An Enhanced SVM Model for Implicit Aspect Identification in Sentiment Analysis

Halima Benarafa[1], Mohammed Benkhalifa[2], Moulay Akhloufi[3]
Algorithms, Networks, Intelligent Systems and Software Engineering (ANISSE),
Department of Computer Science, Faculty of Sciences, Mohammed V University in Rabat, Morocco[1,2]
Perception, Robotics, and Intelligent Machines, Research Group (PRIME),
Department of Computer Science, Moncton University, Moncton, NB, Canada[3]

Abstract—**Opinion Mining or Sentiment Analysis (SA) is a key component of E-commerce applications where a vast number of reviews are generated by customers. SA operates on aspect level where the views are expressed on a specific aspect of a product and have a big influence on the customers' choices and businesses' reputation. Aspect Based Sentiment Analysis (ABSA) is the task of categorizing text by aspect and identifying the sentiment attributed to it. Implicit Aspect Identification (IAI) is a subtask of ABSA. This paper aims to empirically investigate how external knowledge (e.g. WordNet) is integrated into SVM model to address some of its intrinsic issues when dealing with classification. To achieve this research goal, we propose an approach to improve Support Vector Machines (SVM) model to deal with IAI. Using WordNet (WN) semantic relations, we suggest an enhancement for the SVM kernel computation. Experiments are conducted on three benchmark datasets of products, laptops, and restaurant reviews. The effects of our approach are examined and analyzed according to three criteria: (i) kernel function used, (ii) different experimental settings, and (iii) SVM behavior towards Overfitting and Underfitting. The research finding of our work is that the integration of external knowledge (e.g. WordNet) is experimentally proved to be significantly helpful to SVM classification for IAI and especially for addressing Overfitting and Underfitting that are considered as two of the main structural SVM issues. The empirical results demonstrate that our approach helps SVM (i) improve its performance for the three considered kernels and under different experimental settings, and (ii) deal better with Overfitting and Underfitting.**

Keywords*—*Implicit aspect-based sentiment analysis; machine learning; supervised approaches; support vector machines; wordnet; lesk algorithm

Abbreviations

ABSA Aspect Based Sentiment Analysis
ACD Aspect Category Detection
ATE Aspect Term Extraction
IAI Implicit Aspect Identification
IAT Implicit Aspect Term
IR Improvement Rate
LDA Latent Dirichlet Allocation
LSTM Long Short Term Memory
NLP Natural Language Processing
POS Part Of Speech
RNN Reccurent Neural Network
SA Sentiment Analysis
SVM Support Vector Machines
WN WordNet
WSD Word Sense Disambiguation

I. Introduction

Sentiment Analysis (SA), also known as opinion mining, is a research area in the field of Natural Language Processing (NLP) [1] that aims to display emotions and automatically identify the sentiments conveyed in text. SA studies have been conducted at three granularity levels: document level [2], sentence level [3], and aspect level [4]. In Document-level Sentiment Analysis, the entire document is analyzed to determine whether it expresses a positive or negative sentiment. However, in Sentence-level Sentiment Analysis, the opinion of each sentence in the document is analyzed. In Aspect-Based Sentiment Analysis (ABSA), opinions regarding each aspect of the text's existing entities are collected.

The majority of studies are interested in aspect identification task since it is the key task in aspect-level SA. Aspects can be either implicit or explicit. Explicit aspect extraction has attracted a lot of interest, whereas implicit aspects haven't received much attention. Explicit aspects are defined as specific terms that are explicitly stated in the document, they can be expressed using a noun or noun phrase. On the other side, an implicit aspect is not explicitly stated in the text. It takes the form of an adjective, verb, or adverb as shown in [5], [6], and [7]. Implicit aspects are crucial since they can capture the emotions expressed in the text and improve the Opinion Mining Task.

In this study, we propose a method for enriching SVM model by combining its basic kernel function with similarity function inspired from Lesk algorithm [8] when applied to Word Sense Disambiguation (WSD) introduced by Weaver et al. [9]. WSD is the process of automatically assigning a meaning to the ambiguous words in a given context, as defined in [10], [11] and [12]. According to the original Lesk algorithm, a word's appropriate meaning in a particular context is one that has the maximum degree of overlap between its dictionary definition and the given context.

In this paper, we use the fundamental idea of Lesk Algorithm for WSD. However, the originality of our work is established on two different levels: (i) The idea logic: We use WordNet dictionary (WN), developed in [13], to design a similarity function between terms inspired from the Lesk algorithm. We then use this function to create a novel SVM Kernel that assigns higher weights to semantically similar words in terms of the degree of influence they have on classifying new observations. (ii) Model construction: Our similarity function amplifies the similarity score between terms by first squaring the original score and then adding 1. This new formulation ensures significantly greater similarity scores for terms with similar semantic properties. Nevertheless, it maintains the same basic SVM Kernel value for words with different meanings.

We prepare several experiments in accordance with protocols that are intended to support the goals of our investigation. The key findings of our study are summarized as follows: (i) Our method enhances SVM's performance for the three kernels Gaussian, Anova, and Bessel, for the three considered datasets and under different experimental settings, and (ii) Our approach helps SVM perform better even when dealing with Overfitting and Underfitting which are known to be serious intrinsic issues for SVM classification.

The breakdown of the paper's structure is as follows. Related works on Aspect-based SA are discussed in the second section. Our proposed approach is described in the third section. The experimental setting is provided in section four, which is followed by a section on the results and discussion. The final section concludes this work.

II. Related Works

There are two major types of techniques used for Aspect Identification. Lexicon based approaches mainly include dictionary-based methods and corpus-based methods, where as machine learning approaches [15] and deep learning-based approaches [14] include supervised, unsupervised, or semi supervised learning methods.

Finding co-occurrence patterns of opinion words with context-specific orientation is the goal of corpus-based approaches. These techniques rely on syntactic patterns and seed opinion words to find additional opinion words and their orientation in domain corpora [16].

Dictionary-based techniques are methods that make use of WordNet or any other dictionary semantic relations. The work in [17], is an earlier dictionary-based method to identify aspects conveyed by adjectives. The authors of [18] perform an implicit aspect identification task for adjectives and verbs using definition and synonym relations extracted from WordNet. In [19], authors propose a new hybrid model for implicit aspect identification that uses semantic relations combined with a frequency-based method and supervised classifiers.

In [20] and [21], two of the most well-known co-occurrence-based approaches are presented. In [20], Schouten et al. predict implicit aspects according to the co-occurrence frequency between explicit aspects and opinion terms. Potential implicit aspects are determined based on a defined threshold value. In [21], the training data are enhanced by the use of WordNet's semantic relations and the co-occurrence score is computed for each extracted implicit aspect and its WordNet synsets. Additional co-occurrence methods are presented in [22] and [23]. The researchers Devi et al. [22] proposed a novel method to detect implicit aspects from opinionated documents using the co-occurrence of aspects with feature indicators and ranking the pair according to how frequently they co-occur. To determine how well a given candidate implicit aspect matches an opinion word, Rana et al. [23] identified implicit aspects using the co-occurrence approach and normalized Google distance.

Traditional machine learning techniques have been frequently used for ABSA. In [24], Sivakumar et al. make use of semantic relatedness between aspect term and opinion sentence to improve some machine learning algorithms for sentiment classification task. Gupta et al. [25], use an ensemble machine learning technique to perform ATE task. They combine the output of different supervised learning algorithms using a majority voting technique. Topic modeling, an unsupervised machine learning technique, has been widely applied to ACD. [26], [27], and [28] all make use of the well-known topic modeling technique Latent Dirichlet Allocation (LDA). García-Pablos et al. [26] suggest an unsupervised system called W2VLDA. To conduct ACD and sentiment classification, the system uses LDA combined with a Maximum Entropy classifier and word embedding. In [27], Poria et al. provide an original LDA technique to group aspect terms into corresponding aspect categories. To enhance the clustering process, semantic similarity between two words is used. Pathik et al. [28] suggest an unsupervised model for ACD using LDA in combination with linguistic rules. To perform ACD, Aspects are first ranked according to their probability distribution values and then clustered into predefined categories using domain knowledge with frequent terms.

