence appears as follows in the training data set:

<Start: Food_item> Pizza <End> was <Start: Food_item_size> small <End> in that <Start: Environment_size> big <End> <Start: Restaurant> restaurant <End>

Food names are labeled automatically by doing string matching with the collected food names.

4 EATERY SYSTEM

Figure 1 shows the workflow of Eatery that rates different aspects of a restaurant. Categorised food names, trained models to identify explicit and implicit aspects, and weighting model are input to the system. Eatery taxonomy contains the hierarchical relationships among different aspects of a restaurant. A new set

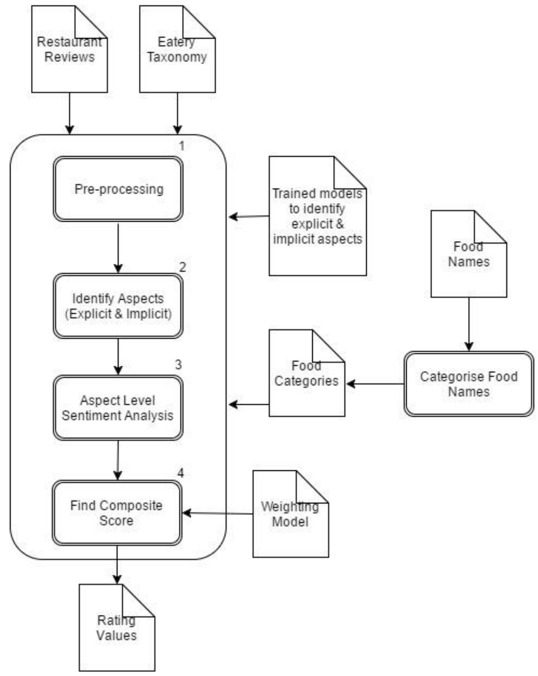

Figure 1: Eatery Flow

of reviews is first pre-processed as described in section 3.2, and the next step identifies explicit and implicit aspects. Individual food items and food categories in the reviews are also identified as explicit aspects.

Once the aspects are identified, sentiment analysis is carried out to find the sentiment polarities of opinion phrases of identified aspects. Scores given by sentiment classification are aggregated to find the rating value for each aspect in the Eatery taxonomy. Using a weighting model, the composite score for each aspect is calculated as a weighted score of its sub-aspects. An improved version of AHP that utilizes the relationship between the aspects is used to obtain the weight for each aspect in the taxonomy. As food items and food categories are considered as a part of the Eatery taxonomy, the composite score for each food category is also calculated as a weighted score of food items that come under that category.

4.1 Food Names Categorisation

Work by Sawant and Pai [20] is the only existing work that categorises food names in restaurants. As we collected only the food names, the data for the food categorisation component in our system is one dimensional. Therefore clustering using algorithms such as k-means as done by Sawant and Pai [20] is not a possible option. Therefore the single pass partitioning (SPPM) [33] text clustering approach was considered as an initial option to categorise food names. SPPM randomly picks an element (food names from the list of food names in our case) as the centroid of a cluster and adds elements to the cluster by measuring the surface similarity between the centroid element and other elements. Jaro [34] distance was used to measure the similarity between two food names. The threshold for the acceptable similarity between two food names to be categorised was decided by manually increasing the threshold till an optimum level of accuracy was achieved.

However, it could be seen that SPPM did not perform well in case of food name categorisation as it considers the entire food name as a single string. For example, consider "Vegetable Burger" and "Chicken Burger". Both food items should be categorized under the food category "Burger". However, since the Jaro distance between "Vegetable" and "Chicken" is large, "Vegetable Burger" and "Chicken Burger" may not get clustered together unless the similarity between these two items exceeded the threshold.

Therefore rather than considering the similarity between entire food names, a set of cluster elements for each food name was created by splitting a food name into multiple words [6]. For example, the food name *Tandoori chicken pizza* is broken into three words *Tandoori*, *chicken,* and *pizza* to create three cluster elements. SPPM was applied for these cluster elements. This allows clustering similar names that refer to the same food item. For example, different users may mention the food *pizza* as *piza*, *pizaa* or *pizzza* in reviews. These similar words are clustered together so that finally a cluster of words represents a food category. As the final step, individual food items are assigned to one of the clusters (food categories) using simple string matching. It is worth to note that in improved SPPM, each cluster element plays the role of a food category whereas a food category cannot be identified in original SPPM without manual effort. Moreover, a very high threshold for an acceptable similarity between two cluster elements is used to avoid clustering of different food names that have high similarity. For example, *pizza* and *pasta* will be clustered together unless a higher threshold is used.

However, due to this modification, the resulting categories had several food category names that do not refer to food names. For example, the food name, *Pizza with cheese* results in a redundant category *with*. Therefore a wiki API[1] was used to determine whether a category name refers to food or not. Each cluster element is given as an input to the wiki API and words in the response are checked against a manually created list of words related to the food domain. For example, if we consider *Chicken pizza*, it is categorised under both *chicken* and *pizza*. When validating the *chicken* cluster by giving the word *chicken* as an input to the wiki API, we get response lines including *chicken, broiler, meat, skin, cooked* and *stewed.* Since the word *cooked* is there in the response, *chicken* is considered as a word related to food and is accepted as a food category. Moreover, it can be noticed that *Pizza with cheese* results in a food category "cheese", which is related to food domain so that it will not be removed by our verification process. However, this is not going to be a useful category. Moreover, the semantic similarity between words is not considered when measuring the similarity between two cluster elements. These are the two limitations that our improved SPPM has at the moment. After completing the categorisation, newly encountered food names in restaurant reviews containing the words in a particular category can be added to that category using simple string matching.

