I. J. Computer Network and Information Security, **2022, 5, 28-46**
Published Online on October 8, 2022 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijcnis.2022.05.03

Sentiment Analysis CSAM Model to Discover Pertinent Conversations in Twitter Microblogs

Imen Fadhli
MARS Research Laboratory LR17ES05, University of Sousse, ISITCom, 4011 Hammam Sousse, Tunisia
E-mail: imen.fdh@gmail.com

Lobna Hlaoua
MARS Research Laboratory LR17ES05, University of Sousse, Tunisia
E-mail: lobna1511@yahoo.fr

Mohamed Nazih Omri
MARS Research Laboratory LR17ES05, University of Sousse, Tunisia
E-mail: mohamednazih.omri@eniso.u-sousse.tn

Received: 21 March 2022; Revised: 25 June 2022; Accepted: 19 August 2022; Published: 08 October 2022

Abstract: In recent years, the most exploited sources of information such as Facebook, Instagram, LinkedIn and Twitter have been considered to be the main sources of misinformation. The presence of false information in these social networks has a very negative impact on the opinions and the way of thinking of Internet users. To solve this problem of misinformation, several techniques have been used and the most popular is the sentiment analysis. This technique, which consists in exploring opinions on corpora of texts, has become an essential topic in this field. In this article, we propose a new approach, called Conversational Sentiment Analysis Model (CSAM), allowing, from a text written on a subject through messages exchanged between different users, called a conversation, to find the passages describing feelings, emotions, opinions and attitudes. This approach is based on: (i) the conditional probability in order to analyse sentiments of different conversation items in Twitter microblog, which are characterized by small sizes, the presence of emoticons and emojis, (ii) the aggregation of conversation items using the uncertainty theory to evaluate the general sentiment of conversation. We conducted a series of experiments based on the standard Semeval2019 datasets, using three standard and different packages, namely a library for sentiment analysis TextBlob, a dictionary, a sentiment reasoner Flair and an integration-based framework for the Vader NLP task. We evaluated our model with two dataset SemEval 2019 and ScenarioSA, the analysis of the results, which we obtained at the end of this experimental study, confirms the feasibility of our model as well as its performance in terms of precision, recall and F-measurement.

Index Terms: Conversational, sentiment analysis, word embedding, belief function, conditional probability.

1. Introduction

1.1. *Context and Issue*

Nowadays, with the emergence of web 2.0 and the different types of social platforms (Facebook, Twitter, etc.), a certain number of research works in the field of reducing the distance between the information retrieval (IR) and online social networks (OSN) have been proposed. The operation of closing the gap between IR and OSN is mainly done through an enrichment of the IR process with information from social networks and is called social information retrieval (SIR). The latter is used to find information that satisfies the user's needs by integrating social relationship with social content. Twitter is the most popular social network characterized by its small size that helps and guarantees easy communication of information, activities, thoughts and opinions between users. This microblogging allows them to interact and respond to messages by creating a conversation to express opinions on products, services and events. The work presented in [1] shows the importance of the opinion expressed by Twitter users on different products/services, restaurants, hotels, etc. Sentiment analysis in Twitter [1] has opened up new challenges. The feeling can be positive negative or neutral (information). Recently, it has become an issue of great importance due to the limited length of tweets (280 characters maximum). This limitation has caused frequent use of abbreviations, with informal language, to express opinions. One of the biggest challenges is sentiment classification with short and noisy tweets. In addition, due to the limited number of terms that appear only once and the language used in twitter, sentiment analysis has become a crucial issue. In this perspective, several works [2-4] have focused on the conversation on Twitter in order to improve the quality of tweets and minimize the limit of brevity on Twitter.

1.2. Research questions

The purpose of the research presented in this article is to answer a few research questions.

- RQ1: How do we calculate and analyse sentiment conversation with the presence of tweets and replies?
- RQ2: How do the main features contribute in sentiment classification?
- RQ3: How do we aggregate positive, negative and neutral conversation items in order to determine the general sentiment of conversation?

1.3. Contributions

In order to solve the problem of reduced sentiment conversations, we aggregate tweets and replies to obtain sentiment conversation. The presence of uncontrolled and uncertain information in the microblog caused by the short length of tweets generates many difficulties of analyzing. For example, the presence of a positive tweet does not confirm a positive conversation. Besides the presence of negative, neutral and positive tweets in the same conversation introduces the uncertainty processing paradigm. Since 2004, many methods for dealing with uncertainty have been developed in the field of information retrieval. We quote, for example, the work presented in [5] and [6]. The work presented in [5] has shown the importance of uncertainty theory in information retrieval compared to probability theory. In this perspective, we propose a method for dealing with uncertainties in microblogs. The major contributions of the work presented in this paper are summarized as follows. Firstly, we calculate the polarity of each tweet in the conversation. Secondly, we propose an aggregation method, to obtain a new aggregated sentiment conversation. Finally, we evaluate our model with two datasets SemEval2019 and ScenarioSA.

1.4. Paper organization

The remainder of this article is organized as follows: section 2 presents a summary of the related work we have done in this area. In section 3, we present the problem formulation based on the conditional probability and the belief function. Section 4 gives the main motivations of the proposed model. In section 5, we detail the proposed model and we present the different algorithms that we have developed in this work. Section 6 presents the experimental study, the obtained results as well as a general discussion. Section 7 concludes this work and gives some perspectives.

