International Journal of Electrical and Computer Engineering (IJECE)
Vol. 11, No. 2, April 2021, pp. 1613∼1626
ISSN: 2088-8708, DOI: 10.11591/ijece.v11i2.pp1613-1626

MTVRep: A movie and TV show reputation system based on fine-grained sentiment and semantic analysis

Abdessamad Benlahbib, El Habib Nfaoui
Computer Science Department, LISAC Laboratory, Faculty of Sciences Dhar EL Mehraz (F.S.D.M), Sidi Mohamed Ben Abdellah University, Fez, Morocco

Article Info

Article history:

Received Jul 21, 2020
Revised Sep 8, 2020
Accepted Sep 28, 2020

Keywords:

Decision making
Fine-grained sentiment analysis
Natural language processing
Reputation generation
Text mining

ABSTRACT

Customer reviews are a valuable source of information from which we can extract very useful data about different online shopping experiences. For trendy items (products, movies, TV shows, hotels, services ...), the number of available users and customers' opinions could easily surpass thousands. Therefore, online reputation systems could aid potential customers in making the right decision (buying, renting, booking ...) by automatically mining textual reviews and their ratings. This paper presents MTVRep, a movie and TV show reputation system that incorporates fine-grained opinion mining and semantic analysis to generate and visualize reputation toward movies and TV shows. Differently from previous studies on reputation generation that treat the task of sentiment analysis as a binary classification problem (positive, negative), the proposed system identifies the sentiment strength during the phase of sentiment classification by using fine-grained sentiment analysis to separate movie and TV show reviews into five discrete classes: strongly negative, weakly negative, neutral, weakly positive and strongly positive. Besides, it employs embeddings from language models (ELMo) representations to extract semantic relations between reviews. The contribution of this paper is threefold. First, movie and TV show reviews are separated into five groups based on their sentiment orientation. Second, a custom score is computed for each opinion group. Finally, a numerical reputation value is produced toward the target movie or TV show. The efficacy of the proposed system is illustrated by conducting several experiments on a real-world movie and TV show dataset.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Abdessamad Benlahbib
Computer Science Department, LISAC Laboratory
Faculty of Sciences Dhar EL Mehraz (F.S.D.M), Sidi Mohamed Ben Abdellah University
Fez B.P. 1796 Fes-Atlas, 30003 Morocco
Email: abdessamad.benlahbib@usmba.ac.ma

1. INTRODUCTION

The exponential growth of Web 2.0 has dramatically impacted the evolution of e-commerce platforms [1–4]. On the one hand, some recent statistics show that 72% of customers will not take action until they read reviews, and only 6% of consumers don't trust customer reviews at all, on the other hand, the number of user-generated reviews attached to an online entity could easily exceed thousands [5, 6]. Thus, a potential customer doesn't have the time or effort to examine all the reviews manually in order to make a decision toward it [7, 8].

Little research has been conducted in mining customer and user reviews with regard to feature-based summarization and reputation generation for the purpose of supporting customer decision making process in E-commerce (buying, renting, booking ...). Over the last two decades, a few opinion summarizer systems

have been proposed to produce a summary for product reviews [9], movie reviews [3], hotel reviews [1] and local service reviews [10]. Backing to reputation generation task, to the best of our knowledge, there are very few reputation systems that have been proposed to compute a single reputation value toward different entities based on fusing and mining user and customer reviews expressed in natural language [11–15]. Yan *et al.* [11] applied opinion mining and fusion techniques on product reviews. Benlahbib and Nfaoui [12] used K-Means clustering algorithm on movie reviews. The same authors [13] incorporated semantic and sentiment analysis to generate a single reputation value from user and customer reviews expressed in natural language (English).

An important issue that was neglected in the past research on reputation generation is identifying the sentiment strength during the phase of sentiment classification and opinion fusion. In fact, existing works have only focused on classifying reviews into positive or negative before generating a single reputation value, disregarding the sentiment strength.

In this paper, we propose MTVRep, a movie and TV show reputation system that applies fine-grained opinion mining to separate reviews into five opinion groups: strongly negative, weakly negative, neutral, weakly positive and strongly positive. Then, it computes a custom score for each group based on the acquired statistics of each group, i.e., the number of reviews in each group, the sum of their ratings and the sum of their semantic similarity (ELMo and cosine metric). Finally, a numerical reputation value is produced toward the target movie or TV show using the weighted arithmetic mean.

In this manner, this study addressed the following research question: with the combination of fine-grained opinion mining and semantic analysis, can the proposed reputation system offer better results in terms of reputation generation than the previous reputation systems (consider only semantic relations)?. The remainder of this paper is organized as follows: Related works are provided in Section 2. Section 3 illustrates the work-flow of the reputation system. Section 4 presents all the experimental results and discusses its comparative performance, finally conclusions are drawn in Section 5.

2. LITERATURE REVIEW

This section describes and examines previous research work done in the area of natural language processing (NLP) techniques for decision making in E-commerce and fine-grained sentiment analysis.

2.1. Fine-grained sentiment analysis on the 5-class stanford sentiment treebank (SST-5) dataset

Xu *et al.* [16] proposed Emo2Vec which are word-level representations that encode emotional semantics into fixed-sized, real-valued vectors. Mu *et al.* [17] presented a simple post-processing operation that renders word representations even stronger by eliminating the top principal components of all words. Socher *et al.* [18] introduced recursive neural tensor networks and the stanford sentiment treebank. Wang *et al.* [19] proposed RNN-Capsule, a capsule model based on recurrent neural network (RNN) for sentiment analysis. Yang [20] presented RNFs, a new class of convolution filters based on recurrent neural networks. McCann *et al.* [21] introduced an approach for transferring knowledge from an encoder pretrained on machine translation to a variety of downstream natural language processing (NLP) tasks. Munikar *et al.* [22] used the pretrained BERT [23] model and fine-tuned it for the fine-grained sentiment classification task on the SST-5 dataset. Table 1 summarizes the latest works on fine-grained opinion mining applied to stanford sentiment treebank dataset (SST-5).

Table 1. State-of-the-art results for sentiment analysis on SST-5 fine-grained classification

Method	Authors and Year	Accuracy %
BCN+Suffix BiLSTM-Tied+CoVe	Brahma (2018) [24]	56.2
BERT large	Munikar *et al.* (2019) [22]	55.5
BCN+ELMo	Peters et al. (2018) [25]	54.7
BCN+Char+CoVe	McCann *et al.* (2017) [21]	53.7
CNN-RNF-LSTM	Yang (2018) [20]	53.4
RNN-Capsule	Wang *et al.* (2018) [19]	49.3
SWEM-concat	Shen *et al.* (2018) [26]	46.1
RNTN	Socher *et al.* (2013) [18]	45.7
GRU-RNN-WORD2VEC	Mu *et al.* (2017) [17]	45.02
GloVe+Emo2Vec	Xu *et al.* (2018) [16]	43.6
Emo2Vec	Xu *et al.* (2018) [16]	41.6

2.2. NLP techniques for decision making in E-commerce

It has been well recognized that user reviews attached to an entity (movie, product, etc...) contain valuable information about it. Recently, few approaches have been proposed to help potential customers during decision-making process in E-commerce websites by automatically mining user and customer reviews. The most popular approaches are feature-based summarization and reputation generation.

