Future Computing and Informatics Journal

Volume 3
Issue 2 (2018) *Vol 3,Issue 2,2018*

Article 2

2018

A linguistic approach for opinionated documents summary

Mahmoud samy
Faculty of Computers and Information Technology, Future University in Egypt, Cairo, Egypt,
msamy@fue.edu.eg

Available online at www.sciencedirect.com

ScienceDirect

Future Computing and Informatics Journal 3 (2018) 152–158
http://www.journals.elsevier.com/future-computing-and-informatics-journal/

A linguistic approach for opinionated documents summary

Mahmoud Othman [a], Hesham Hassan [b], Ramadan Moawad [a], Amira M. Idrees [c,*]

[a] Faculty of Computers and Information Technology, Future University in Egypt, Cairo, Egypt
[b] Faculty of Computers and Information Cairo University, Cairo, Egypt
[c] Faculty of Computers and Information Fayoum University, Fayoum, Egypt

Received 25 May 2017; revised 22 September 2017; accepted 17 October 2017
Available online 11 November 2017

Abstract

Now, the web pages contain opinions on almost anything, at review sites, forums, discussion groups, and blogs which called user generated content. They contain valuable information for different users such as persons or organizations, the processes of collecting, analyzing and classifying them to positive or negative opinions in addition to summarizing the opinions are considered a very important research issue. Summarizing opinions helps users to explore the opinion of others about the key aspects of a topic or an entity. The proposed opinion summarization system receives a document that contains sentences expressing opinions about an entity and generates a summary considering the important aspects, their relations, their sentiments and the textual evidences, as expressed in the reviews. In this paper we present a linguistic approach to summarize the opinionated documents across different domains, our evaluation based on a dataset of hotels, cars and various products reviews. The reviews collected from Tripadvisor, Amazon and Edmunds, each review document consist of a set of unordered, redundant reviews sentence, there are approximately 100 sentences per review document. The summary depends on the type of the opinion which is direct, comparative, or superlative. Each type is assigned to a specialist who is responsible for the summary.

Keywords: Natural language processing; Opinion mining; Opinion discovery; Sentimental analysis

1. Introduction

Several business intelligence applications across several domains are considering opinion mining as key enabling technology for their business [4,10,11]. They are trying to build a system to collect and examine opinions about a product or a topic which have been posted in blogs, comments, reviews or tweets. Opinion mining or sentiment analysis includes different levels; the first level is Document-level which identifies if the document (e.g. product reviews, blogs, and forum posts) expresses opinions and the opinions' polarity if they are positive, negative, or neutral; the second level is Sentence-level which identifies the polarity of each sentence in the document and whether the opinion is positive, negative, or neutral, and the third level is Attribute-level which extract the object attributes (e.g. image quality, zoom size) that is a subject of an opinion and the opinion orientations [1,2,9,12].

Furthermore, summarizing opinions helps users to explore the opinion of others about the key aspects of a topic or an entity. The proposed opinion summarization system receives a document that contains sentences expressing opinions about an entity and generates a summary considering the important aspects, their relations, their sentiments and the textual evidences, as expressed in the reviews.

The applied summarization methods in the proposed system are: Extractive Summary and Abstractive Summary. Extractive Summary is a summary that depends on selecting a part of the text which represents the main aim of the whole text. On the other hand, abstractive summary follows another approach, it performs analytical approach upon the text, and generates new sentences that represent the information in the

* Corresponding author.
E-mail address: ami04@fayoum.edu.eg (A.M. Idrees).
Peer review under responsibility of Faculty of Computers and Information Technology, Future University in Egypt.

https://doi.org/10.1016/j.fcij.2017.10.004

text [5,7]. As abstractive summary requires more complex approaches, therefore, targeting extractive summary has higher interest [6]. In this paper, a summary is produced based on predicting the sentiment of the sentences and generate a summary for the document.

The remaining of the paper discusses the related work, the proposed approach, and the experiment. Finally, a conclusion is provided with presenting the future research points.

2. Related work

Polarities of the opinion are critical to be determined for the corpus' sentences. As these sentences may include more information about the target such as the item's satisfaction score. Another focus that most of the proposed approaches of summarization task have a defined structure which is labeled by the polarity and the field. The summarization techniques can be supportive in determining an opinion summary and the polarity of the document. The summary generation depends totally on the analysis step [5].