2. Related Works

Sentiment analysis and emotion recognition have been extensively studied in [7,8]. They represent a confused expression extensively studied in literature. As defined in oxford dictionary, emotion is a strong feeling such as love, fear, or anger; the part of a person’s character that consists of feelings[1] although sentiments represent a more general idea compared with emotion. Sentiment polarity could be ’positive’, ’negative’ and ’neutral’; for example, if a user says "I am happy" this mean that the emotion of the user is "happy" and the sentiment behind is "positive" [8]. Some authors [9] propose a multimodal framework for conversational emotion recognition. To determine accurate emotion from a text, several methods have been used [8]. Since human emotions are so complex, determining the correct emotion from text presents a number of difficulties that must be addressed and solved. When numerous emotions are expressed in a single tweet, analyzing emotions becomes more challenging. Moreover, implicit emotion in a text makes emotion detection nearly impossible. The term "emotion" refers to people’s feelings or reactions to a specific event. ’Happy,’ ’Sad,’ ’Angry,’ and ’Fear’ are only a few instances of emotions. It is a textual analysis process that extracts and identifies subjective information by calculating the polarity of a text, sentence or feature. The sentiment found in comments such as feedback on products, services or those of political candidates can provide important indicators for different purposes [7]. With social media, especially in the Twitter microblog, users express their opinions using different emoticons such as: ":)"; ":(";":|". This type of sentiment can be classified into 3 categories: positive, negative and neutral [7]. The sentiment analysis process was applied at 3 levels: (i) document level, used to determine the sentiment of the document, e.g., positive, negative or neutral. (ii) The sentence level, which considers each sentence as a separate unit and assumes that this sentence contains only one opinion. This level distinguishes between objective and subjective sentences: the objective sentence expresses factual information and the subjective sentence expresses subjective opinions. (iii) The level of the entity determines the polarity of the aspect or entity (positive, negative and neutral). In the work presented by [10], the authors analyze the sentiment of tweets using different techniques. They use the naive Bayes algorithm for SA tweets along with linguistic methods such as unigram, bigram, trigram, and post tagger.

Sentiment analysis research has primarily focused on two steps: a first step, called sentiment detection, to identify objective/subjective tweets, and the second step is sentiment analysis/calculation to calculate the polarity of subjective tweets. In order to accomplish these two phases, several features extracted from conversations content and from the social context have been used. The latter have been extensively studied in the literature.

[1] https://www.oxfordlearnersdictionaries.com/

2.1. *Features Extraction*

There are many sentiments detection and classification features in Twitter such as post tags, stylistic, emoticons and social content features defined as follows:

- Pos Tag Features: This feature is based on the number of verbs, pronouns, adjectives, nouns, adverbs, etc. In order to determine the sentiment of tweets, the approach presented by [11] proceeds as follows: each term, in a tweet, is marked to identify nouns, adjectives, interjections, verbs, adverbs, etc. Similarly, the work proposed by [12] considers the number of adverbs, nouns, adjectives and verbs as characteristics that will be exploited later in the detection process.
- Stylistic features of content: This feature is important to detect sentiments in microblog posts. Thus, punctuation has also been explored in the [10] literature. As an example of stylistic characteristics of the content, we cite among others the number of words, capitalization and punctuation.
- Characteristics of emoticons: Emoticons are used to express the users' sentiments, so their polarities can be considered a relevant characteristic for sentiment analysis on Twitter [13,14]. Several researches [10,13,14] have looked at the calculation of emoticons in tweets, such as the number of positive and negative emoticons.
- Social content features: These are related to the social features of the microblog, such as the use of repeated letters and the Internet slang. Additionally, Twitter features, such as user mentions (followed by the special character @), retweets (denoted by RT), URLs, and hashtags (followed by the special character #) have also been explored in the literature. In the approach proposed by [15], the authors perform an analysis of sentiment tweets. They consider the hashtags as well as the users' identifier to understand the meaning of their messages.

Many other studies have focused on the classification and calculation of the polarity of feelings (positive, negative and neutral) after the process of detecting subjective and/or objective tweets.

2.2. *Sentiment Detection*

In order to detect the sentiments of browsers within the microblog, many approaches have been developed. We cite the main ones:

- Lexical approaches: in this type of approach, the notions of co-occurrence and keywords in a sentence [16,17] are mainly considered.
- Machine learning approaches: these approaches use supervised and unsupervised classifier methods [4,3].
- Hybrid approaches: this category of approaches, which consists of combining those based on the lexicon and those based on machine learning, uses automatic language processing tools, for a first phase of corpus preprocessing. Then, the supervised learning techniques and/or automated learning tools are used to classify the text to build the opinion dictionary or to create classifiers [18].

2.3. *Classification Sentiment Analysis*

A. *Sentiment analysis tools*

There are two types of classifications: (i) a binary classification which considers two classes, positive class and negative class. (ii) a multi-class classification which considers five classes: strongly positive, positive, neutral, negative and strongly negative. Most of the research in this context has opted for binary classification.

Lexical approaches: For sentiment classification, several solutions, such as manual annotation, dictionaries and corpora, have been used. Each of these methods are used as follows:

- The manual method: this method consists of a manual annotation of feelings. Generally, this annotation is used with a dictionary or a corpus in order to test and evaluate the result.
- The dictionary-based method: in this method, a dictionary extracts different sentiment words with a given score followed by synonyms and antonyms. Several approaches use SentiWordNet which is an opinion lexicon derived from the WordNet database where each term is associated with numerical scores indicating positive and negative sentiment information.
- The corpus-based method: This method uses a set of dictionaries containing additional information such as tags, parts of speech, etc.