Deep learning algorithms have recently begun to be used for ABSA after experiencing great success across a number of application domains. A recent work, [29], provides a hybrid method for detecting implicit aspects that combines a recurrent neural network (RNN) with a similarity function from spaCy and similarity metrics based on WordNet. The authors of [30] suggest a deep learning-based topic-level model for sentiment analysis. They performed ACD and sentiment classification using an LSTM network with a topic-level attention mechanism. Authors in [31] propose a two-step unsupervised model that combines deep learning techniques with language patterns in order to improve the ATE task. First, they extract aspects using a rule-based technique, and then they prune the pertinent aspects using fine-tuned word embedding. The extracted elements from the first phase are used as labeled data in the second phase to train the attention-based deep learning model.

There are numerous challenges and limitations for related works. Some of them conduct evaluations of their proposed models under optimal conditions without considering special situations like Overfitting and Underfitting. Others do not test their models on multiple experimental settings to figure out how they behave in different situations including non-ideal conditions. In addition to the aforementioned general shortcomings, some directly related approaches suffer from particular limitations. It is important to note that every study addresses the same problem, namely "Implicit Aspect Identification". They do, however, operate at various levels. While the techniques proposed in [18] and [19] concentrate on improving training data quality by acting at the data level which is a less challenging level, the approach proposed in [26] and our suggested method operate at the algorithmic level by suggesting modifications or additions. The works in [21], [17], [20] and [28] are hybrid methods that operate at both data level and algorithmic level. The work in [17], treats only aspects implied by adjectives without considering verbs that are very important implicit aspect indicators. In [21], the category is given to a sentence if the greatest conditional probability is greater than the corresponding trained threshold. the main limitation of this technique is that it needs a sufficient amount of training data to work properly. The amount of training data needed to perform well presents also a limitation for the method proposed in [26] since additional text reviews are needed to compute the topic model and domain-based word embeddings. The technique proposed in [20] suffers from two limitations, the first one is the obvious need for labeled data, and the second one is selecting only one implicit feature for each sentence, since they are working on sentence-level and their datasets contain more than one implicit feature and some implicit aspects can be missed by the algorithm. A common limitation to all these directly related approaches and our technique is that they do not address broad aspects which are often omitted, like the "anecdotes/miscellaneous" aspect on the Restaurant dataset [34]. Unlike [20], [21], and [26], our technique doesn't require a huge amount of training data to work properly.

Our research concentrates on implicit aspect-level sentiment analysis and its applications, and how to develop more semantic-oriented sentiment analysis. The motivation of our work is to address some of the structural issues of machine learning classification models applied to Implicit Aspect Identification like Overfitting and Underfitting. In this paper, the proposed approach is using semantic relations from WordNet lexical database for enhancing the SVM classification model so that it can better cope with some of its intrinsic issues. To achieve this motivation, we propose our approach which is specifically appropriate thanks to the fact that it captures similarity information between two aspect terms (from WordNet) and uses this similarity to increase the degree of influence on classification between these two aspect terms. Our approach operates at the SVM kernel which controls this degree of influence on classification between two aspect terms and therefore determines how each training term affects the final SVM classification results.

III. Proposed Approach

In this section, we describe our method, which is illustrated in Fig. 1. Its goal is to integrate relevant external knowledge, namely semantic knowledge obtained from WN lexical database into SVM Kernel calculation. For this purpose, we propose three new semantic kernel functions to SVM.

T_i and T_j are two implicit aspect terms (IAT) in the dataset, and Def_i and Def_j correspond to their respective sets of Wordnet definitions. Def_i and Def_j are defined as follows:

$$Def_i = \{subset_{i1}, ..., subset_{is}\}, s \in [1, n] \tag{1}$$

$$Def_j = \{subset_{j1}, ..., subset_{jt}\}, t \in [1, m] \tag{2}$$

Where n and m are respectively the numbers of definitions in Def_i and Def_j, $subset_{is}$ is the set of words representing the s^{th} definition in Def_i, and $subset_{jt}$ is the set of words representing the t^{th} definition in Def_j. The new kernels are computed according to the following formulas:

$$score(Def_i, Def_j) = \max NCW_{ij}(s,t), s \in [1, n], t \in [1, m] \tag{3}$$

$$sim(T_i, T_j) = score^2(Def_i, Def_j) + 1 \tag{4}$$

$$GaussianNew(T_i, T_j) = exp(-\gamma(\|T_i - T_j\|^2/sim(T_i, T_j))) \tag{5}$$

$$AnovaNew(T_i, T_j) = \sum_{k=1}^{n} exp(-\sigma((T_{ik}-T_{jk})/sim(T_i, T_j))^2)^d \tag{6}$$

$$BesselNew(T_i, T_j) = J_0(\sigma\|T_i - T_j\|) * sim(T_i, T_j) \tag{7}$$

Since equivalent word senses are commonly defined by the same terms, the score is determined by comparing word definitions collected from WordNet lexical database [13]. We can make the following assumption: for two terms, the more similar words that their definitions contain the more similar these two terms are. We inspire from Lesk algorithm [8] to create the proposed score. The Lesk algorithm suggests comparing two concepts using the number of common words in their glosses. First, the number of common terms between each subset in Def_i and each subset in Def_j is computed.

Let's note this number as follows:

$NCW_{ij}(s, t)$ = the number of common terms between $subset_{is} \in Def_i$ and $subset_{jt} \in Def_j$.

As stated in equation (3), the score is then computed as the maximum of all these numbers $NCW_{ij}(s, t)$.

Equation (4) shows how $sim(T_i, T_j)$ is obtained. This latter is calculated by adding 1 to the square of $score(Def_i, Def_j)$.

If T_i and T_j are dissimilar ($score(Def_i, Def_j) = 0$), then the new kernel between them is computed as follows :

- For Gaussian and Anova kernels, the new distance between T_i and T_j is set to the standard distance since $sim(T_i, T_j)$ is equal to 1.
- For Bessel kernel, J_0 (the Bessel function of the first kind) is set to its basic value since $sim(T_i, T_j)$ is equal to 1.

The score is squared to provide higher similarity of terms having a larger number of common words between subsets of their definitions.

In equations (5) and (6), the new SVM kernels ($GaussianNew(T_i, T_j)$ and $AnovaNew(T_i, T_j)$), are calculated by dividing the standard distances used in the original Gaussian and Anova kernel functions by the proposed similarity in equation (4). In each of these new kernels, the division of the distance by the proposed similarity aims to decrease the distance between similar terms and then increase the degree of influence they have on the classification of each others. In other terms, by decreasing the value inside the exponential function, the resulting value of the kernel is amplified for similar terms.