Feature-based summarization approaches aim to produce a feature-based summary for a target entity as shown in Figure 1. The first feature-based summarizer system was proposed by Hu and Liu [9] in which they applied association rule mining to extract product features, and they used a set of seed adjectives to identify the semantic orientation for opinion words. Zhuang *et al.* [3] built a multi-knowledge based system that aims to generate a feature-based summary for online movie reviews. Blair-Goldensohn *et al.* [10] presented a feature-based summarizer for local service reviews. Kangale *et al.* [27] proposed a feature-based summarize system for product reviews that produces a rating as well as review summary of each product feature as shown in Figure 1.

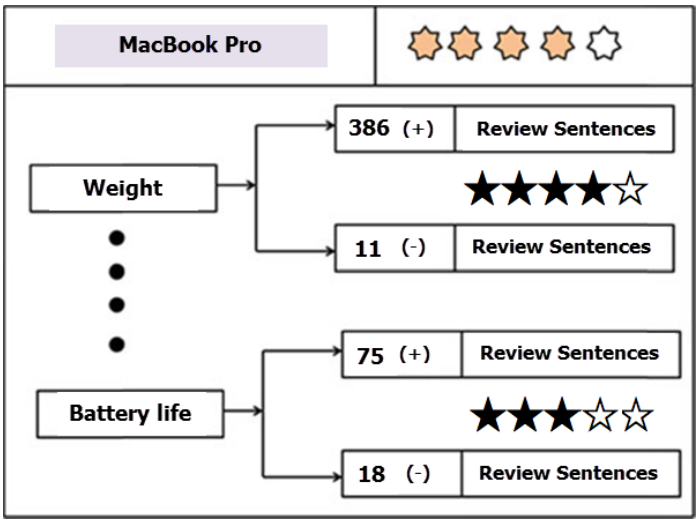

Figure 1. Feature-based summary [27]

Reputation generation systems have interest in providing potential customers with sufficient information toward the target entity (product, movie, hotel . . .) to help them make the right decision toward it (buying, renting, booking . . .). Currently, a few reputation systems have been proposed to tackle the task of reputation generation using opinion mining techniques on user and customer reviews expressed in natural language. Yan *et al.* [11] were the first to propose a reputation system that combines opinion mining and opinion fusion techniques for the purpose of producing a single reputation value toward various products. The system firstly eliminates irrelevant reviews [28], then, the remaining reviews are grouped into different sets based on their semantic relations (latent semantic analysis and cosine metric), and finally, a single numerical reputation value is produced. Benlahbib and Nfaoui [12] used K-Means clustering algorithm to group similar movie reviews into the same cluster based on their semantic relations before generating a reputation value. The same authors [13] designed and built a hybrid reputation system that firstly combines Naïve Bayes and linear support vector machine (SVM) to separate user and customer reviews into positive and negative (document level sentiment analysis), then, it groups them into different sets based on semantic relations, and finally, a single reputation value is computed using weighted arithmetic mean.

3. PROPOSED SYSTEM

3.1. System overview

The proposed approach consists mainly on four steps:

– We collect movie and TV show reviews from IMDb in https://www.imdb.com/, website using the web scraping tool ScrapeStorm in https://www.scrapestorm.com/, then, we preprocess them.

– We train Multinomial Naïve Bayes model on the 5-class stanford sentiment treebank (SST-5) dataset in order to perform fine-grained sentiment analysis. The model classifies the collected reviews to five opinion groups: strongly negative, weakly negative, neutral, weakly positive and strongly positive.
– For each opinion group, we acquire the sum of user ratings and the sum of reviews semantic similarity. The semantic similarity between two reviews is computed as the cosine between their deep contextualized word embeddings (ELMo). These acquired statistics are used to compute a custom score for each opinion group.
– We compute the movie or TV show numerical reputation value based on the opinion groups' scores by applying the weighted arithmetic mean.

Figure 2 illustrates the work-flow of the reputation system (MTVRep).