The approach proposed in [19] uses lexical tools to analyze tweets sentiment in real time of the public reaction towards the announcement of the Indian Union Budget 2020. In this approach, the authors calculate sentiment analysis of tweets with Textblob. They calculate the polarity score: -1 indicates negative tweets, 0 neutral and 1 for positive tweets.

Machine learning approaches: Other sentiment ranking techniques such as Naive Bayes decision tree and SVM have also been used. In a first phase, these methods extract features from sentences, for example, negation, punctuation, emoticons, the presence of frequent n-grams. The approach proposed in [20] proposes to use naive bayes with Textblob and tweepy as a python library to classify tweets into three categories: positive, negative or neutral. Similarly, in [21] different machine learning classifier algorithms were used to analyze human sentiment using twitter data. The model proposed in [3] classifies conversations using logistic regression. They compare different features based on semantic representation using word integration and emotional integration with emoticon and emoji information integration

Hybrid approaches: The first step in the hybrid approach is the automatic annotation of the corpus with lexicon-based tools and then the use of machine learning tools to train the classifier on the corpus. In the work proposed by [18], the authors use a corpus of tweets with different queries firstly. Secondly, they annotate a corpus with a lexical approach using POS and an opinion word. The third phase consists to predict corpus feelings with SVM method. This combination considerably improves the result in terms of precision.

2.4. Tweet vs. Conversation

Most of the work presented in the literature are limited to analyze tweets sentiments and neglect sentiment conversation. However, the study of sentiment conversation is rarely reported. For example, the model proposed in [4] presents a framework for predicting the veracity of rumors on Twitter. [4], propose a new approach to learn tweet representations by aggregating their neighbor reply. They suppose that nearer neighbors in a tweet's conversation thread are more instructive than farther neighbors since their replying relationships are closer, and their stance expressions can help categorize the attitude of the center tweet. The approach presented in [22] proposes grouping tweets by topic in the same user-to-user conversation. While the model presented in the reference [22] consolidates tweets and their replies into a single document and considers users as co-authors. Analyze sentiment conversation is a challenge that has not yet been extensively surveyed. Several works consider that the emotion of each tweet turns to be independent. On the contrary, we focus on sentiment conversation instead of a single tweet. This goal is challenging for three reasons: (i) there is a high correlation between tweet and reply in conversation, sentiment conversation depends not only on the initial tweet but also on the reply; (ii) the sentiment of each user reply is influenced by other users replies; (iii) there could be factual or neutral tweets which is different from positive or negative tweets in which users express their opinions.

2.5. Uncertainty Theory in IR

Several approaches have explored the problem of information imprecision and uncertainty, in several fields, especially in information retrieval. For example, [23] presents a new approach to select the relevant comments in a collaborative framework. Another model proposed in [24] in which the authors propose a new approach based on the probability method in order to solve the retrieval of information from the deep web. The model presented in [12] solve the problem of uncertain fuzzy data by exploiting Word2PLTS for textual representation. Since the type of conversation, we consider in our work is in the form of a set of tweets, we assume that we are in the presence of several sources of information where each tweet alone represents a source of information. On Twitter, the uncertainty over noisy information makes the problem even more complex. This complexity has encouraged us to propose a new method of aggregation based on the belief function which will be used to detect the existence or not of polarity in the text (positive, negative or neutral) processed and then classify these emotions.

3. Motivation of the Proposed Model

The main objective of information retrieval in the microblog is to identify the relevant answer versus a query Q, from a set of messages exchanged between users. Several studies have shown the importance of this set of messages despite the limited length of the tweets, the presence of noise and informal language [25, 22]. In this article, we focus on Twitter conversations rather than single tweets. Inspired by the approach proposed by [25], which defines a conversation as a set of tweets shared between bloggers using the "rep" function of Twitter, we can thus publish, retweet and respond to tweets. Table 1 represents 5 conversations categorized as (happy, angry, sad, and others), where, for example, the initial tweet is shared by blogger 1, replied by blogger 2, and rep 3 is the response from blogger 1 to blogger 2.

The process of analyzing emotions can be considered a difficult problem in textual conversation given the limited length (280 characters) of tweets, due to the presence of unstructured and informal language. We distinguish three cases:

- The first case is positive tweet and positive reply, which produces positive conversation.
- The second case corresponds to different polarities which cause a problem in detecting the polarity of the conversation. For example, positive initial tweet with negative or neutral replies.
- The third case is that of a negative tweet and a negative reply. This case gives a negative conversation, and can also be a neutral or positive conversation.

Table 1. Example of conversation

Initial Tweet	Rep2	Rep3	Sentiment
Money money and lots of money 😍😍	I need to get it tailored but I'm in love with it 😍	😁😁	happy
My gf left ne	Get over it. Go out with someone else.	Me*	Sad
ehat is hehe	Haha is more like: Hehe is more of	;)what*	Others
I hate it too	Guess what, I don't.	Even I don't	Angry
Not always	What about yesterday	Do u know what 69 is?	Others

Motivated by the importance of the conversation instead of a tweet in the detection of feeling, we proceed to an aggregation of tweets without an arithmetic method. We propose a new aggregation method to combine the sentiment of tweets in order to detect the sentiment of conversation. We calculate sentiment conversation based on sentiment tweets and their replies. We therefore propose to consider the conditional probability in order to aggregate the tweets with the score of the responses. In this context, we consider that each tweet is a source of information, we calculate the polarity score and we aggregate it with a conditional probability. A first challenge is the difficulty in understanding sentiment conversation with the presence of the ambiguity. The second challenge is the presence of uncertainty and noisy information on Twitter which makes the task of identification even more complex. In order to overcome these challenges, we proposed to integrate the notion of belief function to detect this type of sentiment conversation. In this perspective, we have hypothesized that a significant correlation exists in the conversation, especially between tweets and replies.