In equation (7), the new Bessel kernel is calculated by multiplying J_0, which is the Bessel function of the first kind, by

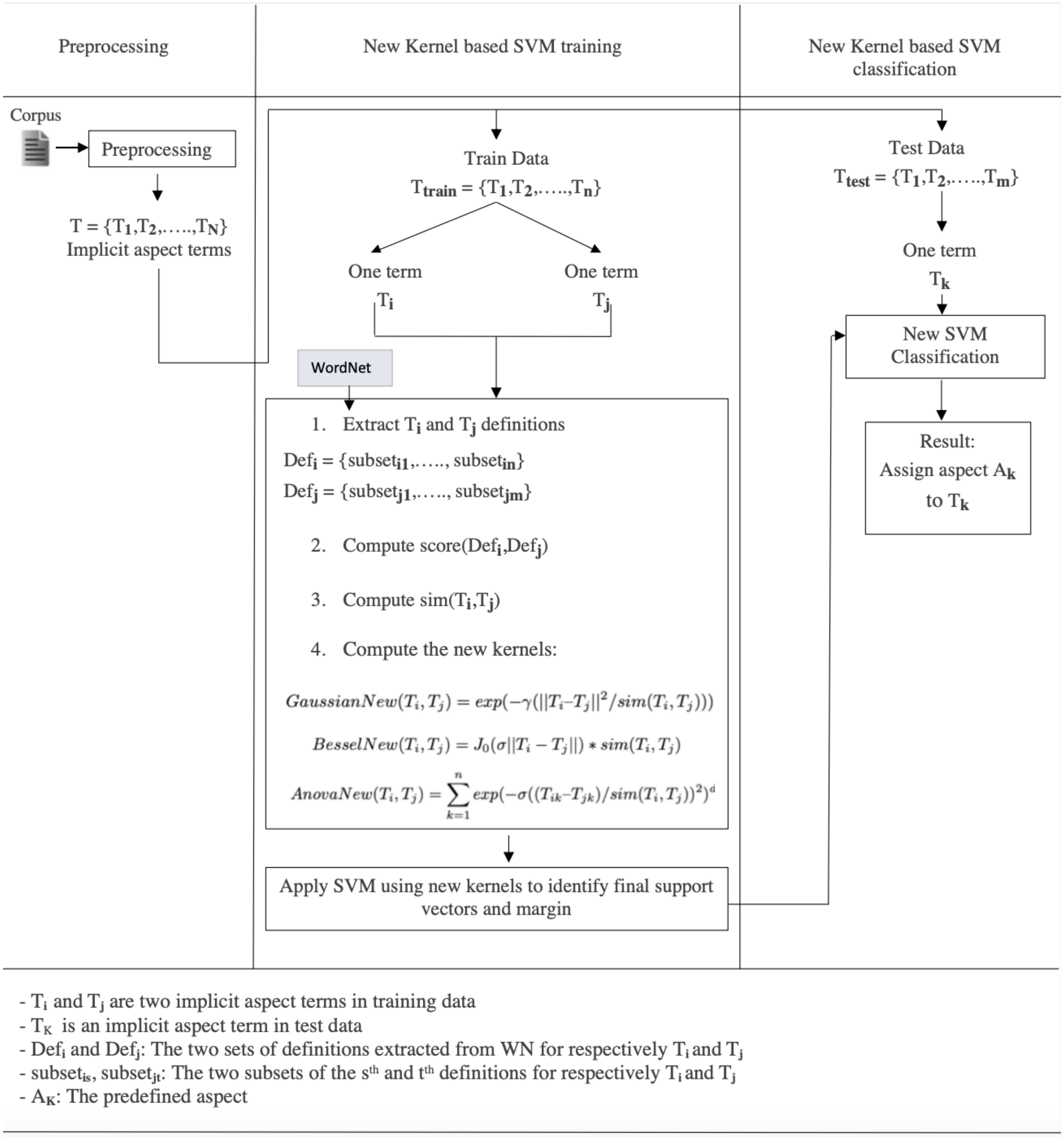

Fig. 1. Summary of our approach.

the proposed similarity in equation (4). The new Bessel kernel value is defined as J_0 multiplied by the proposed similarity function. Consequently, this resulting value is amplified which increases the degree of influence on classification between the nearest terms.

IV. Experiments and Results

This section presents the experiments conducted to assess our proposed method. The pre-processing techniques applied, the classifier used, the utilized datasets, the performance metrics adopted, and the experimental protocols implemented are all detailed below.

A. Experimental Setup and Protocols

1) Pre-processing: Pre-processing begins with corpus parsing to extract a list of adjectives and verbs using Part of Speech Tagger (POS). And then all stop words are removed from the initial list to create the final one.

2) Classifier used: Support Vector Machines (SVM) [32] are a group of supervised learning techniques for classification and regression. Putting more emphasis on classification task, the goal of SVM is to create a hyperplane that divides instances into distinct classes while maximizing the distance (or margin) with the closest data points, known as support vectors.

3) Datasets: To evaluate our technique, we used Restaurant, Products and Laptop datasets. Products dataset was created by Cruz-Garcia et al. [33] who manually labeled each IAT. This dataset is based on the customer review corpus described in [36]. It includes five corpora for various electronic products. The primary considered implicit aspects are functionality, performance, appearance, price, quality, weight, and size.

Restaurant dataset is used for SemEval-2014 ABSA task 4 [35]. It contains 3044 English sentences from Ganu et al.'s [34] restaurant reviews with five predetermined implicit aspects: price, food, ambiance, service, and anecdotes/miscellaneous.

Laptop dataset is a modified version of SemEval-2015 ABSA dataset for laptop domain [37]. This corpus is used for SemEval-2016 task 5 for Aspect Based Sentiment Analysis [38]. The primary addressed implicit aspects are operation performance, usability, price, quality, design features, portability, and connectivity.

4) Evaluation measures: Accuracy, precision, recall, and F1-score are the most widely utilized evaluation measures for assessing the model's performance. Accuracy is the proportion of correctly predicted samples. Precision, recall, and F1-score are employed instead of accuracy when the dataset is unbalanced since accuracy alone is insufficient. The F1-score is the equally weighted average of precision and recall [39].

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{8}$$

Where Precision is the percentage of correct predictions over all positive label samples, whereas recall is the percentage of correct predictions across all positive predictions.

5) Experimental protocols: Our experimental protocols are prepared in order to evaluate our method according to the following issues:

a) SVM behavior depending on kernel functions used,
b) SVM behavior under different experimental settings,
c) SVM behavior under Overfitting and Underfitting,

To lower the uncertainty of data splitting between testing and training data, 10-fold cross-validation is used in all experiments. The experimental protocols will be detailed in the following four subsections, with an emphasis on each protocol's intended purpose and how each protocol is designed to achieve its goal.

a) Kernel functions used: The main function of the kernel is to transform the input data into the required form. There are various types of kernels. In order to evaluate our approach, we used three different kernels.

Gaussian RBF kernel: The Gaussian RBF kernel is one of the most used kernels with SVM. This kernel function is preferred when we do not have any prior knowledge of the data. The equation of Gaussian RBF is presented as follows:

$$K(x,z) = exp(-\gamma\|x-z\|^2) \tag{9}$$

Where $\|x-z\|$ denotes the Euclidean distance between the two data points x and z, respectively. The parameter γ controls the Gaussian curve's shape and determines how each training sample affects the classification result.

Anova kernel: The ANOVA kernel is a radial basis function that is frequently used in kernel-based techniques, such as SVM. The ANOVA kernel is formulated as:

$$K(x,z) = \sum_{k=1}^{n} exp(-\sigma(x_k - z_k)^2)^d \tag{10}$$

Where x and z are two data points, and d denotes the ANOVA kernel's degree. The parameter σ influences both the border of the categorization problem and the shape of the ANOVA kernel.