[1] https://www.mediawiki.org/wiki/API:Main_page

4.2 Eatery Taxonomy

Figure 2 shows the taxonomy that was developed to represent hierarchical relationships between different aspects. This taxonomy was developed using a random sample of 400 reviews from the preprocessed reviews. It was again validated and refined using another set of 400 reviews obtained randomly. This process was carried out with 6 human participants. Level 1 is *restaurant*, which has five main categories: *food*, *service*, *ambience*, *discount/offer*, and worthiness in the second level. Level 2 is further categorized and the final level contains the sub-aspects of Level 3 aspects. In addition to the main sub-aspects of *restaurant*, aspects that cannot be categorised under any of the sub-aspects of the *restaurant* are considered as *others*. For example, *overall experience* of a customer cannot be categorised under any of the sub-aspects of *restaurant*. However, since this is one of the important aspects of a *restaurant*, it is categorised under *others*.

Aspects in the Eatery taxonomy pose three different relations, parent-child, siblings, and grandparent-grandchild. Following are examples of these three different relationships:

Food_item_taste <parent-child> Food_item

Food_item_taste <grandparent-grandchild> Food

Food_item_taste <sibling> Food_item_size

These different relationships between aspects are used by different components of our system.

4.3 Aspect Identification

In order to identify explicit and implicit aspects, two models *M1* and *M2* are created using the annotated 1500 reviews. These annotated reviews serve as the training data set for the aspect identification component.

Explicit Aspect Identification. A standard maximum entropy classifier [35] is used to train the model *M1* to identify explicit aspects. Bigrams is used as the feature of the classifier.

Implicit Aspect Identification. This approach is an extended version of Schouten et al.'s [3] approach, which did not have the capability of identifying multiple implicit aspects appearing in a sentence. It also could not successfully identify implicit aspects of a large number of different aspects.

A model *M2* is created to identify the implicit aspects in reviews. Initially, training data is scanned and opinion words that are labeled as implicit aspects are extracted to create the list of opinion words *O*. In the second iteration of scanning, only the sentences with one or more implicit aspects are extracted. Each sentence is stored under each opinion word identified in that sentence, along with the aspect that it is related to. This stored structure defines the model *M2* that gives information of the list of opinion words and the list of aspects that can be implicitly mentioned by a particular opinion word (candidate aspects for an opinion word). For example, consider the sentences "The restaurant was large enough to have a birthday party" and "We had a large pizza". Both sentences appear in the training dataset with an annotated label. They are indexed under the opinion word *large* in the model *M2* as follows:

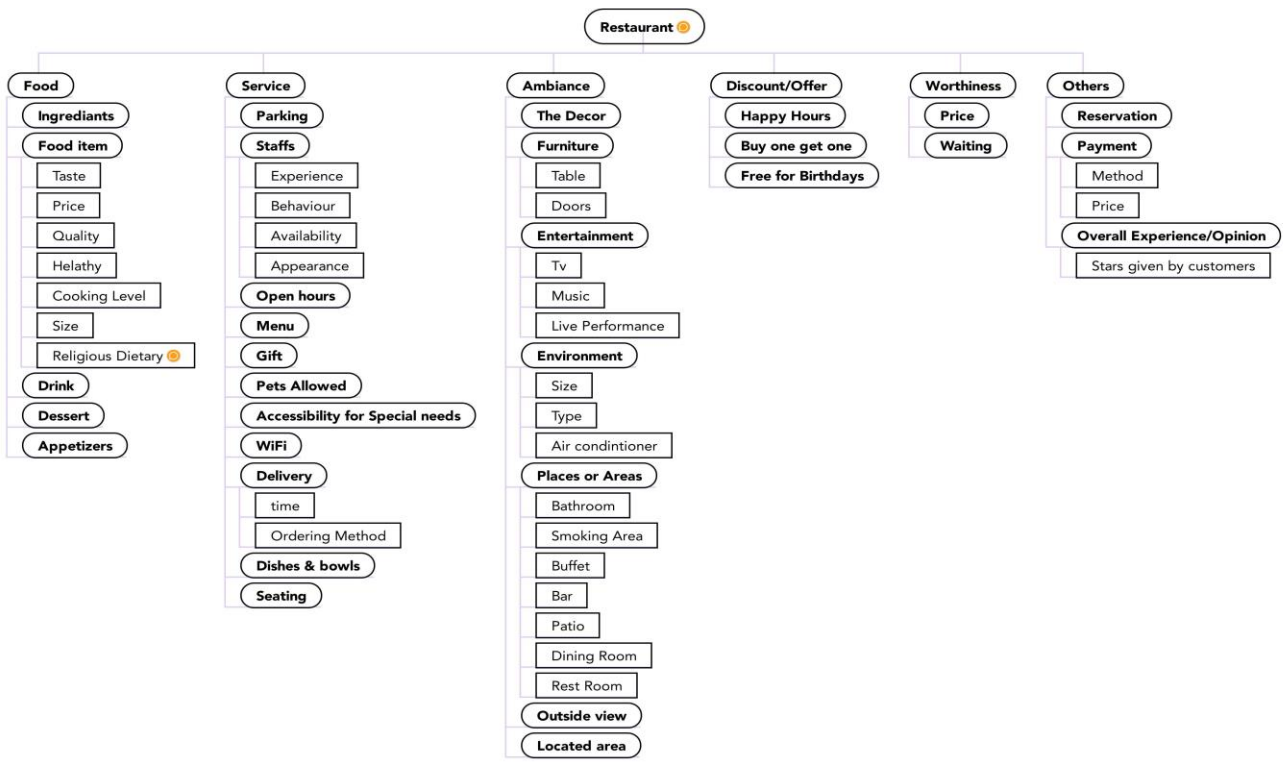

Figure 2: Eatery Taxonomy

large:

Environment_size - The <Start: Restaurant> restaurant <End> was <Start: Environment_size> large <End> enough to have a birthday party

Food_item_size - We had a <Start: Food_item_size> large <End> <Start: Food_item> pizza <End>

When a new review is given, each of its sentences is processed word by word for opinion words available in the opinion list *O*. Processing a sentence word by word allows to identify multiple implicit aspects in a given sentence whereas Schouten et al.'s [3] approach carries out similar approach by processing an entire sentence to identify a single implicit aspect implied by a sentence. For each identified opinion word in a sentence, the list of candidate aspects A is extracted using the model *M1*. For example, if the opinion word *large* is encountered while processing a review sentence, *Environment_size* and *Food_item_size* are listed as candidate aspects. If there is only one candidate aspect, it is chosen as the potential candidate aspect. Otherwise, the score for each candidate aspect is calculated using equation (1). This equation considers the co-occurrence between the opinion word and other words appearing in the input sentence. In equation (1), n is the number of words in the given sentence, A_i is the ith candidate aspect in A for which the score is computed, j represents the jth word in the sentence, C_{ij} is the co-occurrence frequency of aspect A_i and the jth word, f_j is the frequency of the jth word, and d_j is the distance between the jth word and the opinion word which is calculated by counting the number of words that lie between the two strings. $1/d_j$ operates as weight.

$$\text{Score } A_i = 1/n * \Sigma(C_{ij}/f_j * 1/d_j) \qquad (1)$$

This equation is a modified version of the score calculation presented by Schouten et al. [3], where we normalize the co-occurrence by the distance between the opinion word and other words in the sentence, thus removing the impact of faraway words on the sum of co-occurrence. The aspect with the highest score is chosen as the potential candidate aspect. The highest scoring aspect that exceeds the threshold becomes the potential aspect for the next step. If the highest score is lower than the threshold, identified opinion word is discarded. The optimal threshold is identified based on the training data using a simple linear search. The threshold is increased from 0 by a step size of 0.01 until the optimum value for F1-measure is obtained.

Once the potential candidate aspect is chosen, a validation process is carried out to identify one implicit aspect in a given sentence. This, is an addition to the approach suggested by Schouten et al. [8]. Here, opinion target of the potential candidate aspect (aspect on which the opinion is expressed) is extracted to carry out the validation process. For example, in the sentence "Lunch was very expensive", opinion target of the opinion word *expensive* is *lunch*. Opinion targets are extracted using double propagation approach proposed by Qiu et al [7], which propagates information back and forth between opinion words and targets using grammar rules. Extracted target is checked against the Eatery taxonomy to see whether it has any relationship (out of the three relationships) with the potential candidate aspect or not. If the target is the parent aspect, then the potential candidate aspect is chosen as the winning implicit aspect. Otherwise, it is discarded.

The double propagation technique [7] uses the dependency relations mod, pnmod, subj, s, obj, obj2 and desc to define the grammar rules. Dependency between the words in the sentence "The restaurant has good parking" can be explained using the dependency relations as

The restaurant -> *subj* -> has <- *obj* <- parking <- *mod* <- good

The grammar rules are used to carry out the verification process in a sequence as follows:

Verification 1 – Given an opinion word, the target is extracted using grammar rules and is validated to check whether it is the parent or grandparent (only for food hierarchy) aspect of a potential candidate or not in the Eatery taxonomy. Example: in the sentence "Food was delicious", *Food* is identified as the target using the rule *delicious -> mod -> Food*. If *Food_item_taste* is the potential candidate, it is accepted as *Food* is the grandparent of *Food_item_taste*.

Verification 2 – If verification 1 fails, target extracted in verification 1 is used to extract further targets using grammar rules. For example, consider the sentence "Food and dessert are very cheap in that restaurant". When verifying the potential candidate aspect *Food_item_price* for the opinion word *cheap*, dessert is extracted as the opinion target during verification 1. However, it is discarded as it has no relationship with *Food_item_price* in the taxonomy. During verification 2, Food is extracted using the grammar rule *dessert -> conj -> Food* and is verified whether it is the parent/grandparent of the potential candidate aspect.