4. Problem Formulation

In this work, we have hypothesized that a significant correlation exists in conversation, especially between tweets and replies. Therefore, Twitter conversation should be presented as follows: first bloggers express their opinion by sharing tweets and then users respond to tweets, by creating a conversation. Formally, let C^T be the set of tweets T_i and R_i the set of corresponding replies, with $1 \leq i \leq N$.

4.1. *Conversation Representation*

We model the set of conversations C^T by a combination of tweets, denoted $C^T = \{T_i, R_i +1, R_i +2..., R_N\}$. Ti represents the initial tweet in C^T. R_i is the ith tweet reply, in the conversation, which can be either positive, negative or neutral. The two events are not independent.

4.2. *Conditional Probability of R given T*

Given the previous notation that R represents a response and T the corresponding tweet and given that these two events are considered dependent, then the conditional probability of the response R, relative to a tweet T, is defined by [26] as follows:

$$P(R|T) = \frac{\mathrm{P(R \cap T)}}{\mathrm{P(T)}} \tag{1}$$

with:

- P (R | T): represents the conditional probability of reply R given a tweet T.
- P (R ∩ T): represents the probability of the intersection between tweet T and reply R.
- P(T): the probability of tweet T.

Otherwise, the conditional probability can be calculated using Bayesian theory. Mathematically, the Bayes' theorem can be denoted in the following way:

$$P(R|T) = \frac{\mathrm{P(T|R) * P(R)}}{\mathrm{P(T)}} \tag{2}$$

Where:
P(R), P(T) represents the probability of reply and tweet respectively,
P (T | R): represents the conditional probability of reply, given that the tweet has occurred.

The conditional probability is applied to the calculation of events that are independent or mutually exclusive [27]. We consider that two events are independent if the probability of reply cannot influence the probability of tweet [28]:

$$P(R|T) = P(R),\ P(T|R) = P(T).$$

Two events are considered mutually exclusive when these events cannot occur simultaneously. The conditional probability is always zero: $P(R|T) = 0$ and $P(T|R) = 0$.

4.3. Belief of conversation C

let Ω, the space defined by Shafer [29], be the set of conversation, m: is an application from $2^{\Omega} \rightarrow [0,1]$ which represents a belief mass of each element of the power set Ω if $m(\emptyset) = 0$ and $\sum_{C\subseteq\Omega} m(C) = 1$. Then, we define Bel(R) by:

$$Bel(C^T) = \sum_{i=1, c\subseteq C}^{N} m(ci) = 1 \tag{3}$$

is called the belief function corresponding to the basic belief assignment m. It is well known (see [29] for a proof) that m can be retrieved from Bel via:

$$m(ci) = \sum_{i=1, R,T\subseteq C}^{N} m(R_i) + m(T_i) \tag{4}$$

where: $m(T_i)$, $m(R_i)$ represents the conditional probability computed by combining polarity with similarity detailed in the next section.

5. Proposed Conversational Sentiment Analysis Model

5.1. Proposed Model

A positive, negative or neutral conflict score makes the process of conversation sentiment detection more difficult and arithmetic average cannot represent reality. Therefore, we have proposed the integration of the belief function. The different stages of the model detection process that we have proposed are illustrated in Figure 1. The first stage of the process is the data set extraction phase. Then, the second step is the pre-processing phase, which generates a correction of spelling and punctuation errors in order to remove the noise from the dataset. The third step is sentiment analysis. In this step, we calculate the polarity tweets (Positive, Negative and Neutral), then we proceed to aggregate the tweets polarity to obtain a conversation score which can be Positive, Negative or Neutral.

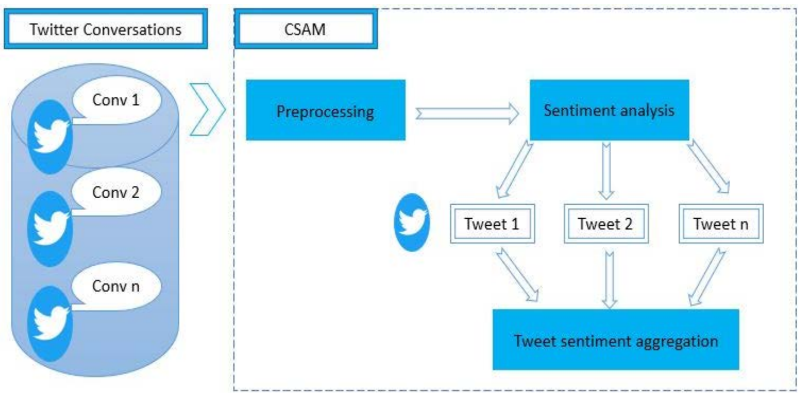

Fig.1. Conversation sentiment analysis model CSAM

Each phase of the proposed model will be described in more details in the following sections.

