Recommendation Model for Infrequent Items

Pramod Kumar Singh and Nidhi Gupta

Computational Intelligence and Data Mining Research Laboratory
ABV - Indian Institute of Information Technology and Management Gwalior
Gwalior, MP, India
pksingh@iiitm.ac.in, nidhsg16@gmail.com

Abstract. Recommender system is an integral part of the e-commerce business to recommend users the items of their interest. Collaborative filtering and content based approaches are traditional and successful methods for recommendation in case of frequent items and users as understanding user-item matrix is easy. However, in case of infrequent items, as users buy items rarely, sufficient information about the users is not available. Therefore, it is tough to recommend items to the users using these traditional approaches. We consider it as new-user cold start problem. Moreover, new items are regularly introduced to the catalogue which should be recommended to the users as per their likings. However, sufficient information about the items is not available; it is a new-item cold start problem. Therefore, we view recommendation of infrequent items as having two parts new-user/old-item cold start problem and new-user/new-item cold start problem and present a recommendation model for the same. An empirical analysis on the mobile handsets dataset indicates that our proposed model is better than the traditional models.

Keywords: Infrequent items, Feature-based recommendation, Customer reviews, Multi-criteria ratings

1 Introduction

The objective of a recommender system is to predict rating of an item to a user (prediction of liking/disliking of an item by a user) or to recommend items that may be of interest to the user (top-N recommendation problem) studying the users' profiles or the items' popularity. Collaborative Filtering (CF) methods are among the most successful and widely used methods of recommender system [1]. The fundamental principle of CF is that if users A and B rate some items similarly in the past or have similar behaviors in buying or reviewing items then it is most likely that they will act similarly on other items too. Thus, we find the similar users (neighbours) to a certain user using N-nearest neighbor and recommend the items bought and reviewed by the neighbours. Content-based recommender systems [2] analyse the textual information and features of users and items and recommend by finding patterns in the content. Both these approaches depends on enough history of users and items and the corresponding ratings to recommend efficiently. Many recommender systems also use hybrid systems which combine both the approaches [3].

In many scenarios, we do not have sufficient information about the users and/or the items popularity. It is called a cold start problem. One such scenario is when we have to recommend infrequent items, i.e., the items which are bought rarely. In this case, it is not possible to obtain enough information about the users and their preferences. Recommending new items is also an example of the cold start problem as we have no information about the items' popularity. Therefore, the cold start problem can be new-user, new-item or new-user/new-item depending on which information is missing. The cold start problem is a big challenge to the recommender systems as the absence of information regarding the users and/or items makes it difficult to predict accurate recommendations to the users. It affects the coverage of recommender systems as recommendations skip the new users and the new items. Therefore, various methods have been proposed to gather information from different sources and produce effective recommendations.

In this work, we deal with the cold-start problem of recommendation of infrequent items. Unlike the case of frequent items, in this case user-item rating matrix is not available because enough history of the user is not available as she/he buys such items rarely. Hence, we treat such user as a new user. Moreover, the new items are regularly introduced in the catalogue. Therefore, recommendation of infrequent items can be seen as a cold-start problem having two parts. First, it may be viewed as new-user/old-item problem for recommending old items which have been rated by other users but user-profile information of the target user for whom recommendation needs to be made is not available. Second, it may be seen as new-user/new-item cold start problem where we do not have information regarding either the item's popularity or the user's preference. In both the cases, the user is considered as a new user.

We solved first part of the problem assuming that the popular items shall also be preferred by the new user for whom we do not have profile information. We built a user interface for asking her/his preferences of the features. We do feature level sentiment analysis of the reviews of every item and recommend items which are rated highly on the features preferred by the user. For recommending new items, we propose to predict the rating of new items based on their feature level similarity to the old items. The more similar the features of new item are to the popular and highly rated features, the more popular the new item shall be among the users. We consider feature level ratings of the old items into account in our proposed mathematical model opposed to the overall rating of the items given in similarity based recommendation models [4]. We compare the efficiency of recommender system by measuring root mean square error (RMSE) to measure the average difference between the actual ratings and the estimated ratings computed by both the models. The RMSE values for our proposed model are comparatively better; it indicates that the predicted ratings are more close to the actual ratings.