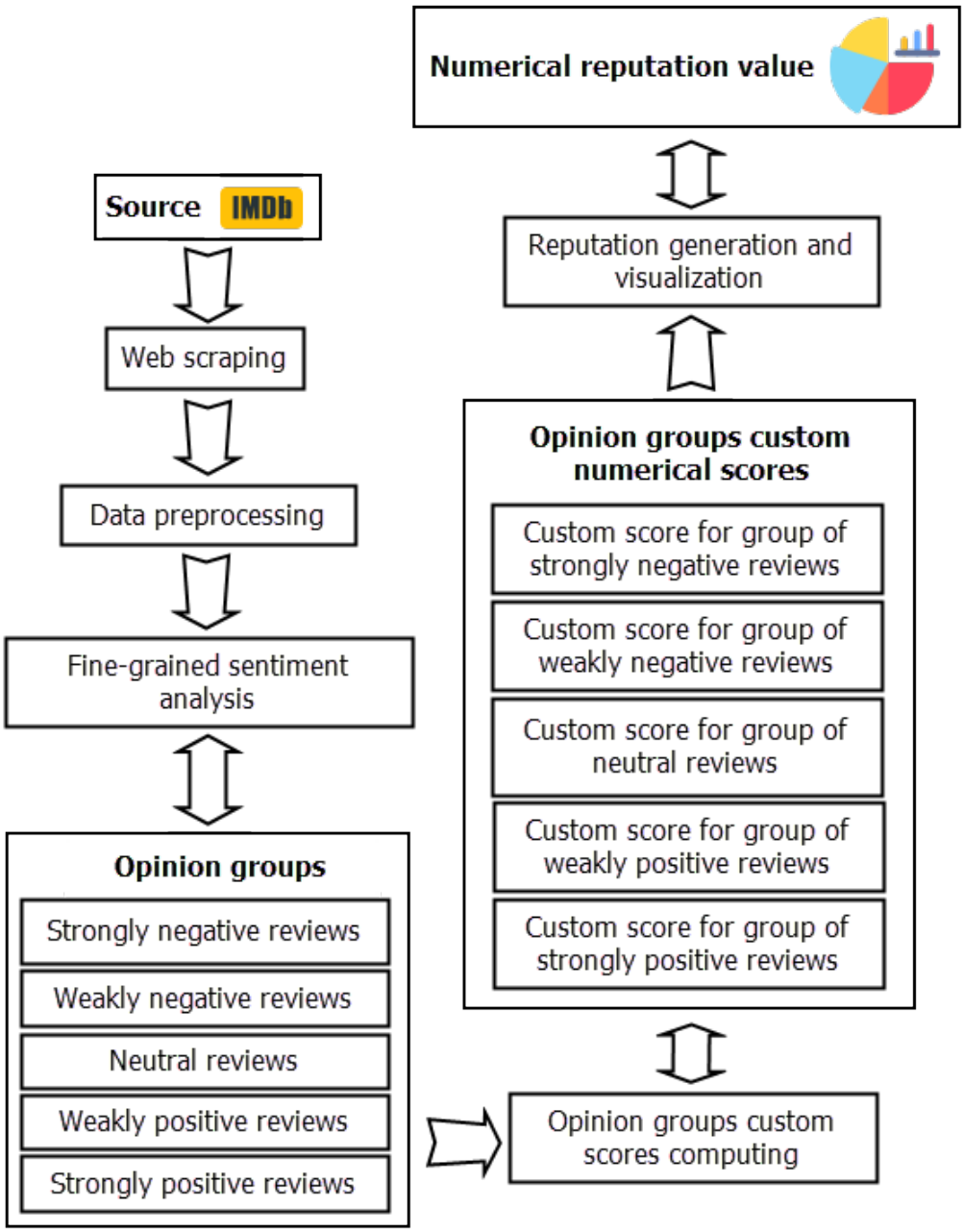

Figure 2. Reputation system pipeline

3.2. Fine-grained sentiment analysis

We classify the collected reviews into five opinion groups based on their sentiment intensities by applying the Multinomial Naïve Bayes model trained on the 5-class stanford sentiment treebank (SST-5) dataset. The reasons behind using Multinomial Naïve Bayes model are discussed in section 4.2.

3.3. Opinion groups custom scores

After separating movie and TV show reviews into five opinion groups: strongly negative, weakly negative, neutral, weakly positive and strongly positive, we compute a custom score for each opinion group based on the sum of their ratings and the sum of their reviews semantic similarity. The statistics of opinion groups are acquired by applying algorithm 1.

Algorithm 1: Opinion groups statistics acquisition

Define : $G^{polarity} = \{r_1^{polarity}, r_2^{polarity}, \ldots, r_n^{polarity}\}$: the opinion group that contains reviews which hold the sentiment orientation $polarity$.
$R^{polarity} = \{rr_1^{polarity}, rr_2^{polarity}, \ldots, rr_n^{polarity}\}$: the set of ratings attached to $G^{polarity}$ reviews.
$SS^{polarity}$: the sum of semantic similarity for $G^{polarity}$ reviews.
$SR^{polarity}$: the sum of ratings for $G^{polarity}$ reviews.
$NR^{polarity}$: the number of reviews in $G^{polarity}$.
$ELMo(r_i^{polarity})$: ELMo embeddings for review i from $G^{polarity}$.
$cos(ELMo(r_i^{polarity}), ELMo(r_j^{polarity}))$: the cosine similarity between ELMo embeddings for review i and j from $G^{polarity}$.
Input : Opinion groups, their lengths and their user ratings: $G^{polarity}$, $NR^{polarity}$ and $R^{polarity}$.
Output: Opinion groups' statistics: $SS^{polarity}$ and $SR^{polarity}$

$polarity \leftarrow [strongly\ negative, weakly\ negative, neutral, weakly\ positive, strongly\ positive]$

/* After applying the trained model on the collected movie and TV show reviews, we separate them into five opinion groups: strongly negative, weakly negative, neutral, weakly positive and strongly positive. For each opinion group, we acquire the sum of their reviews semantic similarity (cosine metric and ELMo embeddings) and the sum of their ratings */

for i *in* $polarity$ **do**
  $SS^i \leftarrow 0$
  $SR^i \leftarrow 0$
  for $j \leftarrow 1$ **to** NR^i **do**
    $SS^i \leftarrow SS^i + cos(ELMo(r_1^i), ELMo(r_j^i))$
    $SR^i \leftarrow SR^i + rr_j^i$
  end for
end for

By applying algorithm 1, we retrieve for each group, the sum of their ratings and the sum of their semantic similarity. We propose formula (1) to compute a custom score for each opinion group.

$$CS(G^{polarity}) = \frac{\frac{maxR \cdot SS^{polarity}}{NR^{polarity}} + \frac{SR^{polarity}}{NR^{polarity}}}{2} \quad (1)$$

Formula (1) could also be written as follows:

$$CS(G^{polarity}) = \frac{maxR \cdot SS^{polarity} + SR^{polarity}}{2 \cdot NR^{polarity}} \quad (2)$$

We denote:
$maxR$: Highest value of user ratings (5 or 10) depending on the range of ratings (1 to 5 or 1 to 10).
$SS^{polarity}$: Sum of similarity for reviews contained in opinion group $G^{polarity}$.
$SR^{polarity}$: Sum of user ratings in opinion group $G^{polarity}$.
$NR^{polarity}$: Number of reviews contained in opinion group $G^{polarity}$.

The custom score of each opinion group ranges between 1 and 5 or 1 and 10 depending on the range of user rating values. Since the cosine metric returns values in the range of [0,1], the average of the sum of semantic similarity for an opinion group is also between 0 and 1, therefore, we multiply the average of the sum of semantic similarity by 5 or 10 ($maxR$) to get a numerical value between 0 and 5 or 0 and 10, then, we add this value to the average of sum of ratings and we divide them by 2.

3.4. Reputation generation

We propose formula (3) (weighted arithmetic mean) to compute the movie or TV show reputation value.

$$Rep(E) = \frac{\sum_{polarity} CS(G^{polarity}) \cdot NR^{polarity}}{\sum_{polarity} NR^{polarity}} \quad (3)$$

$CS(G^{polarity})$ is the custom score for opinion group $G^{polarity}$ computed by applying formula (1) or (2).

The movie or TV show reputation value has values in the range of [1, 5] or [1, 10] depending on the range of user ratings.

4. EXPERIMENTAL EVALUATION

4.1. Dataset gathering

We collect movie and TV show reviews and their numerical ratings from IMDb web site using the web scraping tool ScrapeStorm. Figure 3 depicts the structure of IMDb user reviews.