5.2. Preprocessing Phase

Tweets are generally characterized by their short lengths (280 words) and can be poorly structured and noisy due to the presence of incomplete abbreviations and expressions or due to irregular grammar and acronym. These issues negatively impact the performance of the sentiment classification tweet. In order to reduce noise in tweets, the following preprocessing steps are applied:

- Numbers removal step: usually the number does not contain sentiment. So, when measuring sentiment, the number is useless and in order to improve the content of the tweets, the numbers are removed from these tweets.
- Stopword removal step: In this step stopwords such as "and", "that", "a" will be removed.
- *Tokenization step: This task is performed using TweetNLP.*

5.3. Sentiment Analysis

In this step, we calculate the sentiment score of tweets with the model given by equation 3. We assume that a positive response in a conversation does not necessarily confirm that the polarity of the conversation is positive. Moreover, the sentiment response of score R_i depends on R_{i-1}. Therefore, we consider the tweets and the sentiment of the replies. We compute the conditional probability of a given tweet in response by integration polarity with the similarity measure which will be detailed in the next section.

A. Conversation polarity

The main focus of our model is on the polarity of conversation. Polarity is the score given after analyzing tweets sentiment 1 denoting a positive tweet, -1 denoting a negative tweet and 0 for neutral tweet. We calculate polarity of each tweet in conversation separately. As [3] and [2] show the independence between tweet and reply, our goal is to aggregate tweets with replies in order to calculate the sentiment of all conversation. We represent tweet and reply polarities by Pol(T) and Pol(R) respectively. The challenging problem is the presence of positive, negative and neutral tweets in the same conversation. So how can we deduct the polarity of the conversation? How does it seem? Does it analyse as positive, negative or neutral? The solution is to apply an appropriate aggregation method which guarantees the sentiment polarity of all conversation. We start by calculating the polarity of single tweet with 3 models described in the next section. Next, we assume that the polarity of each tweet depends on the set of replies as shown in figure 2. For example: R5 and R4 are two replies to R1. To determine the polarity of R1, first of all we calculate the polarity of R4 and R5 independently. Next, we calculate the dependency between R4 and R5 by combining polarity with similarity to obtain the sentiment of R1. This step will be detailed in the next section. Formally, if $Pol(R4) \cup Pol(R5) > 0 \Rightarrow Pol(R1)$ depends on the similarity (R4, R1) and similarity (R5, R1) which will be described in the next section.

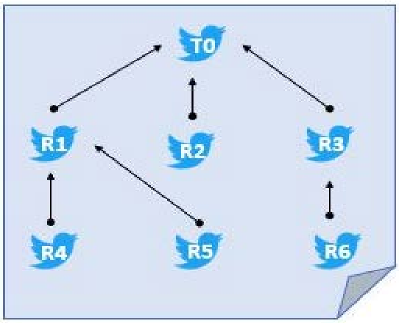

Fig.2. Conversation polarity

B. Polarity calculation

In the step of calculating the polarity score, we use 3 sentiment analysis models with Twitter conversations [30].
Sentiment classification tools:

- Textblob [31]: Textblob is a rule-based sentiment analysis library that focuses on lexical content and integrates the WordNet corpus for sentiment analysis. The calculated polarity score varies between [-1.1] (-1 negative sentiment, 1 positive sentiment).
- Vader [32]: Vader is a valence-sensitive dictionary and represents a tool for sentiment analysis and sentiment reasoning. It is lexicon-based and specially used to detect sentiment in social media. Similar to TextBlob, Vader falls under the category of lexical tools.
- Flair [33]: Flair is an integration-based framework for NLP tasks. This model calculates positive and negative score sentiment between -1 and 1. Flair is a machine learning tool based on Named Entity Recognition (NER) and part-of-speech tagging (POS).

Emoticon features: Several works, such as [34, 10, 35], attempted to improve the importance of emoji and emoticons for detecting sentiment. In our work, we convert emojis and emoticons ":) :(:| " to textual data for example ":)" will be converted to "happy face smiley". To do this, we use the python library to convert these emoji and emoticons. Since, by assumption, we have assumed that a positive tweet does not only confirm that the conversation is positive. Due to the limited appearance of the terms and the short length of the tweets, we add the similarity between the tweets and the replies in order to calculate the sentiment of the conversation. We then developed new methods to calculate sentiment conversation in which we aggregate tweets with responses polarity. We also assumed that sentiment conversation depends on sentiment tweets with their response, which allowed us to incorporate conditional probability to aggregate tweets with response score.

$$P(R|T) = \frac{\mathrm{Pol}(T_0) * \mathrm{Sim}(R_i, T_0)}{\mathrm{Pol}(R_i)} \quad (5)$$

With: Sim (R_i, T_0) = Sim (T_0, R_i)

C. Similarity degree

After the conversation pre-processing phase, we compute the polarity of the tweets using 3 packages detailed in the previous section. Then we encode word-to-vector by the word embedding technique. The final phase is to calculate the similarity between the tweet and the reply, as shown in Figure 3.

Features representation: In order to analyze sentiment in Twitter, different methods are used: the N-gram method, the Meta-level method, and the Word embedding method.

- N-gram method: this method allows a representation of the model for natural language analysis and sentiment analysis. Several studies have shown the performance of n-grams for sentiment classification in Twitter [16, 36].
- Meta level method: this method has been categorized by Cavallo and Plastino [37], and is classified into 5 classes: microblog (retweet, hashtag, etc.), post tag (verb/noun number, etc.), surface (number of marks, number of words. . .), emoticon (positive/negative, etc.), based on the lexicon (number of positive or negative words) [10, 18, 38]
- Word Embedding Method: This method helps improve the performance of the result and several researches have shown the importance of word embedding features.