Rest of the paper is structured as follows. Section 2 recalls literature for solving the cold start problem. An overview of the feature level sentiment analysis is presented in Section 3. Section 4 illustrates the proposed methodology and Section 5 explains the empirical analysis. Conclusions is presented in Section 6 and finally Section 7 completes the paper with future work.

2 Related Work

Melville et al. [5] present content-boosted collaborative filtering which is a hybrid approach of collaborative filtering and content-based method. They make user-item matrix denser using content-based filtering by generating default ratings of items which are not rated and then use traditional collaborative filtering approach in this denser matrix. They show that their proposed method performs better than pure content-based method, pure collaborative filter, and naive hybrid approach. Schein et al. [6] propose Hofmann's aspect model to combine items' contents and users' ratings under a single probabilistic framework.

Both the above research work propose hybrid approaches by combining two or more traditional recommendation methods to improve the efficiency, predictability and coverage of recommendation in the cold-start scenario. Prime objective of these hybrid methods is to improve the prediction accuracy by using multiple data over all the users by generating default ratings. However, these methods deal with the cold-start problem in a scenario where the user has rated at least few items in order to generate default ratings but fail to recommend in case of new users and/or new items.

Aciar et al. [7] collect relevant customer reviews and apply content mining method to separate significant data from the reviews. They employ an ontology to translate sentiment quality and substance into a format that the recommender method can use. Unfortunately, they do not evaluate the efficiency of recommendations. Additionally, they also do not explore other possibilities for enhancing recommendation quality like observing reviewers' preferences of features from the result of sentiment analysis. Guy et al. [8] gather information about the user's preferences from social networks to solve the cold-start problem due to new-user. Iaquinta and Someraro [9] present a hybrid method for cold start problem based on the content-based approach as it is less sensitive to the cold start issue in comparison to the collaborative filtering.

Bykau et al. [4] present three methods to address new-user/new-item cold start problem. The first method, Similarity-based Recommendation, rates a new item based on its similarity to the old items. The second method, Feature-based Recommendation, is based on the independence of features where ratings of features of a new item are predicted individually. The third method, Max Entropy-based Recommendation, is based on max entropy for the new-user/new-item cold start problem with the assumption that there is absolutely no form of external auxiliary information. The main drawback of these approaches is that they do not consider feature level ratings. As the above approaches involve overall ratings of the items, they fail to acknowledge the fact that similarity between the items can exist because of one subset of features whereas the rating is highly influenced by another subset of features and these subsets may be disjoint or may have very less overlapping.

To predict rating for a new item, we assume that more similar a new item is to the popular old items, the more popular it will be. In our recommender model, for new-user/old-item, we propose to explicitly input users' preferences of features by providing a preference form to the user. We extracted feature-level sentiment results from the reviews for each item. Thus instead of recommending item with overall high rating, we recommend items which are rated high on the features important to the user. For recommending new item, we propose a formula for predicting the rating of the new item. Our formula ensures that only those features of the old items which are similar to the features of new item influence the predicted rating of new item. We observe from our experimental findings that our predicted ratings are more close to the actual ratings.

3 Feature-level Sentiment Analysis

Internet is a vast resource of users' reviews in the form of free text in review forums, e-commerce sites and discussion groups. It is known that the reviews highly influence customers' purchasing decisions. As reviews of a item can be exploited to obtain its multi-criteria ratings (ratings at the level of individual features), we propose to explore the significance of item's reviews at feature level for recommendation. We employ text mining methods to retrieve relevant information from the reviews.

Though recently some researchers considered analysing items' reviews for recommendation, they are principally limited to deriving virtual, one-dimensional overall ratings from reviews via the sentimental classification methods. However, reviewers' multi-criteria opinions (ratings) on items' features have been rarely considered and analysed. In other words, very few attention have been paid to infer reviewers' preferences on items' features from their written reviews [10]. Therefore, we emphasize feature-level review analysis. Feature level sentiment analysis from customers' reviews includes following three steps.

3.1 Identifying Features of the Items

If a feature f or its synonym appears in a review, it is called explicit feature. For identifying the explicit features, we do parts of speech tagging (POS-TAGGING) [11] of reviews and retrieve the words tagged as nouns to make a list of potential features by using frequent nouns and noun phrases after pruning to remove duplicates. We constrain the features only to be nouns.

3.2 Identify Opinions regarding Item Features

Most of existing works depended on the co-occurrence of item's features and opinion bearing words for this purpose. However, these methods cannot identify opinions that are not too close to the feature. It is shown in few examples below.