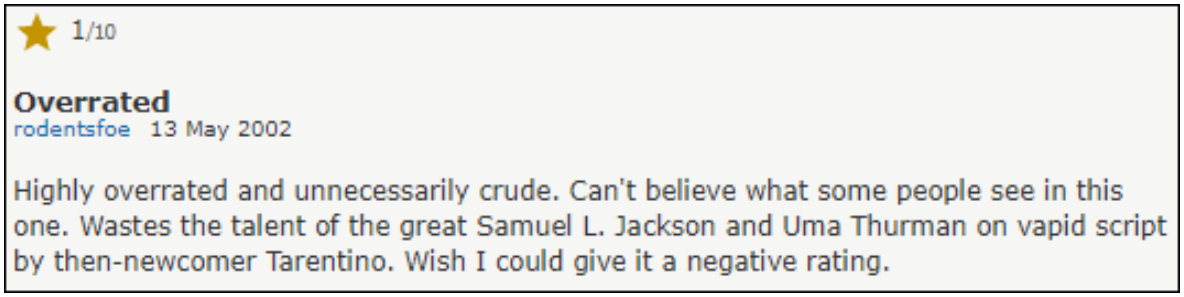

Figure 3. IMDb user reviews structure

The first ten datasets contain movie reviews and the remaining ten datasets contain TV show reviews. Table 2 shows the statistical information of the collected datasets.

Table 2. Statistical information of the collected datasets

	Movies	TV shows	Total
Number of reviews	1000	1000	2000
Number of entities	10	10	20

After collecting the reviews, we replace the missing rating values with the average of the ratings, then, we lowercase them and we remove punctuation marks and numbers.

4.2. Training phase and fine-grained opinion mining

We train the Multinomial Naïve Bayes model with SST-5 dataset. The training set contains 1092 strongly negative reviews, 2218 weakly negative reviews, 1624 neutral reviews, 2322 weakly positive reviews and 1288 strongly positive reviews. The test set contains 279 strongly negative reviews, 633 weakly negative reviews, 389 neutral reviews, 510 weakly positive reviews and 399 strongly positive reviews. Figure 4 depicts the distribution of training and test samples over the five classes.

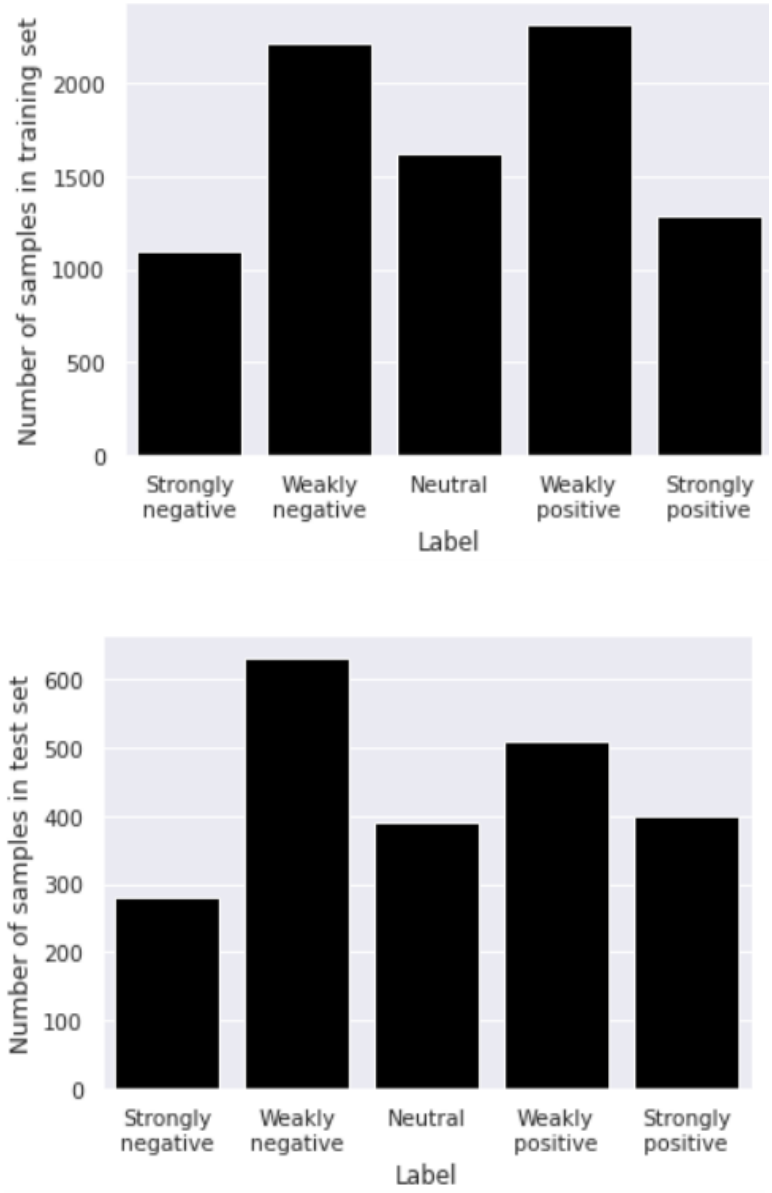

Figure 4. Number of samples in SST-5 training and test set

Before feeding the data to the classifier for training, we preprocess them by removing punctuation marks, numbers and whitespaces, then, we lowercase and lemmatize them. After preprocessing the data, we must choose which classifier we will apply and which features we will use. Since deep learning models require substantial computing power (High-performance CPUs, GPUs and RAM), we decided to work with one of

the four models: Random Forest, Logistic Regression, Multinomial Naïve Bayes and Linear support vector machine (SVM). The last two classifiers (Naïve Bayes and SVM) have been recognized as the most popular supervised machine learning algorithms for polarity classification [29]. For features selection, we have tried many combinations: unigrams, bigrams, trigrams, tf-idf unigrams, tf-idf bigrams and tf-idf trigrams. We discarded some popular models such as: word2vec and doc2vec because Wang *et al.* [30] have conducted experiments on Naïve Bayes, Logistic Regression and linear support vector classifier (SVC) for short text classification using tf-idf weighting, word2vec and paragraph2vec (doc2vec), and they have reported that tf-idf/counter feature has the highest accuracy, while word2vec next, and doc2vec has the lowest accuracy. Table 3 summarizes the classification result of the four classifiers on SST-5 dataset.