In a research conducted by Kim and Zhai [5], a framework has proposed which targets to optimize the extraction for contrastive opinions' sentences. The research has presented two extraction methods based on a clustering technique. The presented method apply a similarity technique to find related sentences which are considered as pairs, then provide the required summary based on this relation. Another research in [3] proposed a summarization framework (Opinosis) which was able to generate an abstractive summary of a document that includes redundant opinions.

3. Formal description of the proposed approach

In this section we will describe the formal description of the proposed approach in details as follows:

3.1. Basic definitions

If we consider d is a document, then we will define Document (d) as a set of sentences follows [9]:

$$\text{Document}\,(d_i) = \{s_1, s_2, s_3 \ldots s_n\}$$

Then, we will define Document (D) as a set of all documents d. The set of all documents Document (D) consists of all sentences in the documents as follows

$$\text{Document}\,(D) = \text{Document}\,(d_1) \cup \text{Document}\,(d_2) \cup \text{Document}\,(d_3) \ldots \cup \text{Document}\,(d_i)$$

The sentence consists of a set of tokens; then we will define sentence (S) as a set of all tokens w.

$$\text{Sentence}\,(s_n) = \{w_1, w_2, w_3, \ldots w_r\}$$

The tagged token (w) contains four classes as follow:

- **NSTagged (d_i)** is the set of all tokens that are of type **non-Opinionated.**
- **CSTagged (d_i)** is the set of all tokens that are of type **Comparative Opinionated.**
- **SSTagged (d_i)** is the set of all tokens that are of type **Superlative Opinionated.**
- **OSTagged (d_i)** is the set of all tokens that are of type **Opinionated.**

3.2. Formal description for opinion classifier phase

The goal of opinion classifier phase is to extract the opinioned statements from each document (d) and determine the type of each statement (direct opinion, comparative opinion or superlative opinion or non-opinionated) and assign it to a class (GO, CO or SO). To reach the goal of this phase, we follow the approach provided in [9]. Fig. 1 describes the general architecture of opinion classifier phase that is demonstrated in [9].

The input of the phase is Document (D) then the output will be set of classes (C) which composed from the classes:

- **1st Class:** Set of general opinions class (GO)

GO (D) = {ops_1, ops_2, ops_3,……, ops_n}, This definition means that the GO (D) contains all sentences (S) which classified as direct opinions where:

ops_i is the sentences that contains OSTagged (d_i)

- **2nd Class:** Set of comparative opinions class (CO)

CO (D) = {$cops_1$, $cops_2$, $cops_3$,……, $cops_n$}, This definition means that the CO (D) contains all sentences (S) that classified as comparative opinions where:

$cops_i$ is the sentences that contains CSTagged (di)

- **3rd Class:** Set of Superlative Opinions Class (SO)

SO (D) = {$sops_1$, $sops_2$, $sops_3$, ……. , $sops_n$}, This definition means that the SO (D) contains all sentences (S) that classified as superlative opinions where:

$sops_i$ is the sentence that contains CSTagged (d_i).

4. Proposed system framework

The proposed system aims to generate an efficient opinion summary from a large corpus of opinions represented in a set of sentences. The proposed framework is able to generate an accurate concise summary by applying an aggregation methods an determine the semantic score. Fig. 2 presents the main components of the proposed system while the main

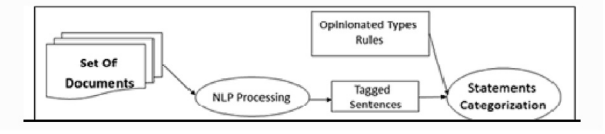


Fig. 1. General architecture of opinion classifier process [9].