– After parsing the sentence "it captures beautiful pictures and was handy to carry and not at all difficult to learn how to use", beautiful is identified with dependency relation AMOD with pictures. It indicates that "beautiful" is an adjectival modifier of the noun word "pictures".
– In another example, "the photos are great", great has NSUBJ relation with photos indicating that "photos" is the noun subject of "great".
– Verbs, e.g., love, like, hate, can also be used to express the sentimental opinion regarding the features. As in "I love its camera but I hate its display", "hate" has DOBJ relation with "display" and "love" has DOBJ relation with "camera". It indicates that "camera" and "display" are verb objects of "love" and "hate", respectively.

We consider all words which have such relations with the items' feature words as opinions. We use a syntactic dependency parser of Stanford core natural language processing toolkit[1] as it returns the syntactic dependency relations between words in a sentence. For example, in Fig. 1 $a$mod(performance-10, super-9) implies that the word "super" at position 9 in the sentence is adjectival modifier of the 10^{th} word "performance" in the sentence.

```
root(ROOT-0, price-4)
nsubj(price-4, it-1)
cop(price-4, is-2)
amod(price-4, worth-3)
amod(quality-7, excellent-6)
prep_with(price-4, quality-7)
amod(performance-10, super-9)
conj_and(quality-7, performance-10)
```

Fig. 1. Syntactic dependency relations among tokens in a sentence.

3.3 Determine Polarity of the Opinions

We identify opinion words and phrases. We have the polarity score of the adjectives, adverbs and verbs with their sentimental score using SentiWordNet dictionary[2]. Negation words and phrases are used to revise the opinion scores and handling BUT-Clauses.

Feature rating of a item is computed as shown in equation 1.

$$feature\ rating = \frac{\sum Polarity\ score\ of\ adjective\ associated\ with\ the\ feature}{Total\ number\ of\ times\ feature\ is\ mentioned\ in\ reviews} \tag{1}$$

[1] http://stanfordnlp.github.io/CoreNLP/
[2] http://sentiwordnet.isti.cnr.it/

It is explained with the help of an example. Consider following two reviews of a item.

- The camera is great. The display is clear. Processing is fast.
- Camera quality is excellent but gets heated up easily.

For this item, rating of the camera is computed as shown in equation 2.

$$rating(camera) = \frac{polarity\_score(great) + polarity\_score(excellent)}{2} \tag{2}$$

Polarity_score is sentimental score of verb, adjective and adverb from SentiWordNet dictionary as shown in figure 2.

```
great  0.08973680845374968
good  0.3883563601675836
better  0.5480991330178288
chinese  0.0
such  -0.041666666666666664
common  -0.021350496905303964
low  -0.07680417032349182
big  0.08006682443206818
worth  0.15306122448979592
excellent  1.0
good  0.3883563601675836
big  0.08006682443206818
good  0.3883563601675836
easily  0.1715328467153285
less  -0.06802721088435375
possible  0.21532846715328466
okay-type  0.0
much  0.1157468243539203
friendly  0.1398071625344353
```

Fig. 2. Polarity score from SentiWordNet dictionary

4 Proposed Recommendation Model

4.1 New-user/old-item Cold Start problem

This scenario occurs in recommending infrequent items where user's history with such items is insufficient. Thus, knowing the user's preference becomes a challenge. Such a user is considered as a new user. For recommending items with missing information of a target user, we assume that the items which are popular in general and are highly rated by other users shall also be liked by the target user. Our methodology for recommendation in this scenario is as follows.

We exploit reviews of the old items and obtain ratings of every feature of all the old items using equation 1. Though we do not have purchasing history of user, we propose to incorporate user's preferences. Therefore, we design an interface to explicitly ask for user's choices and importance index of the features as shown in Figure 3. It ensures that the recommended items have user-desired features.