Bessel kernel: The Bessel kernel is a radial basis function used in kernel-based methods in mathematics, such as SVM. The equation of Bessel kernel is given by:

$$K(x,z) = J_0(\sigma\|x-z\|) \tag{11}$$

Where x and z are two data points, J_0 is the Bessel function of the first kind, and $\|x-z\|$ is the Euclidean distance between them. The parameter σ impacts the boundary of the categorization problem, which also impacts the Bessel kernel's structure.

b) SVM behavior under different experimental settings: SVM is a machine learning classification technique whose performance depends not only on kernel function but also on its parameters. The most critical parameters are C, γ, and σ. Parameter C is the error penalty misclassification, It controls the trade-off between maximizing the margin and minimizing the misclassification error. Parameter γ determines the speed of the decrease of the similarity of two points as the distance between them increases. It is in charge of finding the balance between SVM abilities to fitting training data and to generalizing to testing data. Whereas parameter σ decides the boundary uniformity with respect to the quantity of nearby data points considered for Building this boundary. It decides the breath of its corresponding kernel. Thus, both parameters σ and γ determine how far the influence of each training instance reaches.

Protocol 1: Experiments for issues (a) and (b):

For our comparative protocol, we execute BasicSVM and NewSVM using a grid search with respect to combinations $\{C,\gamma\}$ (for Gaussian RBF Kernel) and $\{C,\sigma\}$ (for Anova and Bessel Kernels), where C, γ and σ range within $[2^{-5}, 2^{15}]$ interval, in order to obtain as many as possible significant values for performance ranging from Underfitting values up to Overfitting values.

As the parameter setting changes, the performance of each of NewSVM and BasicSVM models ranges from Minimum to Maximum values that correspond respectively to Underfitting and Overfitting situations. For each model, we identify from its performance range three different pertinent F1 performance values Minimum Value, Median Value, and Maximum Value. The identified Minimum and Maximum performance values are chosen to be different, if possible, from Underfitting and Overfitting values respectively. This is because Underfitting and Overfitting are treated separately in the next part of this section. Our protocol aims to conduct objective comparisons of F1 performances of both NewSVM and BasicSVM models. In fact, it compares each of the three identified F1 performances of each model to the F1 performance, of the other model, obtained under the same experimental parameter setting leading to the identified performance of the former model.

To deal with issues (a) and (b), we conduct our experiments according to the following protocol:

Protocol 1: kernel functions used and different experimental settings:

For each dataset from {Laptop,Products, Restaurant}:

For each Kernel from {Gaussian RBF, Anova, Bessel}:

If (the best performance is identified for BasicSVM) / (OR the best performance is identified for NewSVM):

Let's denote:
1) BestBasicSVM as the BasicSVM algorithm with the best F1-score performance. (OR BestNewSVM as the NewSVM algorithm with the best F1-score performance.)
2) $\text{NewSVM}_{Param-BestBasicSVM}$ as the NewSVM algorithm using the same parameters used by BestBasicSVM. (OR $\text{BasicSVM}_{Param-BestNewSVM}$ as the BasicSVM algorithm using the same parameters used by BestNewSVM.)
Compare F1-score average performances of $\text{NewSVM}_{Param-BestBasicSVM}$ and BestBasicSVM (OR Compare F1-score average performances of $\text{BasicSVM}_{Param-BestNewSVM}$ and BestNewSVM)

If (the median performance is identified for BasicSVM) / (OR the median performance is identified for NewSVM):

Let's denote:
1) MedianBasicSVM as the BasicSVM algorithm with the median F1-score performance. (OR MedianNewSVM as the NewSVM algorithm with the median F1-score performance.)
2) $\text{NewSVM}_{Param-MedianBasicSVM}$ as the NewSVM algorithm using the same parameters used by MedianBasicSVM. (OR $\text{BasicSVM}_{Param-MedianNewSVM}$ as the BasicSVM algorithm using the same parameters used by MedianNewSVM.)
Compare F1-score average performances of $\text{NewSVM}_{Param-MedianBasicSVM}$ and MedianBasicSVM (OR Compare F1-score average performances of $\text{BasicSVM}_{Param-MedianNewSVM}$ and MedianNewSVM)

If (the worst performance is identified for BasicSVM) / (OR the worst performance is identified for NewSVM):

Let's denote:
1) WorstBasicSVM as the BasicSVM algorithm with the worst F1-score performance. (OR WorstNewSVM as the NewSVM algorithm with the worst F1-score performance.)
2) $\text{NewSVM}_{Param-WorstBasicSVM}$ as the NewSVM algorithm using the same parameters used by WorstBasicSVM. (OR $\text{BasicSVM}_{Param-WorstNewSVM}$ as the BasicSVM algorithm using the same parameters used by WorstNewSVM.)
Compare F1-score average performances of $\text{NewSVM}_{Param-WorstBasicSVM}$ and WorstBasicSVM (OR Compare F1-score average performances of $\text{BasicSVM}_{Param-WorstNewSVM}$ and WorstNewSVM)

Compute all Improvement Rates (IR) of NewSVM over BasicSVM
Report F1-score averages and IR results

c) SVM behavior under overfitting and underfitting: We design a protocol that is intended to examine and compare the impact of Overfitting and Underfitting on the performance of NewSVM and BasicSVM with three kernels, Gaussian RBF, Anova, and Bessel. To accomplish this aim, our suggested protocol should:

1. Be built under conditions that cause SVM Underfitting and Overfitting. Generally, Overfitting and Underfitting are induced by respectively large values of C, γ and σ, and small values of C, γ and σ. The small and large values of these parameters are experimentally identified using grid search along with cross-validation.

The parameters γ and σ determine the extent of a single training example influence (γ is the hyper-parameter of Gaussian RBF Kernel, and σ is the hyper-parameter of Anova and Bessel kernels). When gamma and sigma are very small the model is too constrained and cannot capture the complexity of the data. Consequently, the region of influence of any selected support vector would include the whole training set. In addition to that, small values of γ and σ consider only nearby points in calculating the separation line. As a result, a low value of γ and σ will loosely fit the training dataset, which causes Underfitting. In contrast to small values, large values of γ and σ consider all the data points in the calculation of the separation line. Consequently, a high value of γ and σ will exactly fit the training dataset, which causes Overfitting.

Parameter C represents the error penalty for misclassification for SVM. The C parameter trades off correct classification of training examples against maximization of the decision function's margin. For larger values of C, a smaller margin will be accepted thus the model will be less tolerant, in other words, the model will be more specific and therefore this leads to Overfitting. A lower C will encourage a larger margin, therefore a simpler decision function at the cost of training accuracy, thus the model will be more tolerant to misclassifications, which causes Underfitting. In other words, C behaves as a regularization parameter in SVM.

2. Provide a measure to analyze the impact of Underfitting and Overfitting on SVM performance, in order to make a comparison between BasicSVM and NewSVM with regard to how they behave under Overfitting and Underfitting situations. In Overfitting, SVM has a good training performance and a bad test performance. In contrast, in Underfitting SVM performs poorly on both testing and training data. For assessing how sensitive both models are to Underfitting and Overfitting, we propose different measures that are presented and described in detail in section B "Results and Discussion".

Protocol 2: Experiments for issue (c):

We compare the performances of NewSVM and BasicSVM for each of the three kernels (Gaussian RBF, Anova, and Bessel) and for Overfitting and Underfitting conditions. For this comparison, we execute a grid search with respect to {C,γ} and {C,σ} combinations, where C, γ and σ range within $[2^{-5}, 2^{15}]$ interval. This range is chosen to be very large (with very small lower bound and very large upper bound) so that grid search results in many combinations of {C,γ} and {C,σ} from which we extract relevant values leading to Overfitting and Underfitting that are used to conduct our experimental comparisons of BasicSVM and NewSVM.

In fact, for each situation of Underfitting and Overfitting, grid search identifies several relevant combinations resulting in the same F1-score performance. Thus, for our comparative experiments, we select the combinations of the largest values $\{C_{max}, \gamma_{max}\}$ or $\{C_{max}, \sigma_{max}\}$ and the smallest values $\{C_{min}, \gamma_{min}\}$ or $\{C_{min}, \sigma_{min}\}$ (depending on the kernel used) for respectively Overfitting and Underfitting conditions.