Table 3. Sentiment analysis classification result

	Macro average precision	Macro average recall	Macro average f1-score	Weighted average precision	Weighted average recall	Weighted average f1-score	Accuracy
Random Forest (unigrams)	0.40	0.31	0.30	0.40	0.36	0.33	0.36
Random Forest (bigrams)	0.34	0.29	0.28	0.34	0.32	0.31	0.32
Random Forest (trigrams)	0.29	0.23	0.20	0.31	0.23	0.22	0.23
Random Forest (tf-idf unigrams)	0.40	0.30	0.28	0.39	0.35	0.31	0.35
Random Forest (tf-idf bigrams)	0.34	0.29	0.28	0.34	0.32	0.31	0.32
Random Forest (tf-idf trigrams)	0.28	0.22	0.20	0.29	0.23	0.21	0.23
Multinomial Naive Bayes (unigrams)	0.43	**0.38**	**0.38**	0.43	**0.43**	**0.41**	**0.43**
Multinomial Naive Bayes (bigrams)	0.36	0.30	0.29	0.36	0.35	0.32	0.35
Multinomial Naive Bayes (trigrams)	0.31	0.26	0.24	0.31	0.29	0.26	0.29
Multinomial Naive Bayes (tf-idf unigrams)	**0.48**	0.34	0.29	**0.46**	0.41	0.34	0.41
Multinomial Naive Bayes (tf-idf bigrams)	0.38	0.29	0.24	0.38	0.35	0.29	0.35
Multinomial Naive Bayes (tf-idf trigrams)	0.29	0.24	0.19	0.30	0.29	0.23	0.29
Logistic Regression (unigrams)	0.42	0.37	0.37	0.42	0.41	0.39	0.41
Logistic Regression (bigrams)	0.38	0.28	0.23	0.37	0.34	0.27	0.34
Logistic Regression (trigrams)	0.36	0.23	0.18	0.35	0.28	0.22	0.28
Logistic Regression (tf-idf unigrams)	0.42	0.35	0.34	0.41	0.40	0.37	0.40
Logistic Regression (tf-idf bigrams)	0.43	0.28	0.23	0.41	0.35	0.27	0.35
Logistic Regression (tf-idf trigrams)	0.30	0.23	0.17	0.32	0.29	0.21	0.29
Linear SVM (unigrams)	0.38	0.37	0.37	0.39	0.40	0.39	0.40
Linear SVM (bigrams)	0.33	0.31	0.31	0.34	0.34	0.33	0.34
Linear SVM (trigrams)	0.31	0.25	0.22	0.32	0.29	0.25	0.29
Linear SVM (tf-idf unigrams)	0.38	0.38	0.38	0.39	0.41	0.39	0.41
Linear SVM (tf-idf bigrams)	0.33	0.31	0.31	0.34	0.34	0.33	0.34
Linear SVM (tf-idf trigrams)	0.31	0.27	0.25	0.31	0.30	0.27	0.30

From Table 3, we can see that Multinomial Naïve Bayes classifier achieves the best classification result when it's trained with unigrams. Logistic Regression and linear SVM classifiers also gave good result when they are trained with unigrams or tf-idf unigrams. The worst results are provided by Random Forest since it achieves a 0.36 accuracy in its best. Figure 5 depicts the confusion matrix of Multinomial Naive Bayes (unigrams) for SST-5 test set.

We mention that $BERT_{base}$ achieves a 0.45 accuracy and 0.40 macro average f1-score, GRU-RNN-WORD2VEC achieves a 0.45 accuracy and recursive neural tensor network achieves a 0.46 accuracy, Besides, deep learning algorithm takes a long time to train as shown in Table 4 due to the large number of parameters. Based on that, we have made the choice of applying Multinomial Naïve Bayes classifier since it achieves an accuracy of 0.43 and it doesn't require substantial computing power to be trained. Table 4 depicts the training time of bidirectional gated recurrent unit (Bi-GRU), bidirectional long short-term memory (Bi-LSTM), recurrent neural network (RNN) and multinomial naïve bayes (MNB) for SST-5 dataset.

One of the benefits of fine-grained opinion mining is that it provides a better understanding of the distribution of reviews over the five emotion classes, therefore, visualizing these five classes will help users and customers make up their minds about the target item (buying, renting).

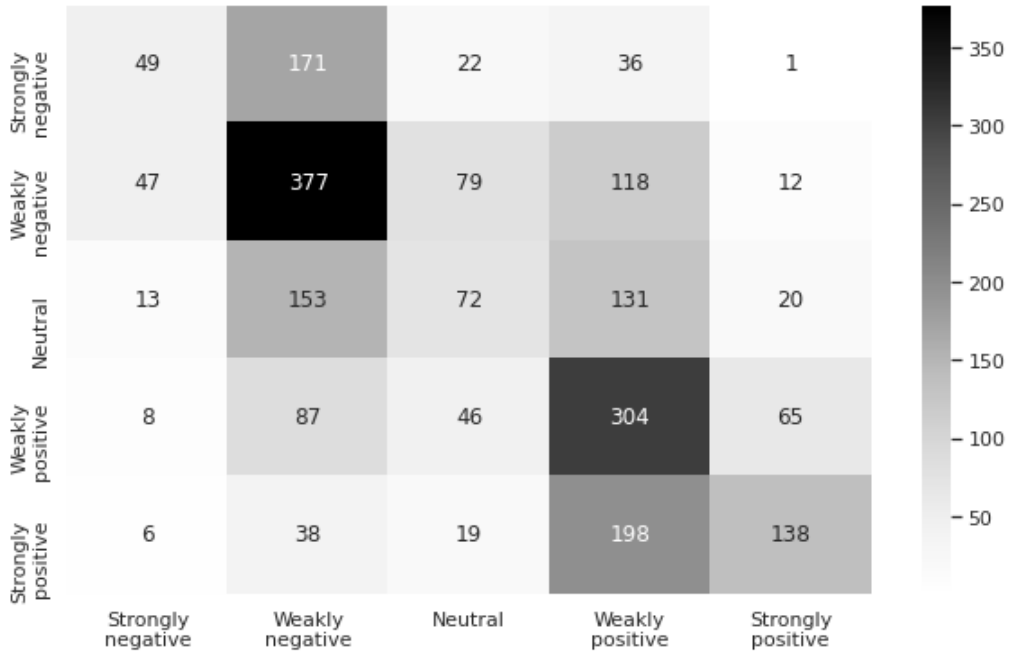

Figure 5. Confusion matrix of multinomial naive bayes (unigrams) for SST-5 test set

Table 4. Training time of bidirectional gated recurrent unit (Bi-GRU), bidirectional long short-term memory (Bi-LSTM), recurrent neural network (RNN) and multinomial naïve bayes (MNB) for SST-5 dataset

Model	Epochs	Batch size	Training time (seconds)
Bi-GRU	50	64	210.10
Bi-LSTM	50	64	180.25
RNN	50	64	85.26
MNB	–	–	**3.77**

4.3. Reputation evaluation

MTVRep offers a holistic reputation visualization form as shown in Figure 6 by depicting the numerical reputation value and the distribution of reviews over the five emotion classes, Table 5 shows comparison results between MTVRep and previous studies in term of visualizing reputation.