Verification 3 – This step allows finding further opinion words using grammar rules. For the extracted opinion word, a new potential candidate aspect is identified using model *M2* and it is verified to see whether the earlier winning potential candidate and current candidate are same or are siblings in Eatery taxonomy. For example, consider the sentence "That restaurant was very big and peaceful". When processing the opinion word *big* for the potential candidate aspect *Environment_size*, it is verified in verification 1 and accepted. Using the grammar rule *big -> conj -> peaceful*, *peaceful* is extracted as an opinion word, and model *M2* is used to extract the potential candidate for the opinion word *peaceful* as explained earlier. If *Environment_type* is selected as the potential candidate aspect, it is verified against the earlier winning candidate *Environment_size* and is accepted as both aspects are siblings in the Eatery taxonomy.

This validation process is required since we deal with many aspects at different levels, which leads to ambiguity in identifying the opinion target. For example, consider the sentence "I am a big fan of that restaurant". Here, "I" is identified as the opinion target of the opinion word *big* with the prediction of either *Food_item_size* or *Environment_size*. If *Environment_size* or *Food_item_size* is chosen as the potential candidate aspect with the highest score, it is discarded as its opinion target "I" has no relationships with the potential candidate aspect in Eatery taxonomy.

4.4 Sentiment Analysis

Once both explicit and implicit aspects are identified, the system calculates the sentiment scores related to those aspects by doing sentiment analysis at aspect-level where the sentiment score for each individual aspect is calculated.

In restaurant reviews, most of the sentences contain more than one aspect. Therefore the sentences are split in such a way that each phrase contains one aspect and the related opinion phrase, using the typed dependency engine designed by Ahiladas et al. [17]. This approach uses the grammatical relationships of words to extract the opinion words and the other related words for each and every aspect identified in a sentence. Once the opinion phrases are identified, sentiment orientation that the corresponding aspects have in the sentences is analyzed using a recursive neural sensor network [36]. A sentence is classified into 5 polarity classes: very negative (1), negative (2), neutral (3), positive (4) and very positive (5).

Every time when an aspect occurs in the review, the sentiment score for that aspect is calculated for that review and is accumulated with the previous scores. In order to accumulate the scores, the lower bound on normal confidence interval method [37] is used with 95% of confidence level. This method considers both the rating value and the number of occurrences to calculate the aggregate ratings. This characteristic is significant in restaurant reviews since it is preferred to have a higher rating for an aspect that has more number of positive ratings and similarly, lower rating for an aspect that has more negative ratings.

4.5 Composition of Scores Using the Weighting Model

At the end of the sentiment analysis process, the system contains individual ratings for each and every aspect in the aspect hierarchy. Next process is to calculate a composite score for the parent aspects using their individual scores and the scores of the corresponding sub-aspects. For this, an improved version of the AHP method [38] was used by improving the creation of pairwise matrix of AHP that utilizes the relationships between aspects [6].

Analytic Hierarchy Process (AHP). Analytic Hierarchy Process (AHP) [38], proposed by Saaty is one of the well-known methods for weight estimation. AHP works as follows. When calculating the weights for *n* attributes, an *nxn* pairwise matrix *A* is created as the initial step. Each entry a_{ij} in the pairwise matrix *A* represents the relative importance of the i^{th} attribute compared to the j^{th} attribute that satisfies the following condition:

$$a_{ij} = 1/a_{ij} \qquad (2)$$

Identifying the relative importance between two attributes is a manual task using Saaty's scale definition as given in Table 1. Upper triangular part of the pairwise matrix is filled using Saaty's scale definition and the rest of the matrix is filled using the condition given in equation (2). It is obvious that $a_{ii} = 1$. Once the pairwise matrix is created, it is normalized through the columns as shown in equation (3):

$$a'_{ij} = a_{ij}\Sigma a_{kj} \qquad (3)$$

6

where k = 1, 2, 3 …. n. Finally, the weight for each attribute is calculated by taking the average of the normalized values using the normalized matrix as shown in equation (4):

$$W_i = (\Sigma a'_{ik})/n \qquad (4)$$

Table 1: The Saaty Scale Definition

Insensitivity of Importance	Definition
1	Equal Importance
3	Moderate Importance
5	Strong Importance
7	Very Strong Importance
9	Extreme Importance
2, 4, 6, 8	Can be used to express intermediate values

As computing the relative importance between two aspects in AHP is a manual task, there can be inconsistencies between two relative importance values. For example, consider the following pairwise matrix,

$$A = \begin{pmatrix} 1 & 3 & 1/3 \\ 1/3 & 1 & 3 \\ 3 & 1/3 & 1 \end{pmatrix}$$

Here, the first attribute is 3 times important than the second attribute, and 1/3 times important than the third attribute. Thus relative importance of the third attribute compared to the second attribute should be nearly 9 in order to maintain the consistency of the pairwise matrix. However, the relative importance of the third attribute compared to the second attribute is 1/3 in the above example, which is not consistent. Therefore it is essential to measure the consistency of a pairwise comparison matrix. For that, Consistency Ratio (CR) [38] of a pairwise matrix is calculated as follows:

$$CR = CI/RCI \qquad (5)$$

Here, *CI* is the Consistency Index and *RCI* is Random Consistency Index, which is the average CI of randomly generated reciprocal matrices with dimension n [38]. Consistency Index of a pairwise matrix is defined as follows:

$$CI = (\lambda_{max} - n)/(n-1) \qquad (6)$$

Where λ_{max} is the highest Eigenvalue for the pairwise comparison matrix and n is the dimension of the pairwise matrix. The pairwise comparison matrix is accepted if the consistency ratio *CR* is less than 10%. Otherwise, we can conclude that the in-consistency is too large so the pairwise comparison matrix values should be revised.