Protocol 2: Overfitting and Underfitting: For each dataset from {Laptop,Products, Restaurant}:

For each Kernel from {Gaussian RBF, Anova, Bessel}:
For each Model from {NewSVM, BasicSVM}:
If Kernel = Gaussian RBF :
If Overfitting :
Select $\{C_{max}, \gamma_{max}\}$ for comparing NewSVM and BasicSVM
Else //Underfitting // :
Select $\{C_{min}, \gamma_{min}\}$ for comparing NewSVM and BasicSVM
If Kernel = Anova or Kernel = Bessel :
If Overfitting :
Select $\{C_{max}, \sigma_{max}\}$ for comparing NewSVM and BasicSVM
Else //Underfitting // :
Select $\{C_{min}, \sigma_{min}\}$ for comparing NewSVM and BasicSVM
Report F1-score average results of Model

B. Results and Discussion

The results of the experiments are shown and discussed in this part considering the following aspects:

- SVM behavior depending on kernel functions used,
- SVM behavior under different experimental settings,
- SVM behavior under Overfitting and Underfitting.

1) SVM behavior depending on kernel functions used and under different experimental settings:: Table I is defined to show the behavior of both BasicSVM and NewSVM models with respect to different experimental settings. It presents, on one hand, the F1-score average performances of BestBasicSVM, MedianBasicSVM, and WorstBasicSVM compared respectively to F1-score average performances of NewSVM$_{Param-BestBasicSVM}$, NewSVM$_{Param-MedianBasicSVM}$ and NewSVM$_{Param-WorstBasicSVM}$, and on the other hand, the F1-score average performance of BestNewSVM, MedianNewSVM, and WorstNewSVM compared respectively to F1-score average performances of BasicSVM$_{Param-BestNewSVM}$, BasicSVM$_{Param-MedianNewSVM}$ and BasicSVM$_{Param-WorstNewSVM}$. It outlines these comparisons for the three considered kernels and the three datasets. Table I reveals that NewSVM outperforms BasicSVM for all kernels and all datasets used (shown by positive IR for all cases). In fact, when we introduce our proposed similarity in SVM kernels this results in tuned kernel values and then enhances the classification performance. These tuned values are obtained by integrating the proposed similarity function in the three considered kernels (Gaussian RBF, Anova, and Bessel), which amplifies kernel values and then increases the level of influence between the nearest terms. As a result, the new kernel functions allow SVM to improve its classification performance.

In addition to global findings marked by positive performance improvement rates of NewSVM over BasicSVM, there are some noteworthy points that clearly show NewSVM's superiority:

a) We observe that NewSVM outperforms BasicSVM with the lowest, the middle, and the highest average IR over all kernels and datasets respectively for the best, the median, and the worst performances of both models. (IR average values are $\{5,78\%, 36,94\%\}$, $\{34,48\%, 118,97\%\}$, and $\{193,53\%, 211,89\%\}$, for respectively the best, the median, and the worst performances). NewSVM is shown to outperform BasicSVM for all cases but its outperformance rate changes with the level of the performance considered for comparison. Indeed, the best performance, that is chosen for any one of both models, usually corresponds to optimal hyperparameters for both NewSVM and BasicSVM. This fact allows this latter to reach high performances in general, and therefore not to be largely exceeded by NewSVM. Conversely, the worst performance, that is identified for any of both models, leads to the worst hyperparameters mainly for BasicSVM. Hence, this latter achieves its worst performance, which helps NewSVM to highly outperform it.

b) We notice that NewSVM outperforms BasicSVM with higher average IR over all kernels and datasets when best and median performances

TABLE I. IMPROVEMENT RATES OF NEWSVM OVER BASICSVM UNDER DIFFERENT EXPERIMENTAL SETTINGS FOR THREE DATASETS AND THREE KERNELS

	Restaurant			Products			Laptop			
Model	Gaussian	Anova	Bessel	Gaussian	Anova	Bessel	Gaussian	Anova	Bessel	Average-IR
BestBasicSVM	81.94%	81.94%	81.94%	77.27%	77.27%	77.02%	85.60%	85.60%	85.60%	
NewSVM$_{Param-BestBasicSVM}$	85.53%	86.56%	87.23%	81.13%	78.76%	80.99%	92.33%	92.47%	92.06%	
IR-BestBasicSVM	**4.38%**	**5.63%**	**6.45%**	**5%**	**1.93%**	**5.15%**	**7.86%**	**8.03%**	**7.55%**	**5.78%**
BasicSVM$_{Param-BestNewSVM}$	34.38%	81.94%	81.94%	71.64%	77.02%	58.42%	46.32%	85.60%	85.60%	
BestNewSVM	85.67%	87.29%	87.23%	81.13%	79.57%	81.14%	92.41%	92.73%	92.06%	
IR-BestNewSVM	**149.19%**	**6.53%**	**6.45%**	**13.25%**	**3.31%**	**38.39%**	**99.50%**	**8.33%**	**7.55%**	**36.94%**
MedianBasicSVM	53.64%	75.89%	49.59%	64.75%	77.02%	58.42%	69.11%	67.29%	64.13%	
NewSVM$_{Param-MedianBasicSVM}$	85.67%	84.50%	87.23%	79.67%	79.57%	81.14%	87.24%	86.66%	91.90%	
IR-MedianBasicSVM	**59.71%**	**11.35%**	**75.90%**	**23.04%**	**3.31%**	**38.72%**	**26.23%**	**28.79%**	**43.30%**	**34.48%**
BasicSVM$_{Param-MedianNewSVM}$	36.16%	75.89%	15.21%	76.31%	77.27%	77.02%	17.66%	85.60%	64.13%	
MedianNewSVM	85.53%	84.50%	86.51%	78.48%	77.90%	80.99%	87.24%	92.29%	91.90%	
IR-MedianNewSVM	**136.53%**	**11.35%**	**468.77%**	**2.84%**	**0.82%**	**5.14%**	**394%**	**7.82%**	**43.30%**	**118.97%**
WorstBasicSVM	24.95%	49.59%	15.21%	26.27%	58.91%	12.63%	31.63%	67.29%	23.88%	
NewSVM$_{Param-WorstBasicSVM}$	52.34%	79.35%	86.51%	62.85%	70.46%	80.88%	60.36%	86.66%	91.79%	
IR-WorstBasicSVM	**109.78%**	**60.01%**	**468.77%**	**139.25%**	**19.61%**	**540.38%**	**90.83%**	**28.79%**	**284.38%**	**193.53%**
BasicSVM$_{Param-WorstNewSVM}$	24.95%	49.59%	15.21%	26.27%	58.91%	12.63%	16.95%	67.29%	23.88%	
WorstNewSVM	52.34%	79.35%	86.51%	62.85%	70.46%	80.88%	60.36%	86.66%	91.79%	
IR-WorstNewSVM	**109.78%**	**60.01%**	**468.77%**	**139.25%**	**19.61%**	**540.38%**	**256.11%**	**28.79%**	**284.38%**	**211.89%**

are used for NewSVM than when they are used for BasicSVM (Average-IR(IR-BestNewSVM) ¿ Average-IR(IR-BestBasicSVM) and Average-IR(IR-MedianNewSVM) ¿ Average-IR(IR-MedianBasicSVM)). In fact, the newly included similarity into SVM kernels helps NewSVM to be much less sensitive to the change of setting, the error misclassification, and the influence of training data instances that are controlled by hyperparameters (C, γ, and σ). Whereas, BasicSVM remains very sensitive as usual to these factors. Therefore, the performances of NewSVM do not significantly change even when we change hyperparameters from values leading to its best, median, and worst performances to values corresponding respectively to the best, median, and worst performances of BasicSVM. At the same time, BasicSVM is generally penalized when its own parameters are changed to NewSVM parameters.

c) We also note that for the worst performances, NewSVM outperforms BasicSVM with higher average IR over all kernels and datasets (Average-IR(IR-WorstNewSVM) ¿ Average-IR(IR-WorstBasicSVM). However, NewSVM is shown to exceed BasicSVM with the same IR for every kernel and dataset except for the Gaussian kernel on Laptop dataset. This is simply explained by the fact that both models share the same hyperparameter values for their worst performances. In others terms, the values of the parameters that correspond to the worst performance of BasicSVM lead to the worst performance of NewSVM and vice versa.