An important issue that was neglected in the past research on reputation generation is identifying the sentiment strength during the phase of opinion mining. Actually, existing studies have only focused on classifying reviews as positive or negative, disregarding sentiment intensity. Therefore, we propose MTVRep, a movie and TV show reputation system that combines fine-grained sentiment analysis and semantic analysis for the purpose of generating and visualizing reputation toward movies and TV shows. Table 6 depicts the features exploited by previous studies and MTVRep during reputation generation and visualization.

In order to evaluate the performance of MTVRep in generating accurate reputation values toward various movies and TV shows, we compared it with Yan *et al.* [11] reputation system. We set the opinion fusion threshold t_0 to 0.15 since the authors mentioned that their reputation system performs in its best when $t_0 = 0.15$. We applied the two reputation systems on the twenty collected datasets. The chosen evaluation measure is the squared error between the movie or TV show IMDb weighted average ratings and the numerical reputation value computed by one of the two reputation systems.

The formula of the squared error is: $SE = (x_i - y_i)^2$ where x_i is the reputation value returned by one of the two systems and y_i is the IMDb Weighted Average Ratings toward the target movie or TV show. Figure 7 depicts the IMDb weighted average ratings for forrest gump movie.

According to IMDb in https://help.imdb.com/article/imdb/track-movies-tv/weighted-average-ratings/GWT2DSBYVT2F25SK?ref_=$helpsect_pro_{28}$# : *"IMDb publishes weighted vote averages rather than raw data averages. Various filters are applied to the raw data in order to eliminate and reduce attempts at vote*

stuffing by people more interested in changing the current rating of a movie than giving their true opinion of it. The exact methods we use will not be disclosed. This should ensure that the policy remains effective. The result, is a more accurate vote average."

The motivation behind choosing the squared error instead of absolute error resides in the fact that reputation systems don't tolerate high error values. Consequently, the squared error will penalize large errors more. Figure 8 and 9 show the comparison result between the two reputation systems over the twenty datasets.

As illustrated in Figure 8, MTVRep produces the nearest reputation value to IMDb weighted average ratings for the first ten datasets that contain movie reviews compared to reputation system [11]. We observe that the squared error of reputation system [11] exceeds 2.5 in dataset 1, dataset 4, dataset 7 and dataset 9. We also observe that the squared error of MTVRep doesn't surpass 0.1 in dataset 3, 5 and 10, which implies that the system generates accurate reputation values toward movies since the highest squared error achieved by MTVRep is 1.87 (dataset 6).

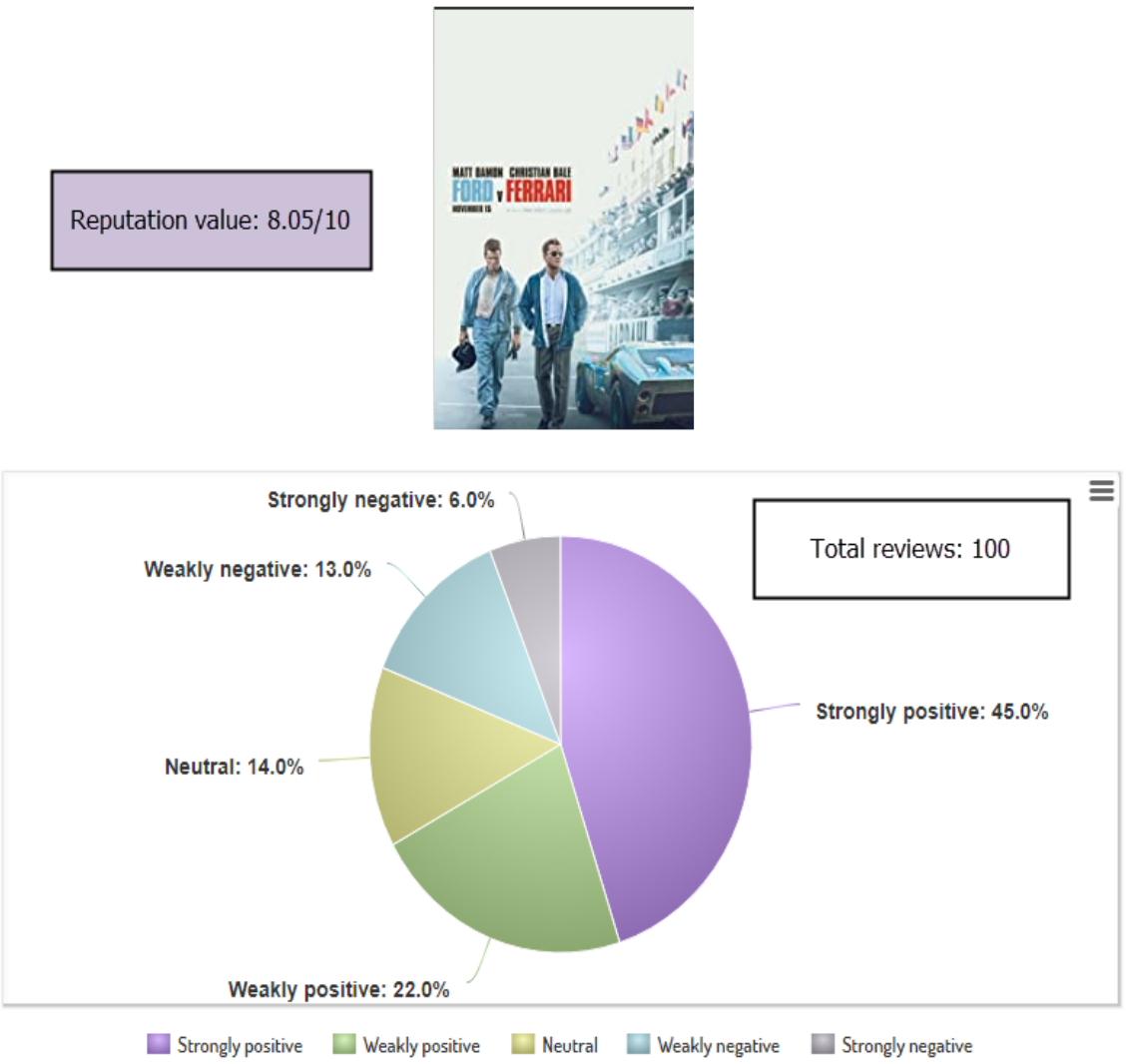

Figure 6. Reputation visualization

Table 5. Comparison results: Reputation visualization

Work	Distribution of reviews polarity	Numerical reputation value
[11]	✗	✓
[12]	✗	✓
[14]	✗	✓
[13]	✓	✓
MTVRep	✓	✓

Table 6. Comparison results: Exploited features

Work	Semantic	Sentiment (binary)	Sentiment (fine-grained)
[11]	✓	✗	✗
[12]	✓	✗	✗
[14]	✓	✓	✗
[13]	✓	✓	✗
MTVRep	✓	✗	✓