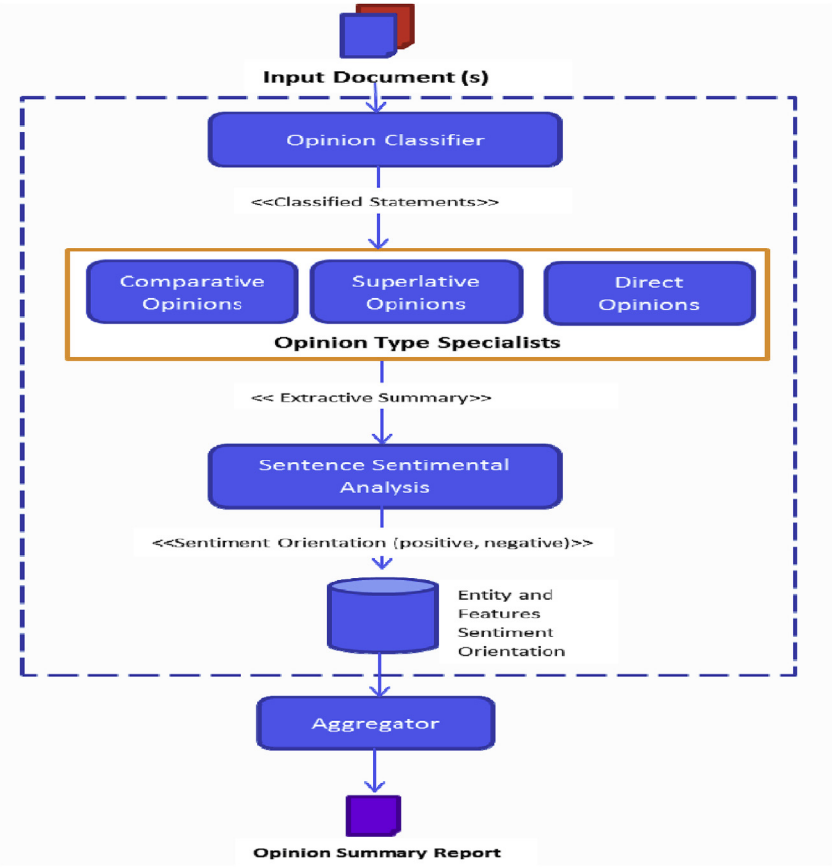


Fig. 2. General architecture of the proposed system.

approach of the system can be summarized to the following steps:

1. Processing the input document(s) targeting to tag each word in the document.
2. Extracting the entities, its attributes (features) and opinion words from each sentence.
3. Defining the opinion type of each sentence or a group of sentences according to the sentiment value the POS tags.
4. Generating the extractive summary of each statement. This step is performed using three types of specialists which are comparative opinions, superlative opinions and direct opinion specialists.
5. Generating the final opinion summary based on the outputs of these specialists. This output is sent as an input to the sentimental analysis process which is responsible for generating the sentiment orientation of each opinion word related to its entity and its attributes then store them in the data repository for further use by the aggregator to generate the final opinion summary report.

4.1. Opinion classifier process

The purpose of the opinion classifier process is to discover the opinions in a document or a set of documents which may be different types of opinionated statements, including the opinionated, comparative, and superlative opinions. The main approach of the classifier is to process the input document(s) targeting to tag each word in the document. Then in each sentence, the opinionated words are highlighted with providing a weight the highlighted words. Finally, the system defines the opinionated type of each sentence or a group of sentences according to the sentiment value highlighted opinionated words. Fig. 1 presents the main components of the classifier process [9].

4.2. Opinion type specialists

In this component, there are three types of specialists; each specialist is responsible for one of the opinion types, (Direct, Comparative, Superlative), in order to perform the extractive summary of the opinion. The association between the entities (or features) and adjectives (or adverbs) included in the sentence are considered according to the specialist, Table 1 illustrates the opinion type and its specialist.

1) Comparative Opinion Specialist

The comparative specialist is the first type of specialists which deals with the relations that express a total ordering of some entities with regard to their shared features [8]. It is responsible for generating the extractive summary of any

comparative sentence by analyzing the part of speech tags and identifying its sub-constituents into each sentence such as noun chunks. As a result, it identifies the associations between the features or entities and their related information. The following presents the algorithm describing the main steps performed in this component.

```
Get Comparative Sentence X
Get Comparative Sentence Xpos // Part of speech
Define EntityFeatureList1← Empty List
Define EntityFeatureList2← Empty List
Define Relation← Null
Tokens [ ] ← Split(x," ")
POS [ ] ← Split(Xpos," ")
i← 0
j← 0
while(i<POS.size()){
    if( xpos [i] = "NN"  and j=1)  // from POS
        EntityFeatureList1.add(x[i])
   Else if (xpos [i] = "NN"  and j=0)
        EntityFeatureList2.add(x[i])
    Else if (xpos [i] =" JJR"   or xpos [i] =" RBR")
        Relation= x[i]
         j←1
    i←i+1
}
Return   <EntityFeatureList1,   Relation,   Entity
FeatureList2>
```

For example, the sentence, "Mobile X's battery life is longer than that of Mobile Y", orders "Mobile X" and "Mobile Y" based on their shared feature "battery life" and summarize the opinion as follows:

< (Entity Features List1), Relation, (Entity Features List2)>

The output of this specialist is the input to the "sentence sentimental analysis" module. Table 2 presents some examples of the input and corresponding output of this component.