To better show the behavior of both NewSVM and BasicSVM with respect to kernel functions for all datasets, we create Table II that represents an aggregated view of Table I. Indeed, Table II shows for each kernel function and for each dataset: (i) Average-F1-BasicSVM which is the average of F1-score performances of BestBasicSVM, MedianBasicSVM, and WorstBasicSVM, BasicSVM$_{Param-BestNewSVM}$, BasicSVM$_{Param-MedianNewSVM}$ and BasicSVM$_{Param-WorstNewSVM}$, (ii) Average-F1-NewSVM which is the average of F1-score performances of BestNewSVM, MedianNewSVM, and WorstNewSVM, NewSVM$_{Param-BestBasicSVM}$, NewSVM$_{Param-MedianBasicSVM}$ and NewSVM$_{Param-WorstBasicSVM}$ and (iii) IR which is the improvement rate of Average-F1-NewSVM over Average-F1-BasicSVM. From Table II, it can be observed that the average improvement rates of NewSVM over BasicSVM reach their highest values with Bessel kernel and their lowest values with Anova kernel for all datasets. This observation may be explained by the low BasicSVM performance with Bessel kernel and the high BasicSVM performance with Anova kernel. This shows that BasicSVM performance is one among other impacting factors of the improvement rate of NewSVM over BasicSVM.

2) SVM behavior under overfitting and underfitting:

a) Overfitting: To analyze the behavior of the new and original model in Overfitting conditions, and as stated previously in our protocol, the comparative experiments are conducted using the combination $\{C_{max}, \gamma_{max}\}$=[$\{32768, 32768\}$ for Gaussian kernel, and $\{C_{max}, \sigma_{max}\}$ = [$\{32768, 32768\}$ for Anova and Bessel kernels, corresponding to the largest values of parameters.

Table III shows F1-score averages for NewSVM model and BasicSVM model under Overfitting situations (each average is obtained across multiple folds). In Overfitting, the two models perform well on training data but badly on test data.

We provide three indicators in Table III that are utilized to measure how sensitive BasicSVM and NewSVM are to Overfitting.

Delta-test (Delta-test = F1-test(NewSVM) − F1-test(BasicSVM)) values are positive in all experiments in Table III. This demonstrates that NewSVM outperforms BasicSVM for all kernels and for all datasets, even in Overfitting situation. The fact that NewSVM outperforms BasicSVM on test data is the first indicator of NewSVM's less Overfitting sensitivity in comparison to BasicSVM.

The two other indicators of BasicSVM and NewSVM Overfitting sensitivity are respectively Delta-BasicSVM and Delta-NewSVM (Delta-BasicSVM = F1-Train(BasicSVM) − F1-Test(BasicSVM), Delta-NewSVM = F1-Train(NewSVM) − F1-Test(NewSVM)). These two metrics measure the performance losses that are made respectively by BasicSVM

TABLE II. AVERAGE IMPROVEMENT RATES OF NEW SVM OVER BASIC SVM WITH RESPECT TO KERNELS AND DATASETS

Dataset / Kernel	Gaussian	Anova	Bessel
Average-F1-BasicSVM$_{Restaurant}$	42.67%	69.14%	43.18%
Average-F1-NewSVM$_{Restaurant}$	74.51%	83.59%	86.87%
IR$_{Restaurant}$	74.62%	20.90%	101.18%
Average-F1-BasicSVM$_{Products}$	57.92%	71.07%	49.36%
Average-F1-NewSVM$_{Products}$	74.35%	76.12%	81%
IR$_{Products}$	28.37%	7.10%	64.10%
Average-F1-BasicSVM$_{Laptop}$	44.54%	76.44%	57.87%
Average-F1-NewSVM$_{Laptop}$	79.99%	89.58%	91.92%
IR$_{Laptop}$	79.59%	17.19%	58.84%
Average-IR	60.86%	15.06%	74.71%

and NewSVM between testing and training data. A higher Delta-BasicSVM (Delta-NewSVM) results in a poorer performance on testing data than on training data for BasicSVM (NewSVM). This means that BasicSVM (NewSVM) sensitivity to Overfitting increases. The model that is more sensitive to overfitting is indicated by Delta (Delta = Delta-BasicSVM − Delta-NewSVM). The BasicSVM is more sensitive when Delta is positive; otherwise, the NewSVM is more sensitive. Additionally, BasicSVM becomes more sensitive than NewSVM as Delta increases. Table III shows that for all kernels and for all datasets, all Delta values are positive. This means that the differences between F1-score averages in training data and F1-score averages in test data are smaller for the NewSVM model, and this denotes a lower performance loss between testing and training data, and thus, lower sensitivity to Overfitting.

Therefore, our method aids SVM coping with Overfitting more effectively. Thus, the suggested model is less sensitive than the basic one to Overfitting.

b) Underfitting: To analyze the behavior of the original and new models under Underfitting, and as stated previously in our protocol, the comparative experiments are conducted using the combination $\{C_{min}, \gamma_{min}\}$=$\{0.03125, 0.03125\}$ for Gaussian kernel, and $\{C_{min}, \sigma_{min}\}$ = $\{0.03125, 0.03125\}$ for Anova and Bessel kernels, corresponding to the lowest values of parameters.

Table IV shows the behavior of BasicSVM and NewSVM under Underfitting when both models show poor performance on both testing and training data.

In order to analyze both models sensitivity to Underfitting, we introduce two indicators in Table IV to measure BasicSVM and NewSVM tolerence to Underfitting.

Delta-test (Delta-test = F1-test(NewSVM) − F1-test(BasicSVM)) values are positive in all experiments in Table IV (except for Gaussian kernel on Restaurant dataset). This shows that NewSVM outperforms BasicSVM for all kernels and for all datasets, even in Underfitting situation. The fact that NewSVM outperforms BasicSVM on test data is the first indicator of NewSVM's less Underfitting sensitivity in comparison to BasicSVM.

Delta-train (which is equal to F1-train(NewSVM) − F1-train(BasicSVM)) is the second indicator. Delta-train values are positive in all experiments in Table IV (except for Gaussian kernel on Restaurant dataset). This implies that NewSVM is more performant than BasicSVM on training data. This indicates that NewSVM is more tolerant to Underfitting than BasicSVM.

V. COMPARISON WITH OTHER WORKS

In order to evaluate the effectiveness of the proposed approach, it is compared against various existing methods from the literature. Table V shows a comparison between the traditional and deep learning methods and our suggested method for Implicit Aspect Identification. It is crucial to note that all the works use the same datasets. However, they operate at distinct levels. While W2VLDA [26], and our proposed method (using 3 kernels) work at the algorithmic level by proposing adjustments or additions, the rest of the techniques focus on enhancing the quality of training data by operating at the data level. Schouten et al.'s supervised method [21] is an hybrid method that operates at both data level and algorithmic level.