Improved AHP. Original AHP requires manual work to identify the relative importance of a set of attributes, thus it does not utilize any relationship between those attributes to obtain the relative importance. Therefore we introduced an improved version of AHP that utilizes the relationship between aspects to obtain the weights for the set of aspects [4]. This improved AHP is applied to each non-leaf node in the hierarchy to obtain the weights for its sub-aspects. Along with the sub-aspects, weight is obtained for the parent as well. Finally, each non-leaf aspect gets two weights: one as a parent and one as a child. Each leaf aspect gets a single weight. For example, composite rating for *staff* is obtained as follows,

$$\text{Composite score for staff} = W_{experience}{}^*R_{experience} + W_{behaviour}{}^*R_{behaviour} + W_{appearance}{}^*R_{appearance} + W_{availability}{}^*R_{availability} + W'_{staff}{}^*R_{staff} \qquad (7)$$

where W represents the weight of an aspect as a child node, W' represents the weight of an aspect as a parent, and R represents the rating value of an aspect obtained using aspect level sentiment analysis.

Improved AHP works as follows. For a particular non-leaf aspect, a pairwise comparison matrix A is created with the dimensions of $n x n$ where n is the total number of sub-aspects + 1 for the parent aspect. For each aspect, the occurrence of an aspect in the training data set is counted. If an aspect is a parent for which the pairwise comparison matrix is created, the occurrence of an aspect is simply the number of explicit and implicit aspect labels in the training data set. If the aspect is a sub-aspect, the occurrence of an aspect is obtained by adding up the occurrences of that aspect and all its sub-aspects (both direct and indirect sub-aspects). This enables to utilize the relationship between aspects in such a way that the occurrence of an aspect is obtained by considering the occurrence of all its direct and indirect sub-aspects.

Relative importance between two aspects is obtained as the ratio of occurrence of aspects and is used as an element in the pairwise comparison matrix. Rest of the AHP processes as the usual flow [38] with this improved pairwise comparison matrix to obtain the weights for n + 1 aspects. It is worth to note that AHP is applied to each non-leaf aspect in the Taxonomy separately for different restaurants so that the resulting weights are specific to a particular restaurant. Moreover, this improved AHP can be applied to any domain where the aspects of that particular domain can be modeled as a hierarchy.

5 EVALUATION

5.1 Preprocessing

Spell correction. Evaluation for spell correction of the pre-processing step was carried out with 800 manually misspelled words related to restaurant domain. With the default dictionary file that is used by the Peter Norvig algorithm [30], we obtained an accuracy of 57.35%. With the inclusion of restaurant domain-specific words to the dictionary as explained in section 4.1, the accuracy was increased up to 85.25%.

5.2 Aspect Identification

This component of the system was evaluated with the annotated 1500 reviews as the training data set using 10 fold cross validation. For each instance of the algorithm, 1400 reviews are used as the training dataset and remaining 100 reviews are used for testing. Figure 3 shows the distribution of aspects in 1000 reviews randomly picked from the training data set. The

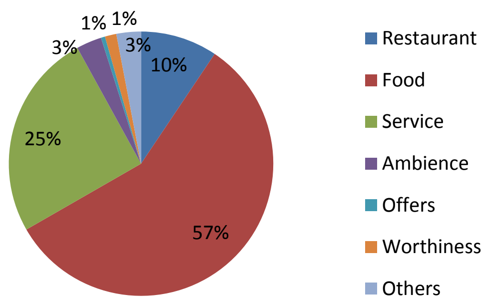

Figure 3: Distribution of Level 1 and Level 2 aspects in 1000 reviews

occurrence of level 3 and level 4 aspects are counted as the occurrence of level 2 aspect to show the distribution up to 2 levels. It can be observed that the aspects *food* and *ambience* are highly mentioned in restaurant reviews both directly and indirectly.

Explicit Aspect Identification. Model *M1* explained in section 4.3 was evaluated and a precision and recall of 0.9317 and 0.8348 (respectively) were obtained for this evaluation. Thus this gave an F1-measure of 0.88 for model *M1*.

The evaluation of individual aspect identification figures out how the above evaluation result is distributed among individual aspects. This evaluation is done for all the aspects in the Eatery taxonomy, and Table 2 shows the results for the main aspects *Food*, *Service*, *Ambience*, *Worthiness* and *Others*. These evaluations consider the identification of that particular aspect (not their sub-aspects). Apart from these, we evaluated for the restaurant aspect as well.