2) Superlative Opinion Specialist

This module is responsible for generating the extractive summary of the superlative sentences by analyzing the part of speech tags and identifying the sub-constituents of each sentence such as noun chunks. As a result, it identifies the associations between the features or entities and their related information. The following presents the algorithm which represents the main steps performed in this component.

Table 1
Opinion types and specialists.

Sentimental Category	Specialist	POS
Non-opinionated statement	Discarded	–
Comparative opinionated statement	Comparative specialist	JJR, RBR
Superlative opinionated statement	Superlative specialist	JJS, RBS
Opinionated statement	Direct specialist	JJ

Table 2
Examples of *Comparative Opinion Specialist* input and corresponding output.

Example	Output
Mobile X is better than Mobile Y	<(Mobile X), NULL, Better, (Mobile Y),NULL>
Mobile X and Y are better than Z	<(Mobile X, Mobile Y), NULL, Better, (Mobile Z),NULL>
Mobile X voice quality is better than Mobile Y voice quality	<(Mobile X, Voice Quality), Better, (Mobile Y), (Voice Quality)>
Mobile X voice quality is better than Mobile Y	<(Mobile X, Voice Quality), Better, (Mobile Y)>

```
Get Superlative Sentence X
Get Superlative Sentence Xpos // Part of speech
Define EntityFeatureList ← Empty List
Define RelationList← Empty List
Tokens [ ] ← Split(X," ")
POS [ ] ← Split(Xpos," ")
i← 0
while(i<POS.size()){
    if( xpos [i] = "NN")  // from POS
        EntityFeatureList.add(x[i])
    Else if (xpos [i] =" JJS"   or xpos [i] =" RBS")
        RelationList.add( x[i])

    i←i+1
}
Return < EntityFeatureList, RelationList>
```

In the proposed system, the superlative specialist deals with the relations of the type greatest or least of some entities with regard to their shared features [8]. For example, the sentence, "Mobile X's battery life is the longest", orders "Mobile X" based on its feature "battery life" and summarize the opinion as follow:

< (Entity, Features List), (Relations List)>

The output of this specialist is send to sentence sentimental analysis module. Table 3 presents some examples of the input and corresponding output of this component.

Table 3
Examples of *Superlative Opinion Specialist* input and corresponding output.

Example	Output
Mobile X is the best	<(Mobile X), (best) >
Mobile X is the worst voice quality and worst worst camera.	<(Mobile X, Voice quality,camera), (worst, worst)>

3) Direct Opinions Specialist

The third type of specialists is the direct opinion which deals with the direct opinions, the following presents the algorithm which represents the main steps performed in this component.

```
Get opinion Sentence X
Get opinion Sentence Xpos // Part of speech
Define EntityFeatureList ← Empty List
Define Relation← NULL
Tokens [ ] ← Split(X," ")
POS [ ] ← Split(Xpos," ")
i← 0
while(i<POS.size()){
    if( xpos [i] = "NN")  // from POS
        EntityFeatureList.add(x[i])
    Else if (xpos [i] =" JJ")
        Relation= x[i]
    i←i+1
}
Return < EntityFeatureList, Relation>
```

For example, the sentence, "Mobile X's battery life id good", orders "Mobile X" based on its feature "battery life" and summarize the opinion as follow:

< (Entity, Features List), Relation>

The output of this specialist is send to sentence sentimental analysis module. Table 4 presents some examples of the input and corresponding output of this component.

4.3. Entity and feature sentimental orientation module

The output of the specialists is an input for this module to get the sentiment orientation of the extracted relations, then store the result in a data repository for further summary. We use SentiWordNet to get the score of the relation according to the algorithm presented as follows.

```
Get relation
Calculate score    //sentiwordnet
 if(score>=0.75)
      sentiment orientation = "strong_positive"
else if(score > 0.25 && score<=0.5)
      sentiment orientation = "positive"
else if(score > 0 && score<=0.25)
      sentiment orientation = "weak_positive"
else if(score < 0 && score>=-0.25)
      sentiment orientation = "weak_negative"
else if(score < -0.25 && score>=-0.5)
      sentiment orientation = "negative"
else if(score<=-0.75)
      sentiment orientation = "strong_negative"
```

Table 4
Examples of *Direct Opinions Specialist* input and corresponding output.