Figure 7. IMDb weighted average ratings

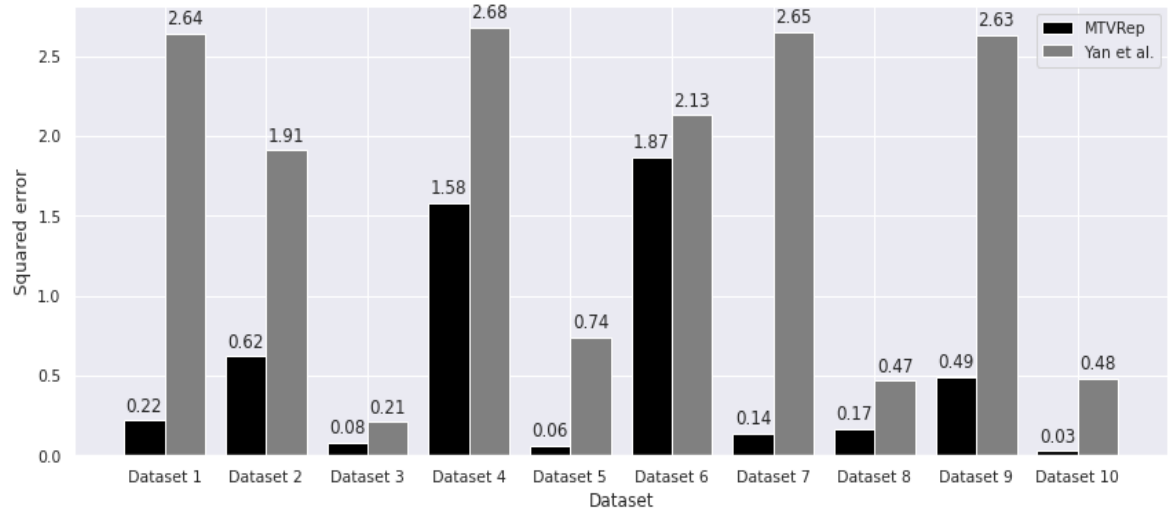

Figure 8. Squared error comparison result: Dataset 1 to dataset 10

Figure 9 shows that except for dataset 15, MTVRep outperforms Yan *et al.* [11] reputation system on all the remaining nine datasets that contain TV show reviews. We also observe that the squared error of reputation system [11] exceeds 3.5 in dataset 20, on the other hand, MTVRep doesn't exceed 1.44 in its worst. We conclude that the proposed reputation system MTVRep performs well in generating and visualizing reputation for movies and TV shows since it produces the nearest reputation value to IMDb weighted average ratings for both movies and TV shows compared to Yan *et al.* [11] reputation system.

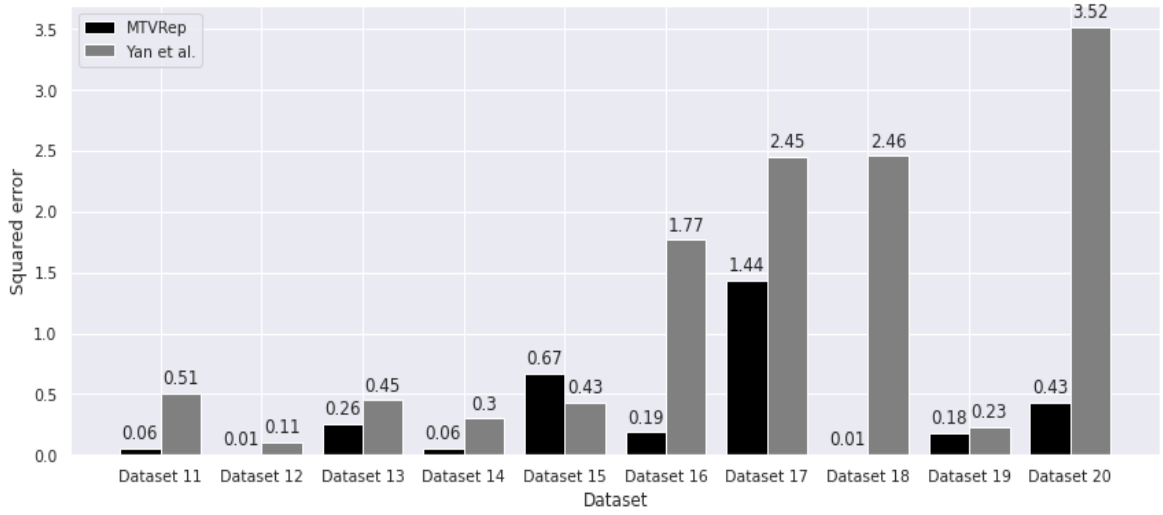

Figure 9. Squared error comparison result: Dataset 11 to dataset 20

5. CONCLUSION, SUMMARY AND FUTURE DIRECTION

In this paper, we have proposed MTVRep, a system that combines fine-grained opinion mining and semantic analysis for the purpose of generating and visualizing reputation toward movies and TV shows. The web scraping tool ScrapeStorm was used to collect 2000 movie and TV show reviews and their numerical ratings from IMDb, and Multinomial Naïve Bayes classifier was trained on SST-5 dataset to perform fine-grained opinion mining task. Experimental studies showed that MTVRep outperforms Yan *et al.* reputation system since it produces the nearest reputation values to the ground truth (IMDb weighted average ratings) for both movies and TV shows. We believe that MTVRep could be integrated in any platform where users share their reviews and ratings freely toward movies and TV shows.

Future works will focus on, using more sophisticated models for opinion mining such as BERT and XLNet, exploiting further features some of which are user credibility, review time and review helpfulness, and incorporating aspect based opinion mining to enhance the reputation visualization form by showing more useful information toward the target movie or TV show (aspects).

REFERENCES

[1] Y.-H. Hu, Y.-L. Chen, and H.-L. Chou, "Opinion mining from online hotel reviews a text summarization approach," *Inf. Process. Manage.*, vol. 53, no. 2, pp. 436–449, Mar. 2017.

[2] K. Bafna and D. Toshniwal, "Feature based summarization of customers' reviews of online products," *Procedia Computer Science*, vol. 22, pp. 142-151, 2013.

[3] L. Zhuang, F. Jing, and X.-Y. Zhu, "Movie review mining and summarization," *Proceedings of the 15th ACM international conference on Information and knowledge management*, pp. 43-50, 2006.

[4] T. Hou, B. Yannou, Y. Leroy, and E. Poirson, "Mining customer product reviews for product development: A summarization process," Expert Systems with Applications, vol. 132, pp. 141–150, 2019.