It can be observed that the identification of three main popular aspects *food*, *service* and *ambience* give similar evaluation results as model *M1* (F1-meausre of model *M1* and F1-measure of the three main aspects). Here the results for the *Worthiness* and *Others* deviate from other aspects since the number of occurrences of these aspects in the reviews is much lower compared to other aspect verticals.

Table 2: Evaluation results for the individual aspects in explicit aspect extraction

Aspect	Precision	Recall	F1- measure
Restaurant	0.6493	0.9615	0.7751
Food	0.7369	0.9533	0.8312
Service	0.7245	0.9890	0.8364
Ambience	0.7241	0.9292	0.8139
Worthiness	0.0625	0.3333	0.1052
Others	0.9189	1.0000	0.9577

Implicit Aspect Identification. Even though restaurant domain deals with many aspects, not all the sentences contain implicit aspects. In 1000 reviews (each review contains an average of 10 sentences) picked randomly from the training data set, 15.6% of the sentences contain one or more implicit aspects. However, it is essential to identify that small fraction of implicit aspects as some of the important aspects are most likely to appear implicitly in customer reviews. For example, 92% of each sub-aspects (behavior, experience, appearance and availability) of staff aspect were found to be implicit in a randomly picked set of 1000 reviews from the training data set.

Table 3 shows the evaluation results of 10-fold-cross validation for several methods. Methods 1 to 3 use the annotated explicit aspects so that the accuracy of model *M1* does not impact model *M2*. Method 4 shows the results when both *M1* and *M2* are used to obtain the explicit and implicit aspects, respectively. Method 2 extends Schouten et al.'s [3] approach to identify multiple implicit aspects in a sentence. This extended approach does not have a validation process. Method 3 extends Method 2 by validating the potential candidate using opinion target extraction as explained in section 4.3.

It can be seen in Table 3 that our approach gives the best result. Moreover, extending the approach suggested by Schouten et al. [3] (Method 2) fails in the case of identifying a large number of interrelated implicit aspects. Therefore adding potential candidate validation to that method improves precision from 0.49 to 0.91. The result for our approach is slightly higher than this (Method 3), as our approach considers the distance between opinion words and other words in the sentence.

Table 3: Evaluation results for implicit aspect extraction

	Method	Precision	Recall	F1 Measure
1	Our solution	0.947	0.758	0.842
2	Method 2	0.495	0.929	0.645
3	Method 3 - Method 2 with validation process	0.916	0.752	0.826
4	Our solution with trained model *M1*	0.886	0.694	0.779

Table 4 shows the evaluation results for 10-fold-cross validation of the model *M2* for sentences with more than one aspect. It can be observed that the F1-Measure is above 0.82.

Table 4: Evaluation results for sentences with multiple implicit aspects

	Method	Precision	Recall	F1-Measure
1.	Sentences with two	0.978	0.709	0.822
2.	Sentences with more than two	0.975	0.725	0.832

5.3 Sentiment Analysis

A separate set of 400 random reviews from the Yelp dataset was used as the test set for this component and two types of evaluations were carried out.

Evaluation of Aspect-Opinion relationship in opinion phrases: In this evaluation, it is checked whether the relevant opinion words are correctly associated with the aspects in the opinion phrase. An average accuracy of 71.82% was obtained for this evaluation.

Evaluation of sentiment analysis: Reviews in the test data set were split into small phrases and are manually marked as (P)/ neutral (O)/ negative (N) in the sentiment polarity. The output of the sentiment analysis tool was compared against the manually given polarity. An average accuracy of 72.55 was obtained for this evaluation. Here the training data set used for sentiment analysis is a general corpus associated with the sentiment analysis tool [36]. This accounts for the drop in the accuracy in this system since it does not contain any opinions specific to restaurant domain.

5.4 Weighting Model

Improved AHP approach explained in section 4.5 is executed by giving 1500 annotated reviews as the input, and weights for all the aspects in our Eatery taxonomy is calculated. Table 5 shows the consistency ratio for three pairwise matrices of a random restaurant. It can be seen that all values are very close to zero so that all three pairwise comparison matrices are acceptable. Weights obtained for the sub-aspects of that restaurant using our approach are shown in Table 6.

Table 5: Consistency ratio for three different levels of pair wise matrices

Parent Aspect	Consistency Ratio (CR)
Restaurant	2.2428 × 10−14
Service	0.0
Staff	0.0

Table 6: Weights for sub-aspects of restaurant

Aspect	Weight
Restaurant	0.0943
Food	0.5734
Service	0.2534
Ambience	0.0308
Offers/Discount	0.0048
Worthiness	0.0135
Others	0.0296

5.4 Food Name Categorisation

Evaluating improved SPPM: 5 sets of random samples, each with 100 food names were selected from the dataset and were manually categorized to be used as the test data set. An average accuracy of 90% was obtained for the evaluation of the food categorisation component. However, it was taking high execution time as the individual word processing is done with wiki API to identify the actual food names.