Example	Output
Mobile X is good	<(Mobile X), good >
Mobile X voice quality is very bad.	<(Mobile X, Voice quality), bad>

4.4. Aggregator module

The purpose of the aggregator module is to calculate the total positive and negative opinion for a certain entity or feature, the calculations extracted from the *Feature Sentimental Orientation* repository which is filled from the previous step *Entity and Feature Sentimental Orientation Module* (see Fig. 1).

5. Experimental examples

The following are examples presenting the proposed system's output.

Example 1. The first example for the comparative opinionated statement. System Input: Iphone is better than Blackberry.

The output is presented in Fig. 3.

Example 2. The second one is for the superlative opinionated statement. System Input: Yesterday, I searched the web by different search engines I find that google is the best search engine.

The output is presented in Fig. 4.

Example 3. The third one for the direct opinion. System Input: I Bought a Lancer Car, It is amazing Car

The output is presented in Fig. 5.

6. Experiment results' evaluation

The applied experiment is performed using the dataset in [3]. The dataset consists of reviews of hotels, cars and various products such as mobiles and tablets classified as comparative, superlative or opinionated which count is presented in Table 5. According to Kavita and et al. [3], the reviews are collected from Tripadvisor, Amazon and Edmunds, Each document includes a set of sentences which are considered the set of

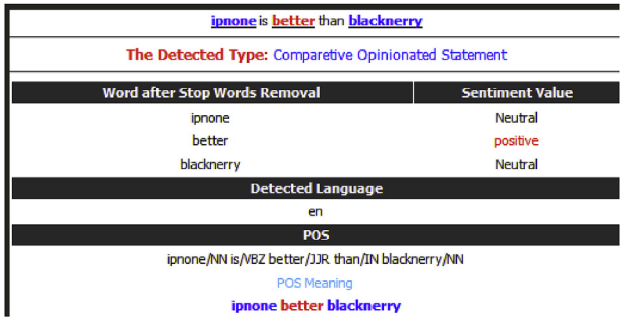


Fig. 3. Output of the proposed system for first example.


yesterday I searched the web by different search engines i find that google is the best search engine

The Detected Type: Superlative Opinionated Statement

Word after Stop Words Removal	Sentiment Value
yesterday	Neutral
searched	Neutral
web	Neutral
different	positive
search	Neutral
engines	Neutral
google	Neutral
best	positive
search	Neutral
engine	Neutral

Detected Language

en

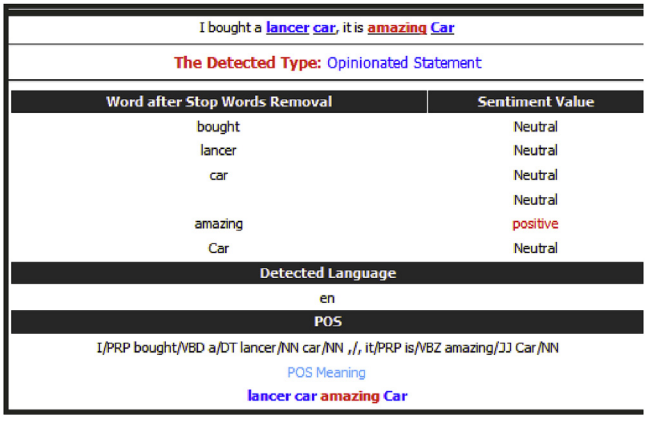
POS

yesterday/NN I/PRP searched/VBD the/DT web/NN by/IN different/JJ search/NN engines/NNS i/FW find/VBP that/DT google/NN is/VBZ the/DT best/JJS search/NN engine/NN

POS Meaning

google best search engine

Fig. 4. Output of the proposed system for second example.


I bought a lancer car, it is amazing Car

The Detected Type: Opinionated Statement

Word after Stop Words Removal	Sentiment Value
bought	Neutral
lancer	Neutral
car	Neutral
	Neutral
amazing	positive
Car	Neutral

Detected Language

en

POS

I/PRP bought/VBD a/DT lancer/NN car/NN ,/, it/PRP is/VBZ amazing/JJ Car/NN

POS Meaning

lancer car amazing Car

Fig. 5. Output of the proposed system for third example.

reviews. The document includes about 100 sentences while there corpus includes approximately 360 documents.

A research presented in [3] aimed to evaluate the Opinosis' output represented in the generated summary by comparing this output with manual summary generated by an expert. The research [3] applied the ROUGE technique [7] for the required evaluation. The same methodology has been applied in the current research to evaluate the proposed summarization method. Fig. 6 presents the comparison between the proposed approach and Opinosis. Evaluation measures presented in Table 6 clarify the advance of the proposed approach over Opinosis.