From Table V, we observe that:

- In the case of Restaurant dataset, despite the difficulty of adjusting the core model that is more challenging and sensitive, our proposed technique (with the three kernels) shows a highly competitive performance level when compared to all works even the ones operating on data level which is less challenging and even the deep learning methods of [19] that are generally reputed for high classification performance.
- In the case of the Products dataset, our three proposed approaches, which operate at the algorithm level without modifying the training data structure, are mostly surpassed by all methods of [18] and [19] that make use of data-level techniques. These techniques enhance the training data by incorporating semantic relations from WN, which should help mitigate the issue of high-class imbalance present in the Products dataset. However, Our technique (with three kernels) outperforms KNN [40] with its three versions, which is an algorithmic-level technique.
- In the case of Laptop dataset, our proposed approach with all kernels outperforms LSTM+WN+Frequency [19] and Att-LSTM+WN+Frequency [19] which are not only deep learning methods that are generally reputed for high classification performance, but also operating on less sensitive and less challenging data level.

TABLE III. F1-score Average Performances of NewSVM and BasicSVM under Overfitting for All Datasets and using Three Kernels

Kernel	Gaussian			Anova			Bessel		
	$\{C_{max}, \gamma_{max}\}$			$\{C_{max}, \sigma_{max}\}$			$\{C_{max}, \sigma_{max}\}$		
	Rest	Prod	Lap	Rest	Prod	Lap	Rest	Prod	Lap
F1-test(BasicSVM)	80.13	76.30	85.45	74.68	77.27	85.50	81.94	77.27	85.60
F1-test(NewSVM)	**86.85**	**78.61**	**90.48**	**85.57**	**77.38**	**91.91**	**87.23**	**80.95**	**92.06**
Delta-test	**6.72**	**2.31**	**5.03**	**10.89**	**0.11**	**6.41**	**5.29**	**3.68**	**6.46**
F1-train(BasicSVM)	100	96.90	99.52	100	96.90	99.52	100	96.90	99.52
F1-train(NewSVM)	100	96.12	98.91	100	96.32	99.38	100	96.90	99.52
Delta-BasicSVM	19.87	20.6	14.07	25.32	19.63	14.02	18.06	19.63	13.92
Delta-NewSVM	**13.15**	**17.51**	**8.43**	**14.43**	**18.94**	**7.47**	**12.77**	**15.95**	**7.46**
Delta	**6.72**	**3.09**	**5.64**	**10.89**	**0.69**	**6.55**	**5.29**	**3.68**	**6.46**

*Rest refers to Restaurant dataset.
*Prod refers to Products dataset.
*Lap refers to Laptop dataset.

TABLE IV. F1-score Average Performances of NewSVM and BasicSVM under Underfitting for All Datasets and using Three Kernels

Kernel	Gaussian			Anova			Bessel		
	$\{C_{min}, \gamma_{min}\}$			$\{C_{min}, \sigma_{min}\}$			$\{C_{min}, \sigma_{min}\}$		
	Rest	Prod	Lap	Rest	Prod	Lap	Rest	Prod	Lap
F1-test(BasicSVM)	15.21	5.07	7.50	15.21	14.97	23.96	15.21	5.07	7.50
F1-train(BasicSVM)	15.23	5.07	7.50	15.23	20.76	24.88	15.23	5.07	7.50
F1-test(NewSVM)	15.21	19.15	26.05	16.98	38.63	50.57	83.35	78.30	88.1
F1-train(NewSVM)	15.23	21.82	26.37	17.48	44.11	53.87	98.47	92.35	98.02
Delta-test	0	**14.08**	**18.55**	**1.77**	**23.66**	**26.61**	**68.14**	**73.23**	**80.6**
Delta-train	0	**16.75**	**18.87**	**2.25**	**23.35**	**28.99**	**83.24**	**87.28**	**90.52**

*Rest refers to Restaurant dataset.
*Prod refers to Products dataset.
*Lap refers to Laptop dataset.

TABLE V. Performances of Selected Traditional and Deep Learning Techniques and Our Proposed Techniques for IAI on Restaurant, Products and Laptop Datasets

Method	**Type**	**F1-score (Restaurant)**	**F1-score (Products)**	**F1-score (Laptop)**
W2VLDA [26]	traditional	72.00%	-	-
Schouten et al. Supervised [21]	traditional	83.80%	-	-
MNB+WN [18]	traditional	77.40%	90.00%	-
BNB+WN [18]	traditional	78.40%	93.30%	-
SVM+WN+frequency [19]	traditional	85.30%	91.80%	-
KNN+WN+frequency [19]	traditional	85.30%	91.80%	-
MNB+WN+frequency [19]	traditional	87.55%	91.80%	-
LSTM+WN+frequency [19]	deep learning	85.20%	89.09%	86.71%
Att-LSTM+WN+frequency [19]	deep learning	87.83%	94.36%	88.26%
KNN with Cosine dist. [40]	traditional	**87.80%**	**74.60%**	-
KNN with Jaccard dist. [40]	traditional	**84.40%**	**74.00%**	-
KNN with Euclidian dist. [40]	traditional	**77.60%**	**72.60%**	-
Proposed SVM with Gaussian	traditional	**88.83%**	**80.21%**	**89.35%**
Proposed SVM with Anova	traditional	**88.54%**	**79%**	**92.84%**
Proposed SVM with Bessel	traditional	**89.81%**	**80.89%**	**93.42%**

VI. Conclusion

In this work, we suggest a method to enhance SVM algorithm to address Implicit Aspect Identification. We provide an improvement for SVM kernel computation to support the IAI task through the use of WordNet semantic relations. For empirical evaluation, experiments are conducted on three datasets of laptop reviews, electronic product reviews, and restaurant reviews, and the effects of our approach on SVM performance are examined and analyzed according to three criteria: (i) kernel function used, (ii) different experimental settings, and (iii) SVM behavior under Overfitting and Underfitting.

The key conclusions of our research can be summarized as follows:

a) Our technique helps SVM improve its performance under different experimental settings and for the three considered kernels and datasets.
b) Our method helps SVM deal with Overfitting and Underfitting more effectively by minimizing their effects on SVM and thereby enhancing its performance.

Even though our approach helps SVM classifier better deal with some of its main issues, it has some limitations at different levels:

- Machine learning model: it only uses one popular eager machine learning model. It would be more interesting to test other types of machine learning models such as lazy or deep learning techniques.
- WordNet semantic relations: it uses only one semantic relation which is "definition relation". It would be also more interesting to explore other semantic relations offered by WordNet like synonyms, antonyms, and their combinations. These relations seem to have significant linguistic importance that may help improve machine learning models to address their critical issues when applied to IAI.
- Datasets used: it uses three datasets that are medium-sized and noise-free that better suit the SVM classification model. We plan to use other less suitable datasets like noisy and large data which present many challenges to the SVM model.

Future work will investigate the use of our method to improve SVM model with non-distance-based kernels and evaluate it under different aspects like dataset size, curse of dimensionality, and noise tolerance. It will also look into considering our approach to address the above-mentioned limitations of our work.

References

[1] Chowdhary, K.R. (2020). Natural Language Processing. In: Fundamentals of Artificial Intelligence. Springer, New Delhi. https://doi.org/10.1007/978-81-322-3972-7_19

[2] A. Tripathy, A. Anand, and S.K. Rath, "Document-level Sentiment Classification using Hybrid Machine Learning Approach," Knowledge Information Systems, vol. 53, no. 3, pp. 805831, 2017.

[3] B. Liu, "Sentiment Analysis and Subjectivity," Handbook of Natural Language Processing, pp. 627-666, 2010.

[4] K. Schouten and F. Frasincar, "Survey on Aspect-Level Sentiment Analysis," IEEE Trans. Knowledge and Data Eng., vol. 28, no. 3, pp. 813-830, 2016.

[5] B. Liu, Minqing Hu, and Junsheng Cheng. Opinion observer: analyzing and comparing opinions on the web. In WWW '05, 2005.