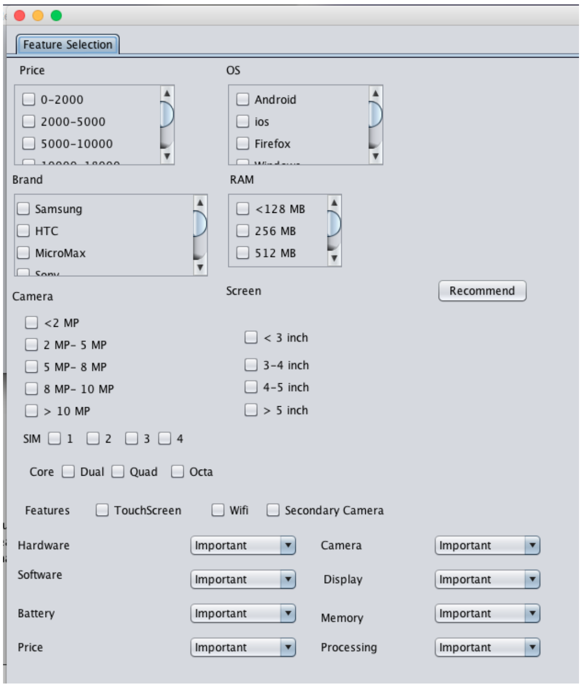

Fig. 3. User interface.

When an user inputs the preference form, items from the catalogue are selected whose features match with the criteria of user's choices. For every selected item overall rating is computed using equation 3.

$$Overall\_rating = \frac{\sum_{f=1}^{8}(rating[f] \times importance[f])}{scaling\_factor} \tag{3}$$

Here, $rating[f]$ is rating of the feature of the item we got by its feature level sentiment analysis and $importance[f]$ is input from the user that how much important the feature f is for the user; its permissible values are 0.5 (not important), 1.0 (important), and 2.0 (very important). Thus, if a feature is not important for the user, its influence on overall rating of the item will be less and rating of a very important feature will greatly influence the overall rating of the item. The $scaling\_factor$ ensures that the overall rating remains in range [0,1]; it is computed as shown in equation 4.

$$scaling\_factor = \frac{1}{num\_features \times max\_value(importance[f])} \tag{4}$$

Here, $num\_features$ is the number of features and $max\_value(importance[f])$ is the maximum value of the importance index a feature can have. As mentioned above, it is 2.0. Thus, the scaling factor = 1/16 as the number of considered features in this work is 8 (refer Fig. 3).

Thus, the overall rating computed on the basis of user-desired features and preferences (multi-criteria ratings) is basis of the recommendation of items. An example of a multi-criteria ratings of a item is shown in figure 4.

```
micromax-q55-bling-limited-iifa-edition
Battery 0.3883563601675836
Hardware 0.12113649712911453
Price 0.0
Camera 0.0
Display 0.09090909090909091
Memory 0.0
Processing 0.0
Software 0.0
```

Fig. 4. Multi-criteria rating of an item.

Once the overall rating of the selected items according to the user's choice is computed, we recommend top rated items to the user as these items are popular among other users and are also highly rated on the features considered important for recommendation by the user.

4.2 New-user/new-item Cold Start Problem

As new items are regularly introduced in the catalogue, they should also be considered for recommendation by the recommender system. However, in absence of any information about the item's popularity, the basic approach is to predict the popularity of the item.

Given a new item i_0, the similarity based recommendation predicts its liking on the basis of ratings of similar items in the past. The more similar i_0 is to the popular old items, the more likely that it itself will be a popular item. Therefore, its recommendation score should be high. The expected rating of the new item i_0 can be computed as the average of the ratings of the old items y weighted according to the similarity of the (respective) items with i_0 as shown in equation 5.

$$Predicted\_rating(i_0) = \frac{\sum_{y=old\_items}(overall\_rating(y) \times sim(y, i_0))}{\sum_{y=old\_items} sim(y, i_0)} \tag{5}$$

Here, $overall\_rating(y)$ is rating of an old item y and $sim(y, i_0)$ is similarity between the new item i_0 and y. The similarity score is computed using the Euclidean distance. This approach makes sure that ratings of the old items that are more similar to i_0 contribute more to its final rating.

However, a limitation of the above approach is that it fails to acknowledge the fact that similarity between the items can exist because of one subset of features whereas the rating is highly influenced by another subset of features and these subsets may be disjoint or may have very less overlapping.