Table 5
Number of statements in the dataset.

Category	No. of statements
Opinionated	1500
Comparative	1000
Superlative	1000
Total	**3500**

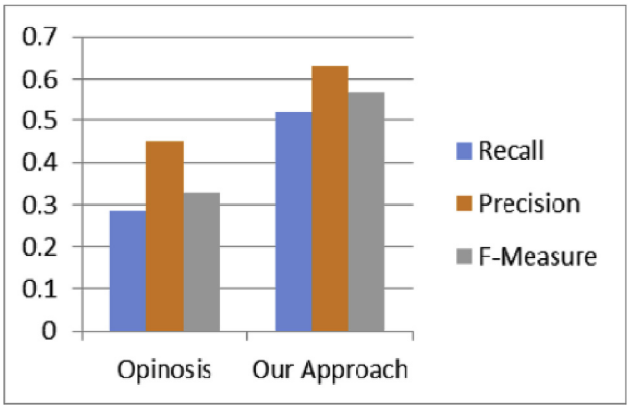


Fig. 6. Comparative analysis between the proposed approach and previous studies.

Table 6
Comparative measures between the proposed approach and previous studies.

Evaluation measures	Opinosis [3,7]	Proposed approach
Recall	0.2831	0.52
Precision	0.4482	0.63
F-Measure	0.3271	0.5697

7. Conclusion and future research

Several opinion Mining and sentimental analysis systems have been developed to analyze comments, tweets related to services and products. Also there are more than one approach such as machine learning approaches and lexicon based approaches. In this paper, we describe a linguistic approach for opinionated document summary. The objectives of the proposed approach are to discover the opinion of the sentences in a document or set of documents, determine the types of opinionated statements including the Opinionated, comparative, superlative, and non- Opinionated, determine the sentimental orientation for the entities, then generate the opinion summary for the document(s).

An experiment has been applied for evaluation and comparison with recent research has been presented which proved the novelty and advances of the proposed approach.

Our future work is the extension of this work to include multi-language opinions and construct an opinion search engine in different domains. We also aim to develop a generic configurable approach for opinion mining.

Funding information

No funding organization.

Author's contributions

Authors of this research has contributed equally in all parts.

Ethics

No Ethical Issues.

Acknowledgement

No acknowledgment.

References

[1] Popesu, Etzioni O. Extracting product features and opinions from reviews. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. EMNLP; 2005.

[2] Cruz F, Troyano J, Ortega F, Vallejom C. In: Long autonomy or long delay? The importance of domain in opinion mining, vol. 40. Elsevier : Expert Systems with Applications; 2013.

[3] Kavita G, ChengXiang Z, Jiawei H. Opinosis: a graph based approach to abstractive summarization of highly redundant opinions. In: Proceedings of the 23rd International Conference on Computational Linguistics (COLING '10); 2010.

[4] Fei G, Liu B, Hsu M, Castellanos M, Ghosh R. A dictionary-based approach to identifying aspects implied by adjectives for opinion mining. In: Proceedings of COLING; 2012. p. 309−18.

[5] Kim H, Zhai. Generating comparative summaries of contradictory opinions in text". In: CIKM '09: Proceeding of the 18th ACM conference on information and knowledge management. New York, NY, USA: ACM; 2009. p. 385−94.

[6] Toutanova K, Manning C. Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In: Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpor. EMNLP/VLC; 2000.

[7] Chin-Yew L. Rouge: a pack- age for automatic evaluation of summaries. In: Proceedings of the Workshop on Text Summarization Branches Out (WAS); 2004.

[8] Ganapathibhotla M, Liu B. Mining opinions in comparative sentences. In: Proceedings of the 22nd International Conference on Computational Linguistics, Manchester; 2008.

[9] Othman M, Hassan H, Moawad R, Idrees A. Using NLP approach for opinion types classifier. J Comput 2016;11(5) (Accepted paper).

[10] Sobkowicz P, Kaschesky M, Bouchard G. In: Opinion mining in social media: modeling, simulating, and forecasting political opinions in the web, vol. 29. Elsevier : Government Information Quarterly; 2012.

[11] Dang Y, Zhang Y, Chen H. A lexicon-enhanced method for sentiment classification: an experiment on online product reviews. IEEE Intell Syst 2010;25.

[12] Lei Z, Bing L. Sentiment analysis and opinion mining: introduction and survey book. publisher: Morgan & Claypool; 2011.