Another drawback of this approach is the resulting categorisation containing redundant categories (i.e. same word falling into multiple categories. E.g. "Tandoori chicken pizza" categorized under both *chicken* and *pizza*). In order to evaluate the level of redundancy in every category, another 5 random samples of 500 food names were used. Those samples were again categorized manually and also by the improved SPPM method, and then the redundancy test was done. It is found that the average redundancy is around 25%.

Categorizing food names appearing in restaurant reviews: Annotated 1500 reviews were used as the test data set, which contained the food names annotated as explicit aspects. An average accuracy of 82.6% was obtained for this evaluation.

5.5 Evaluation of the overall system

For the overall system evaluation, two restaurants were randomly picked and 400 reviews for each restaurant were randomly obtained from Yelp reviews as test data. For the selected restaurants, ratings were given manually by human judges for four aspects from four different levels of the taxonomy. At the end of the manual scoring, these four aspects had their individual scores, which are then compared with the corresponding score calculated by the system. The evaluation results are shown in Figure 4 and Figure 5. It can be seen that the results from the system are relatively similar to the actual values given by manual judges with small variations. The main reason for the system values being low compared to the manual values is the accuracy drop caused by the general corpus used for sentiment analysis.

In order to measure the consistency among the annotators, three data sets, each with 100 reviews were used. Each set was tagged by two different annotators. Two types of measure of consistency were computed, absolute agreement, and the Kappa coefficient [39]. Result for the former is 0.917, and the result for the latter is 0.834 (a fair agreement).

6 CONCLUSION

This paper presented Eatery, a multi-aspect restaurant rating system. This research introduced a new taxonomy to the restaurant domain that captures the hierarchical relationships among entities and aspects. It also contains a novel approach to find multiple implicit aspects appearing in a sentence, a new food categorisation technique, and a weighting model that helps calculate the sentiment score of an aspect as a composite of the sentiment scores of its sub-aspects. However, the entire system can be applied to other domains as well, where the aspects of those domains are modeled as a hierarchy.

As future work, sentiment analysis component should be enhanced by using a corpus specific to the restaurant domain and identifying implicit opinion which is a limitation in current sentiment analysis approach. It would be interesting to extend this work dynamically to improve the Eatery taxonomy, as new aspects are found while processing reviews.

Figure 4: Results for Restaurant 1

Figure 5: Results for Restaurant 2

REFERENCES

[1] Zhang, L. and Liu, B. Aspect and entity extraction for opinion mining. In Data mining and knowledge discovery for big data, (2014), Springer, 1-40.

[2] Sharma, R., Nigam, S. and Jain, R. Mining of product reviews at aspect level. *International Journal in Foundations of Computer Science and Technology*, 4(3). (2014).

[3] Schouten, K., de Boer, N., Lam, T., van Leeuwen, M., van Luijk and R., Frasincar, F. Semantics-driven implicit aspect detection in consumer reviews. In *24th International Conference on World Wide Web Companion*, (2015), 109-110.

[4] Panchendrarajan R., Murugaiah, B., Sivakumar, S., Ahamed, M.N.N., Ranathunga and S., Pemasiri, A. Cheap food or friendly staff? Weighting hierarchical aspects in the restaurant domain. In *2nd Moratuwa Engineering Research Conference*, (IEEE, 2016), 24-49.

[5] Panchendrarajan R., Ahamed, N., Murugaiah, B., Sivakumar, S., Ranathunga, S. and Pemasiri, A. Implicit Aspect Detection in Restaurant Reviews using Cooccurence of Words. In *7th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, (ACL, 2016), 128-136.

[6] Sivakumar, P., Ahamed, N., Panchendrarajan, R., Murugaiah, B.,Ranathunga and S., and Pemasiri, A. Categorizing food names in restaurant reviews. In *2nd Moratuwa Engineering Research Conference*, (IEEE, 2016), 1-5.

[7] Qiu, G., Liu, B., Bu, J. and Chen, C. Opinion word expansion and target extraction through double propagation. *Computational linguistics* 37(1), 9-27, (2011).

[8] Gupta, A., Tenneti, T. and Gupta A., Sentiment based Summarization of Restaurant Reviews. *Final Year Project.* (2009).

[9] Lu, B., Ott, M., Cardie, C. and Tsou, B. K. Multi-aspect sentiment analysis with topic models. In *11th IEEE International Conference on Data Mining Workshops*, (IEEE, 2011). 81-88.

[10] Mittal, N., Agarwal, B., Laddha, S. and Sharma, M. Aspect Based Analysis for Rating Prediction of the Restaurant Reviews. In *International Journal of Computer Systems*, 59-62. (2015).

[11] Snyder, B. and Barzilay, R. Multiple aspect ranking using the good grief algorithm. In *Joint Human Language Technology/North American Chapter of the ACL*, (ACL, 2007). 300-307.

[12] Govindarajan, M. Sentiment Analysis of Restaurant Reviews using Hybrid Classification Method. In *International Journal of Soft Computing and Artificial Intelligence*, 2(1). (2014).

[13] Ganu, G., Elhadad, N. and Marian, A. Beyond the Stars: Improving Rating Predictions using Review Text Content. In *12th International Workshop on the Web and Databases*, (2009). 1-6.