4.3 Our Proposed Formula

Acknowledging the limitation of the above approach, we propose a new formula which ensures that instead of the overall rating of the old items only those features which are similar to the new item i_0 influence its predicted rating as shown in equation 6.

$$Predicted\_rating(i_0) = \frac{\sum_{y=old\_item} \sum_{f=features}(rating(y_f) \times sim(y_f, i_{0_f}))}{\sum_{y=old\_item} \sum_{f=features} sim(y_f, i_{0_f})} \tag{6}$$

Here, y_f denotes features of the old items y and i_{0_f} denotes feature value of the new item i_0. As the formula employ multi-criteria ratings, it is necessary to normalize the feature values in a common range. We normalize the feature values as shown below in equation 7.

$$\frac{Feature\_value - min\_value}{max\_value - min\_value} \tag{7}$$

Here, min_value and max_value are the minimum and maximum values, respectively, of the features over all the items.

An example: Let Table 1 shows the normalized feature values for two old items item1 and item2 and Table 2 provides individual feature ratings of these items. Suppose we want to predict the rating of a new item i_0 having normalized feature values as shown in Table 3, the computations are as follows.

Table 1. Normalized feature values of the old items

Item	Camera	Display	Memory	Price
item1	0.55	0.84	0.30	0.24
item2	0.80	0.65	0.34	0.72

Computations for the similarity based recommendation

Table 2. Feature ratings of the old items

Item	Camera	Display	Memory	Price	Overall
item1	0.33	0.63	-0.55	-0.42	0.29
item2	-0.75	-0.56	0.86	0.64	0.32

Table 3. Feature values of the new item i_0

Item	Camera	Memory	Display	Price
i_0	0.25	0.48	0.36	0.75

$sim(item1, i_0) = 1 - \frac{\sqrt{\sum (0.55-0.25)^2+(0.84-0.36)^2+(0.30-0.48)^2+(0.24-0.75)^2}}{4} = 0.22$

$sim(item2, i_0) = 1 - \frac{\sqrt{\sum (0.80-0.25)^2+(0.65-0.36)^2+(0.34-0.48)^2+(0.72-0.75)^2}}{4} = 0.37$

Therefore, the predicted rating of the new item $i_0 = \frac{0.29*0.22+0.32*0.37}{0.22+0.37} = 0.308$

Computational steps for the proposed formula

$x = sim(0.55, 0.25)*0.33+sim(0.84, 0.36)*0.63+sim(0.30, 0.48)*(-0.55)+sim(0.24, 0.75)*(-0.42)$

$y = sim(0.80, 0.25)*(-0.75) + sim(0.65, 0.36)*(-0.56) + sim(0.34, 0.48)*0.86 + sim(0.72, 0.75)*0.64$

$z = sim(0.55, 0.25) + sim(0.84, 0.36) + sim(0.30, 0.48) + sim(0.24, 0.75) + sim(0.80, 0.25) + sim(0.65, 0.36) + sim(0.34, 0.48) + sim(0.72, 0.75)$

Therefore, the predicted rating of the new item $i_0 = \frac{x+y}{z} = 0.423$. Here, $sim(a, b) = 1 - absolute\_difference(a - b)$

5 Experimental Results

We use mobile handset dataset for an empirical study of the traditional similarity based recommendation formula and the proposed formula. We scrap e-commerce website Flipkart to collect feature information and customer reviews of 232 mobiles and compute ratings for every feature of each item by doing feature level sentiment analysis of the reviews. We consider 30 out of these 232 mobile handsets as new items to compare actual ratings with the predicted ratings by similarity based recommendation formula and the proposed formula. Remaining items are considered as the old items.

For our experiment, we present the user with an input form where she/he inputs her/his choice of features and their importance. Using this input, we select items whose features matches with the features selected by the user and compute their overall rating using equation 1 and recommend top 5 highest rated items. Further, we compare the actual ratings of 30 mobile handsets with the predicted ratings by similarity recommendation formula and the proposed formula using root mean square error (RMSE) as shown in equation 8.

$$RMSE = \sqrt{\frac{\sum_{y=new\_item}(actual\_rating(y) - predicted\_rating(y))^2}{n}} \tag{8}$$

Table 4. Comparison Results

Comparison metric	Similarity based	Proposed Formula
RMSE	0.03448566	0.02911068

The RMSE values for the similarity based recommendation and the proposed formula are 0.03448566 and 0.02911068, respectively (refer Table 4). Moreover, the ratings of all the 30 items computed by our proposed formula are better and more closer to the actual ratings than the ratings computed by the similarity based recommendation formula. As predicted ratings by the proposed formula are closer to the actual ratings, the RMSE value of the proposed formula is better. Therefore, we infer that it is better to consider feature level ratings instead of overall rating of the old items to predict rating of a new item.