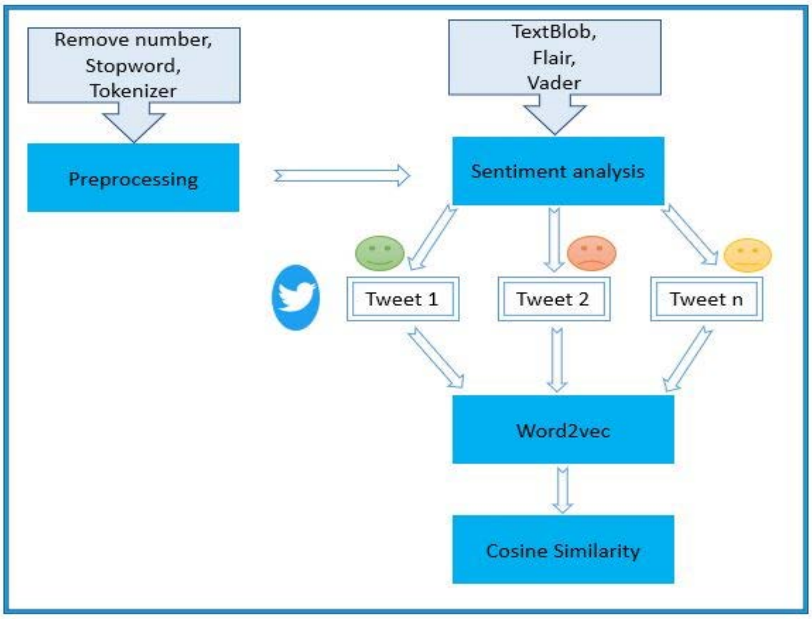

Fig.3. CSAM: Different steps of sentiment analysis based on aggregation model

In this article, we use the word2vec [39] technique to encode our textual data and calculate the similarity between tokens in tweets.

The cosine similarity: Cosine similarity is a measure of the degree of similarity between two vectors given by the angle between these two vectors. This measure is the most commonly used and is based on the co-occurrence of terms by fixing the dimension on the different concepts. It is defined as follows:

$$\boldsymbol{Sim_{cos}}(\mathbf{twe}, \mathbf{rep}) = \frac{\mathbf{w2vec(T)} * \mathbf{w2vec(R)}}{||\mathbf{w2vec(T)}|| * ||\mathbf{w2vec(R)}||} \quad (6)$$

Where W2vec(T), w2vec(R) are two vectors calculated by word2vec techniques. Cos (T, R) is within the range [-1, 1], the value -1 indicates that the vectors are opposite, 0 for independent vectors and 1 for similar.

5.4. Tweet Sentiment Aggregation

Abbreviations and informal languages, used by bloggers, characterize tweets and make them incomplete and imperfect. These tweets often contain error because of the uncertainties and inaccuracies that characterize them. In addition to the presence of ambiguities caused by the reduced sizes of the tweets. Several related works, such as [40] and [25] have proved the importance of exploiting conversations instead of treating tweets separately. In an attempt to identify the meaning of these tweets, we have developed module to aggregate all conversation's elements (original tweet and replies).

A. Belief function

Since we are in the presence of several sources of imperfect information, where each tweet is a source of information, several methods in the literature have proposed to merge these sources of imperfect information in order to be able to generate comprehensive and complete information. The most well-known main method of information fusion is that of the theory of evidence [41]. In our work, we assume that if we have a positive response in the conversation, it does not confirm that the conversation is positive and the same for negative tweets. Indeed, our solution consists in calculating a degree of belief using the theory of evidence, allowing us to identify the meaning of the ideal sentimental conversation. This degree of belief, denoted by "Bel", measures the total belief attributed to a sentimental conversation. $\forall C \in 2^{\Omega}$, this degree of belief is defined as follows:

$$m(c) = \sum_{i=1, c \subseteq C}^{N} m(c_i) \tag{7}$$

Bel is the credibility function associated with the mass of the corresponding hypothesis. In our case, $m(c_i)$ is a conditional probability computed by combining polarity with similarity

$$Bel(C^T) = \sum_{i=1}^{N} m(c_i) = \sum_{i=1}^{N} Prob_i(R_i | T_i) \tag{8}$$

B. Mass distribution

The structure of the considered datasets is composed of pairs formed by tweets and their replies $\Omega = T_0, R_1, R_2, RN$. Belief theory, given by Shafer [29], represents a generalization of Bayesian theory. Dempster defines a confidence interval to model the uncertainty, composed of two probability values: an upper probability and a lower one. This interval, on the belief function, is known as the belief mass denoted by m. In this context, we propose to calculate a conditional probability to aggregate the scores of tweets [27, 28]. Conditional probability is the probability that an event will occur in relation to one or more other events. Using this probability, we calculate the base probability assignment called BPA between a tweet and the corresponding reply. We assume that the sentiment response depends on the sentiment tweet. In this case we note:

$$\mathrm{Prob(T|R)} = \frac{\mathrm{Prob(T \cap R)}}{\mathrm{Prob(R)}} \tag{9}$$

where:

$$\mathbf{Prob(T \cap R) = Pol(T) * Sim_{cos}(T, R)} \tag{10}$$

and

$$\mathbf{Prob(R) = Pol(R)} \tag{11}$$

In our case we calculate belief mass with multiply conditional probability by similarity between tweet and response.

$$\mathrm{Prob(T|R)} = \frac{\mathrm{Pol(T)*Sim_{cos(T,R)}}}{\mathrm{Pol(R)}} \tag{12}$$

where: T represents a tweet and R is a reply.

Pol: score polarity calculated by sentiment analysis model Textblob. We suppose that a positive tweet does not necessarily confirm that the conversation is positive so we need to aggregate tweets sentiment.