[14] Schouten, K.. and Frasincar, F. Finding implicit features in consumer reviews for sentiment analysis. In *Web Engineering*, (Springer, 2014). 130-144.

[15] Wang, W., Xu, H. and Wan, W. Implicit feature identification via hybrid association rule mining. In *Expert Systems with Applications*, 40(9), 3518-3531 (2013).

[16] Zhang, Y. and Zhu, W. Extracting implicit features in online customer reviews for opinion mining. In *22nd international conference on World Wide Web companion*, (2013). 103-104.

[17] Ahiladas, B., Saravanaperumal, P., Balachandran, S., Sripalan, T. and Ranathunga, S. Ruchi: Rating individual food items in restaurant reviews. In *12th International Conference on Natural Language Processing*, (2015).

[18] Trevisiol, M., Chiarandini, L. and Baeza-Yates, R. Buon appetito: recommending personalized menus. In *25th ACM conference on Hypertext and social media*, (ACM, 2014). 327-329.

[19] Yelp Inc. Yelp. http://www.yelp.com/, 2017. Web. 17 Feb. 2017

[20] Sawant, S. and Pai, G. Yelp food recommendation system. http://cs229.stanford.edu/proj2013/SawantPai-YelpFoodRecommendationSystem.pdf (2016).

[21] Kang, J. S., Kuznetsova, P., Luca, M. and Choi, Y. Where Not to Eat? Improving Public Policy by Predicting Hygiene Inspections Using Online Reviews. In *Empirical Methods in Natural Language Processing*, (2013). 58-77.

[22] Pontiki, M., Galanis, D., Pavlopoulos, J., Papageorgiou, H., Androutsopoulos, I., and Manandhar, S. Semeval-2014 task 4: Aspect based sentiment analysis. In *8th International Workshop on Semantic Evaluation*, (SemEval 2014), 27–35.

[23] Pavlopoulos, J. Aspect based sentiment analysis. Ph.D. thesis, Dept. of Informatics, Athens University of Economics and Business, Greece. (2014).

[24] Pontiki, M., Galanis, D., Papageorgiou, H., Manandhar, S., and Androutsopoulos, I. Semeval-2015 task 12: Aspect based sentiment analysis. In *9th International Workshop on Semantic Evaluation*, (SemEval 2015).

[25] Pontiki, M., Galanis D., Papageorgiou, H., Androutsopoulos, I., Manandhar, S.,
AL-Smadi, M., Al-Ayyoub, M., Zhao., Y., Qin, B., Clercq, O., Hoste, V., Apidianaki, M., Tannier, X., Loukachevitch, N., Kotelnikov, E., Bel, N., Jiménez-Zafra, S. M., Eryigit, G., . Semeval-2016 task 5: Aspect based sentiment analysis. In *10th International Workshop on Semantic Evaluation*, (SemEval 2016).

[26] Cena, F., Likavec, S., Lombardi, I. and Picardi, C., Synthesis of Collective Tag-Based Opinions in the Social Web. In *XIIth International Conference on AI*IA 2011: Artificial Intelligence Around Man and Beyond*, (ACM, 2011), 286-298.

[27] Ayto, J. An AZ of food and drink. Oxford University Press (2004)

[28] Food timeline: Food timeline food list. http://www.foodtimeline.org/foodfaqindex.html, 2017. Web. 17 Feb. 2017

[29] Yelp Inc.: Yelp data challenge. https://www.yelp.com/dataset_challenge/dataset, 2017. Web. 17 Feb. 2017

[30] GitHub, optimaize/language-detector, https://github.com/optimaize/language-detector, 2017. Web. 17 Feb. 2017

[31] Norvig, P.: How to write a spelling corrector. http://norvig.com/spell-correct, 2017. Web. 17 Feb. 2017

[32] Apache OpenNLP Developer Documentation: opennlp. https://opennlp.apache.org/documentation/manual/opennlp.html, 2017. Web. 17 Feb. 2017

[33] Gupta, R. and Satsangi, C. An efficient range partitioning method for finding frequent patterns from huge database. In *International Journal of Advanced Computer Research*, 2(2), 62-69. (2012).

[34] M Pilar. An Espino Comparison of methods Hamming Distance, Jaro, and Monge-Elkan. In 7th International Conference on Advances in Databases, Knowledge, and Data Applications , (2015).

[35] Apache Software Foundation.: Opennlp. https://opennlp.apache.org/, 2017. Web. 17 Feb. 2017

[36] Socher, R., Perelygin, A., Wu, J.Y., Chuang, J., Manning, C. D., Ng, A.Y. and Potts, C. Recursive deep models for semantic compositionality over a sentiment treebank. *In Empirical methods in natural language processing*, (2013).

[37] McGlohon, M., Glance, N. and Reiter, Z. Star Quality: Aggregating Reviews to Rank Products and Merchants. In *Fourth International Conference on Weblogs and Social Media (ICWSM)*, (2010).

[38] Saaty, T.L. What is the analytic hierarchy process?. *Mathematical Models for Decision Support*, (Springer, 1988). 48, 109-121.

[39] Carletta, J. Assessing agreement on classification tasks: The Kappa statistic. *Computational Linguistics*, 22(2), 249-254. (1996)