6 Conclusion

An accurate and efficient recommender system is essential for a successful e-commerce business; it helps to attract customers by providing them with a list of items that they may be interested in. Moreover, it also helps users to make buying choices when they don't want to scroll every item in the catalogue. Traditional methods, i.e., collaborative filtering, content based approach, hybrid approaches, are successful only when the items are bought frequently as it provides sufficient history of user-item rating to analyse users' behaviour, users' preferences and finding relation among the items. However, these methods are not useful for infrequent items as the user-item matrix is very sparse. Thus, it is considered as new-user cold start problem. As new items are regularly introduced in the catalogue, the recommender system should be able to recommend them also. It is seen as new-item cold start problem. Therefore, this paper propose a different approach called interactive recommender system which takes user's preferences as input and recommend both the old and the new items which are rated high on the preferred features instead of overall ratings of the items. In case of old items, customers' textual reviews are analysed for feature-level sentiment analysis to obtain features' ratings of the items. However, the new items are recommended as per the predicted rating computed by their feature level similarity to the old items; new items with popular and highly rated features are considered popular and recommended.

The proposed approach is tested on the mobile handset dataset. The empirical results show that the proposed approach predicts ratings for new items which are closer to the actual ratings. Moreover, for recommending old items also, it incorporates users' preferences.

7 Future Work

In future work, we intend to include users' skill level as a parameter in evaluating the reviews as reviews by frequent users of items tend to be more reliable than the reviews of occasional users. Therefore, evaluations of the experienced users should attract more weight in comparison to the general reviews by users. Further, for the computation of predicted rating in the similarity based recommendation, it is better to consider k closest similar items instead of taking all the items.

References

1. Billsus, D., Pazzani, M.J.: Learning Collaborative Information Filters. In: Proc. 5th Intl. Conf. on Machine Learning, ICML 1998, Vol. 98, pp. 46–54 (1998)
2. Basu, C., Hirsh, H., Cohen, W.: Recommendation as Classification: Using Social and Content-based Information in Recommendation. In: Proc. 15th National / 10th Conf. on Artificial Intelligence / Innovative Applications of Artificial Intelligence, AAAI/IAAI 1998, pp. 714–720 (1998)
3. Jung, K.-Y., Park, D.-H., Lee, J.-H.: Hybrid Collaborative Filtering and Content-based Filtering for Improved Recommender System. In: 4th Intl. Conf. on Computational Science, ICCS 2004, LNCS, Vol. 3036, pp. 295–302 (2004)
4. Bykau, S., Koutrika, G., Velegrakis, Y.: Coping with the Persistent Cold-start Problem. In: Personalized Access, Profile Management, and Context Awareness in Database, PersDB 2013, (2013)
5. Melville, P., Mooney, R.J., Nagarajan, R.: Content-boosted Collaborative Filtering for Improved Recommendations. In: Porc. 18th National Conference on Artificial Intelligence, pp. 187–192 (2002)
6. Schein, A.I., Popescul, A., Ungar, L.H., Pennock, D.M.: Methods and Metrics for Cold-start Recommendations. In: Proc. 25th Annual International ACM SIGIR Conf. on Research and development in Information Retrieval, ACM, pp. 253–260 (2002)
7. Aciar, S., Zhang, D., Simoff, S., Debenham, J.: Informed Recommender: Basing Recommendations on Consumer Product Reviews. Intelligent Systems, IEEE, 22(3), 39–47 (2007)
8. Guy, I., Zwerdling, N., Carmel, D., Ronen, I., Uziel, E., Yogev, S., Ofek-Koifman, S.: Personalized Recommendation of Social Software Items based on Social Relations. In: Proc. 3rd ACM Conf. on Recommender Systems, ACM, pp. 53–60 (2009)
9. Iaquinta, L., Semeraro, G.: Lightweight Approach to the Cold-start Problem in the Video Lecture Recommendation. In: Proc. ECML/PKDD Discovery Challenge Workshop, CEUR, Vol. 770, pp. 83–94 (2011)
10. Popescu A.-M., Etzioni, O.: Extracting Product Features and Opinions from Reviews. Natural Language Processing and Text Mining, Springer, pp. 9–28 (2007)
11. MacKinlay, A., Baldwin, T.: POS Tagging with a more Informative Tagset. In: Proc. The Australasian Language Technology Workshop, pp. 40–48 (2005)