C. Proposed CSAM algorithm

As a first step, we first execute all the tweets and retrieve the corresponding replies. In a second step, we calculate the probability score for each answer. And in a third step, we proceed to a correspondence of these scores with the probability score of the initial tweets in order to obtain the probability score of all the conversations.

```
Algorithm 1: Proposed CSAM algorithm
Input: C^T = {T_0, R_1, R_2}
Output: Sentiment Conversation
begin
    for i in C^T do
        /* Calculation polarity with Textblob Model
        Pol(T_i) ← Calculation Polarity (T_i)
    for i in C^T do
        for j in range (i+1,C^T ) do
            /* Calculation similarity
            Sim_cos(T_i, R_j) ← Cosine similarity eq(9) (T_i)
    for i in C^T do
        for j in range (i+1,C^T ) do
            /* conditional probability score
            Function ConditionalProbability
            Pol(T_i), Pol(R_j), Sim_cos(T_i, R_j)
                if Pol(T_i);
                & Pol(R_i) then
                    Prob ← Pol(R_j)* Sim_cos(T_i,R_j)/ Pol(T_i) (eq 16)
                else
                    Prob ← Pol(R_j)* Sim_cos(T_i,R_j)
            return Prob
    for i in C^T do
        for j in range (i+1,C^T ) do
            /* Credibility score
            Function Aggregationrely
            (Prob_ij, Pol(T_i), Pol(R_j))
                m(c)←Pol(T_i)+Pol(R_j)+Prob_ij + m(conv)
                Bel(C_T) ← m(c)
            return Bel_ij
End
```

6. Experimental Study and Results Analysis

6.1. Used Data Collection

To evaluate our proposed method, we use two benchmark datasets SemEval2019 and ScenarioSA. SemEval-2019 task 3 EmoContext: Contextual Emotion Detection in Text dataset includes 2755 conversation composed by (id, turn1, turn2, turn3, label). SemEval-2019 contain:

- id: Column to identify each conversation, contains a unique number.
- Turn1: Is the first tweet in conversation written by User1
- Turn2: represents a reply to Turn1 (the first tweet) in conversation written by User2.
- Turn3: Is a reply to Turn2 written by User1.
- Label: label value can be: "happy", "sad", "angry" and "others". This column contains human judgment of conversation sentiment.

ScenarioSA large scale conversational dataset, for conversational sentiment analysis created by [42]. Figure 4 present an example of conversation in ScenarioSA where A and B are two users the final sentiment label is the aggregation sentiment of each user. We are interested to analyze sentiment conversation instead of user sentiment. Due to the lack of conversation dataset which consist to aggregate multi-turn conversation we extract from ScenarioSA conversations with one final sentiment for example when A and B express the same sentiment (A and B are positive, both negative or neutral). We extract 2822 conversation from ScenarioSA labeled positive=1, negative=-1 and neutral=0, we consider positive conversations as happy, negative as angry and neutral as others.

Fig.4. An example of conversation in ScenarioSA between two users A and B, where 1=positive, 1=negative, 0=neutral. [42]

Table 2. Characteristics of used collection data

Dataset	Happy	Angry	Others
SemEval2019	360	1021	1374
ScenarioSA	1399	627	796

6.2. Evaluation Measures

In this subsection, we present the different standard measures that we have used to evaluate the proposed model. These are precision, recall, F-measure, accuracy and G-mean which we will use to evaluate the performance of our algorithms. These measures are defined based on the following confusion matrix (see table 3).

Table 3. Confusion matrix

Conversation	Labeled Conversation	
	Relevant	Irrelevant
Relevant	N_{RC}	N_{RNRC}
Irrelevant	N_{IRRC}	N_{IRC}

We compute the four following performance indices:

- N_{RC} is the number of relevant classified conversation retrieved after applying our model.
- N_{RNRC} is the number of relevant classified conversation not retrieved by our model.
- N_{IRC} is the number of irrelevant classified conversation not retrieved with our model.
- N_{IRRC} is the number of irrelevant classified conversation retrieved by our model.

The Precision measurement Accuracy is calculated as the ratio of the number of relevant conversations returned by our model. Any relevant conversation that is not returned constitutes what is called “the noise”, and precision opposes this noise. When this accuracy value is high, it means that the conversation, classified as the most relevant, exists and that the model will be considered "accurate".

$$\text{Precision} = \frac{N_{RC}}{N_{RC}+N_{RNRC}} \tag{13}$$

The Recall measurement The Recall coefficient is calculated as the ratio of the number of relevant conversations found by the total number of relevant conversations. When this coefficient is high then the recall rate is high. Conversely, if the system presents many relevant classified conversations but they will not be returned, by the proposed model, we speak of “silence” which opposes the recall.

$$\text{RECALL} = \frac{N_{RC}}{N_{RC}+N_{RNRC}} \tag{14}$$

The F-measure measurement the F-measure is the average of the precision P and the recall R to express the global performance of the system:

$$\text{F} - \text{measure} = \frac{2*(\text{Precision}*\text{Recall})}{(\text{Precision}+\text{Recall})} \tag{15}$$

The Accuracy measurement The Accuracy is the proportion of relevant classified conversation among the total number of conversations: Accuracy is one of measures most known as intuitive performance measure. High accuracy shows the performance of our model.

$$\textbf{Accuracy} = \frac{\mathbf{N_{RC}+N_{IRC}}}{\mathbf{N_{RC}+N_{IRC}+N_{RNRC}+N_{IRRC}}} \tag{16}$$