[6] Li Sun, Sheng Li, Jiyun Li, and JuTao Lv. A novel context-based implicit feature extracting method. 2014 International Conference on Data Science and Advanced Analytics (DSAA), pages 420–424, 2014.

[7] Geli Fei, B. Liu, Meichun Hsu, Malú Castellanos, and Riddhiman Ghosh. A dictionary-based approach to identifying aspects implied by adjectives for opinion mining. In COLING, 2012.

[8] Michael E. Lesk. Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In SIGDOC '86, 1986.

[9] Weaver, Warren. 1955. Translation. In William N. Locke and A. Donald Booth, editors, Machine Translation of Languages. John Wiley & Sons, New York, pages 15-23. (Reprint of mimeographed version, 1949.)

[10] Ide, Nancy & Véronis, Jean (1998). Word sense disambiguation: The state of the art. Computational Linguistics. 24. 1-41.

[11] Marco Maru, Simone Conia, Michele Bevilacqua, and Roberto Navigli. 2022. Nibbling at the Hard Core of Word Sense Disambiguation. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 4724–4737, Dublin, Ireland. Association for Computational Linguistics.

[12] Simone Conia and Roberto Navigli. 2021. Framing word sense disambiguation as a multi-label problem for model-agnostic knowledge integration. In Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, pages 3269–3275, Online. Association for Computational Linguistics.

[13] George A. Miller. Wordnet: A lexical database for english. Communications of the ACM, 38:39–41, 1992.

[14] Zhou, J, JX Huang, Q Chen, QV Hu, T Wang and L He (2019). Deep learning for aspect-level sentiment classification: Survey, vision, and challenges. IEEE Access, 7, 78454–78483.

[15] Tubishat, M, N Idris and MAM Abushariah (2018). Implicit aspect extraction in sentiment analysis: Review, taxonomy, opportunities, and open challenges. Information Processing & Management, 54(4), 545–563.

[16] Agarwal, B and N Mittal (2016). Semantic orientation-based approach for sentiment analysis. In Prominent Feature Extraction for Sentiment Analysis, B Agarwal and N Mittal (eds.), pp. 77–88. Springer International Publishing.

[17] Geli Fei, B. Liu, Meichun Hsu, Malú Castellanos, and Riddhiman Ghosh. A dictionary-based approach to identifying aspects implied by adjectives for opinion mining. In COLING, 2012.

[18] Hajar El Hannach and M. Benkhalifa. Using synonym and definition wordnet semantic relations for implicit aspect identification in sentiment analysis. In NISS19, 2019.

[19] Hajar El Hannach and M. Benkhalifa. A new semantic relations-based hybrid approach for implicit aspect identification in sentiment analysis. Journal of Information and Knowledge Management, 19:2050019:1–2050019:37, 2020.

[20] Kim Schouten and Flavius Frasincar. Finding implicit features in consumer reviews for sentiment analysis. In ICWE, 2014.

[21] Kim Schouten, Onne van der Weijde, Flavius Frasincar, and Rommert Dekker. Supervised and unsupervised aspect category detection for sentiment analysis with co-occurrence data. IEEE Transactions on Cybernetics, 48:1263–1275, 2018.

[22] Devi Sri Nandhini, M., Pradeep, G. A Hybrid Co-occurrence and Ranking-based Approach for Detection of Implicit Aspects in Aspect-Based Sentiment Analysis. SN COMPUT. SCI. 1, 128 (2020). https://doi.org/10.1007/s42979-020-00138-7

[23] Rana TA, Cheah YN. Hybrid rule-based approach for aspect extraction and categorization from customer reviews. In: IT in Asia (CITA), 2015 9th international conference on IEEE, p. 1–5, 2015.

[24] M. Sivakumar and U. S. Reddy, "Aspect based sentiment analysis of students opinion using machine learning techniques," 2017 International Conference on Inventive Computing and Informatics (ICICI), 2017, pp. 726-731, doi: 10.1109/ICICI.2017.8365231.

[25] Gupta, Deepak & Ekbal, Asif. (2014). IITP: Supervised Machine Learning for Aspect based Sentiment Analysis. 319-323. 10.3115/v1/S14-2053.

[26] Aitor García-Pablos, Montse Cuadros, and German Rigau. W2vlda: Almost unsupervised systemfor aspect based sentiment analysis. Expert Systems with Applications, 91:127–137, 2018.

[27] Soujanya Poria, Iti Chaturvedi, E. Cambria, and Federica Bisio. Sentic LDA: Improving on LDA with semantic similarity for aspect-based sentiment analysis. 2016 International Joint Conference on Neural Networks (IJCNN), pages 4465–4473, 2016.

[28] Pathik, N.; Shukla, P. Aspect Based Sentiment Analysis of Unlabeled Reviews Using Linguistic Rule Based LDA. Journal of Cases on Information Technology (JCIT) 2022, 24, 1–9.

[29] Soni, P. K., Rambola, R. (2021). Deep Learning, WordNet, and spaCy based Hybrid Method for Detection of Implicit Aspects for Sentiment Analysis. 1–6. https://doi.org/10.1109/conit51480.2021.9498372

[30] Ajeet Ram Pathak, Manjusha Pandey, Siddharth Rautaray, Topic-level sentiment analysis of social media data using deep learning, Applied Soft Computing, Volume 108, 2021, 107440, ISSN 1568-4946, https://doi.org/10.1016/j.asoc.2021.107440.

[31] Ganpat Singh Chauhan, Yogesh Kumar Meena, Dinesh Gopalani, Ravi Nahta, A two-step hybrid unsupervised model with attention mechanism for aspect extraction, Expert Systems with Applications, Volume 161, 2020, 113673, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2020.113673.

[32] Vapnik VN (1995) The nature of statistical learning theory. Springer Verlag, New York.

[33] Ivan Cruz, Alexander Gelbukh, and Grigori Sidorov. Implicit aspect indicator extraction for aspect-based opinion mining. International Journal of Computational Linguistics and Applications, 5:135–152, 2014.

[34] Gayatree Ganu, Noémie Elhadad, and Amélie Marian. Beyond the stars: Improving rating predictions using review text content. In WebDB, 2009.

[35] Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. SemEval-2014 task 4: Aspect based sentiment analysis. In Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), pages 27–35, Dublin, Ireland, August 2014. Association for Computational Linguistics.

[36] Minqing Hu and Bing Liu. Mining and summarizing customer reviews. Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, 2004.

[37] Maria Pontiki, Dimitrios Galanis, Harris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. SemEval-2015 Task 12: Aspect Based Sentiment Analysis. In Proceedings of the 9th International Workshop on Semantic Evaluation, Denver, Colorado.

[38] Pontiki, Maria & Galanis, Dimitris & Papageorgiou, Haris & Androutsopoulos, Ion & Manandhar, Suresh & AL-Smadi, Mohammad & Al-Ayyoub, Mahmoud & Zhao, Yanyan & Qin, Bing & de clercq, Orphee & Hoste, Véronique & Apidianaki, Marianna & Tannier, Xavier & Loukachevitch, Natalia & Kotelnikov, Evgeny & Bel, Nuria & Zafra, Salud María & Eryiğit, Gülşen. (2016). SemEval-2016 Task 5: Aspect Based Sentiment Analysis. 19-30. 10.18653/v1/S16-1002.

[39] B.C. Vickery. Reviews : van rijsbergen, c. j. information retrieval. 2nd edn. london, butterworths,i978. 208pp. Journal of librarianship, 11(3):237–237, 1979.

[40] Benarafa Halima, Benkhalifa Mohammed & Akhloufi Moulay, WordNet Semantic Relations Based Enhancement of KNN Model for Implicit Aspect Identification in Sentiment Analysis. Int J Comput Intell Syst 16, 3 (2023). https://doi.org/10.1007/s44196-022-00164-8