[5] M. Hu and B. Liu, "Mining opinion features in customer reviews," *Proceedings of the 19th National Conference on Artifical Intelligence, ser. AAAI'04*, pp. 755–760, 2004.

[6] J. Lovinger, I. Valova, and C. Clough, "Gist: General integrated summarization of text and reviews," *Soft Comput.*, vol. 23, no. 5, pp. 1589-1601, 2019.

[7] S. Pecar, "Towards opinion summarization of customer reviews," *Proceedings of ACL 2018, Student Research Workshop. Melbourne, Australia: Association for Computational Linguistics*, pp. 1-8, 2018.

[8] K. Zhang, R. Narayanan, and A. Choudhary, "Voice of the customers: Mining online customer reviews for product feature-based ranking," *Proceedings of the 3rd Wonference on Online Social Networks*, 2010.

[9] M. Hu and B. Liu, "Mining and summarizing customer reviews," *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 168–177, 2004.

[10] S. Blair-Goldensohn, K. Hannan, R. McDonald, T. Neylon, G. Reis, and J. Reynar, "Building a sentiment summarizer for local service reviews," *WWW Workshop on NLP Challenges in the Information Explosion Era (NLPIX)*, 2008.
[11] Z. Yan, X. Jing, and W. Pedrycz, "Fusing and mining opinions for reputation generation," *Inf. Fusion*, vol. 36, no. C, pp. 172–184, 2017.
[12] A. Benlahbib and E. H. Nfaoui, "An unsupervised approach for reputation generation," *Procedia computer science*, vol. 148, pp. 80–86, 2019.
[13] A. Benlahbib and E.-H. Nfaoui, "A hybrid approach for generating reputation based on opinions fusion and sentiment analysis," *Journal of Organizational Computing and Electronic Commerce*, vol. 30, no. 1, pp. 9–27, 2020.
[14] A. Benlahbib, A. Boumhidi, and E. H. Nfaoui, "A logistic regression approach for generating movies reputation based on mining user reviews," *International Conference on Intelligent Systems and Advanced Computing Sciences (ISACS)*, pp. 1–7, 2019.
[15] A. Benlahbib and E. H. Nfaoui, "Aggregating customer review attributes for online reputation generation," *IEEE Access*, vol. 8, pp. 96550–96564, 2020.
[16] P. Xu, A. Madotto, C.-S. Wu, J. H. Park, and P. Fung, "Emo2Vec: Learning generalized emotion representation by multi-task training," *Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis. Brussels, Belgium: Association for Computational Linguistics*, pp. 292–298, 2018.
[17] J. Mu, S. Bhat, and P. Viswanath, "All-but-the-top: Simple and effective postprocessing for word representations," *arXiv preprint arXiv:1702.01417*, 2017.
[18] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1631–1642, 2013.
[19] Y. Wang, A. Sun, J. Han, Y. Liu, and X. Zhu, "Sentiment analysis by capsules," *Proceedings of the 2018 World Wide Web Conference*, pp. 1165–1174, 2018.
[20] Y. Yang, "Convolutional neural networks with recurrent neural filters," *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 912–917, 2018.
[21] B. McCann, J. Bradbury, C. Xiong, and R. Socher, "Learned in translation: Contextualized word vectors," *Advances in Neural Information Processing Systems*, pp. 6294-6305, 2017.
[22] M. Munikar, S. Shakya, and A. Shrestha, "Fine-grained sentiment classification using bert," *Artificial Intelligence for Transforming Business and Society (AITB)*,vol. 1, pp. 1–5, 2019.
[23] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.
[24] S. Brahma, "Improved sentence modeling using suffix bidirectional lstm," *arXiv preprint arXiv:1805.07340*, 2018.
[25] M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2227–2237, 2018.
[26] D. Shen, G. Wang, W. Wang, M. R. Min, Q. Su, Y. Zhang, C. Li, R. Henao, and L. Carin, "Baseline needs more love: On simple word-embedding-based models and associated pooling mechanisms," *arXiv preprint arXiv:1805.09843*, 2018.
[27] A. Kangale, S. K. Kumar, M. A. Naeem, M. Williams, and M. K. Tiwari, "Mining consumer reviews to generate ratings of different product attributes while producing feature-based review-summary," *Intern. J. Syst. Sci.*, vol. 47, no. 13, pp. 3272–3286, 2016.
[28] J.-z. Wang, Z. Yan, L. T. Yang, and B.-x. Huang, "An approach to rank reviews by fusing and mining opinions based on review pertinence," *Inf. Fusion*, vol. 23, no. C, pp. 3–15, 2015.
[29] J. Shawe-Taylor and S. Sun, "A review of optimization methodologies in support vector machines," *Neurocomput.*, vol. 74, no. 17, pp. 3609–3618, 2011.
[30] Y. Wang, Z. Zhou, S. Jin, D. Liu, and M. Lu, "Comparisons and selections of features and classifiers for short text classification," *IOP Conference Series: Materials Science and Engineering*, vol. 261, no. 1, 2017.

BIOGRAPHIES OF AUTHORS

Abdessamad Benlahbib has received his Master degree in Computer Science. Currently, he is pursuing his Ph.D. studies at the Faculty of Science Dhar El Mahraz, Fez, Morocco. His research interests concern the application of natural language processing (NLP) techniques for decision making in E-commerce platforms. He has published several papers in journals and conferences in the area of computer science (e.g., IEEE Access, Journal of Organizational Computing and Electronic Commerce, International Journal of Electrical and Computer Engineering and Procedia Computer Science). Abdessamad can be contacted at abdessamad.benlahbib@usmba.ac.ma

El Habib Nfaoui is currently a Professor of Computer Science at the University of Sidi Mohamed Ben Abdellah, Fez, Morocco. He received his PhD in Computer Science from the University of Sidi Mohamed Ben Abdellah, Morocco, and the University of Lyon, France, under a Cotutelle agreement (doctorate in joint-supervision), in 2008, and then his HU Diploma (Accreditation to supervise research) in Computer Science, in 2013, from the University of Sidi Mohamed Ben Abdellah. His current research interests include Information Retrieval, Language Representation Learning, Machine learning and Deep learning, Web mining and Text mining, Semantic Web, Web services, Social networks, and Multi-Agent Systems. Dr. El Habib Nfaoui has published in international reputed journals, books, and conferences, and has edited seven conference proceedings and special issue books. He has served as a reviewer for scientific journals and as program committee of several conferences. He is co-founder and Chair of the IEEE Morocco Section Computational Intelligence Society Chapter. He is a co-founder and an executive member of the International Neural Network Society Morocco regional chapter. He co-founded the International Conference on Intelligent Computing in Data Sciences (ICSD2017) and the International Conference on Intelligent Systems and Computer Vision (ISCV2015).