The Geometric Mean (G-mean) measurement This metric indicates the geometric mean of the recall. G-mean is the balance of classification performance on majority and minority classes. A poor performance of the event in classification causes a low G-mean if the irrelevant conversations are correctly classified. It is given by the next equation:

$$\mathbf{G-mean} = \sqrt{\frac{\mathbf{N_{RC}}}{\mathbf{N_{RC}+N_{RNRC}}}\frac{\mathbf{N_{IRC}}}{\mathbf{N_{IRC}+N_{IRRC}}}} \tag{17}$$

6.3. Experimental Setup and Discussion

In this work, we have tested our model with 3 sentiment analysis models Textblob, Flair and Vader, using a python library which provides access to different NLP tasks such as POS tagging, sentiment analysis, noun phrase extraction, translation, and classification. We have applied our model with 3 different cases:

- We have tested our model without similarity between tweets. In this case we aggregate polarity of tweet and reply ($CSAM_{AP}$). We use this formula:

$$\mathbf{Prob(T|R)} = \frac{\mathbf{Pol(T)}}{\mathbf{Pol(R)}} \tag{18}$$

We have applied our model with similarity without integration of conditional probability. Since we have to calculate sentiment of all conversation, we need to calculate similarity between tweet and reply ($CSAM_{APS}$):

$$\mathbf{Prob(T \cap R) = Pol(T) * Sim_{cos}(T, R)} \tag{19}$$

- We use our model with 3 models Textblob, Flair and Vader.

$$Bel(C_T) = \sum_{i=1}^{N} Prob_i(R|T) \tag{20}$$

We calculate the precision, recall, F-measure, G-mean and accuracy of 3 sentiments: happy, angry and others. Happiness and other positive feelings are taken as positive emotions (1), whereas sadness, angry are considered as negative emotions (-1) and others as neutral.

6.4. Experimental Results and Analyzes

Firstly, we have tested our model with three sentiment analysis packages: Textblob, Flair, Vader. The results of this analysis are summarized in table 4, 5 and 6 that represent respectively happy, angry and others. We have applied our model with SemEval2019 dataset of 2755 (360 positive, 1021 negative and 1374 neural) conversations manually labeled [43] using f-measure, precision, recall, accuracy and g-mean as the performance metrics. As shown in table 7, the mean precision given by our model CSAM using Textblob package is 53%, using Flair is 47% and using Vader is 57%. On the one hand, Vader out performs (Textblob and Flair) in mean precision measure. On the other hand, F-measure confirms that Textblob and Flair are better than Vader. Hence, to choose the best model we calculate accuracy, G-mean and F-measure for three sentiment classes happy, angry and others (see table 7).

In table 4 we observe that we achieved the best precision and accuracy with $CSAM_{Vader}$ to classify happy conversation, while the highest recall, f-measure and g-mean was obtained with $CSAM_{TextBlob}$. Tables 5 and 6 to classify angry conversations and other conversations (negative, neutral) respectively. Values in bold indicate the best results. As we can observe in the tables 4, 5 and 6, our model reaches the highest F-measure and G-mean with 3 sentiment analysis classes (happy, angry and others). We obtain 87% precision with $\mathit{CSAM}_{\mathit{Vader}}$ for angry, on the contrary we achieved the best recall-measure, g-mean and accuracy with $\mathrm{CSAM}_{\mathrm{TextBlob}}$. However, with others conversations we have the best precision, g-mean and accuracy with CSAMTextBlob while the best recall and f-measure with $\mathrm{CSAM}_{\mathrm{Vader}}$. We observe that the best result was achieved with $\mathrm{CSAM}_{\mathrm{TextBlob}}$ and CSAMvader in contrary the results of $\mathrm{CSAM}_{\mathrm{flair}}$ are relatively similar.

Table 4. The average score results of the proposed approach "CSAM" with state-of-the-art sentiment analysis models TextBlob,Flair and Vader using happy characteristic with SemEval2019 dataset

Results (in %) using Happy characteristic			
Measure	$\mathrm{CSAM}_{\mathrm{TextBlob}}$	$\mathrm{CSAM}_{\mathrm{Flair}}$	$\mathrm{CSAM}_{\mathrm{Vader}}$
Precision	23	24	**36**
Recall	**78**	43	12
F-measure	**36**	31	18
G-mean	**64**	55	34
Accuracy	57	66	**77**

Table 5. The average score results of the proposed approach csam with state-of-the-art sentiment analysis models TextBlob, Flair and Vader using angry characteristic with SemEval2019 dataset

Results (in %) using Angry characteristic			
Measure	$\mathrm{CSAM}_{\mathrm{TextBlob}}$	$\mathrm{CSAM}_{\mathrm{Flair}}$	$\mathrm{CSAM}_{\mathrm{Vader}}$
Precision	68	67	**87**
Recall	**53**	34	4
F-measure	**60**	46	7
G-mean	**63**	54	19
Accuracy	**65**	61	58

Table 6. The average score results of the proposed approach "CSAM" with state-of-the-art sentiment analysis models TextBlob, Flair and Vader using other characteristic with SemEval2019 dataset.

Results (in %) using Others characteristic			
Measure	$\mathrm{CSAM}_{\mathrm{TextBlob}}$	$\mathrm{CSAM}_{\mathrm{Flair}}$	$\mathrm{CSAM}_{\mathrm{Vader}}$
Precision	**68**	5	5
Recall	38	58	**95**
F-measure	49	54	**66**
G-mean	**54**	48	24
Accuracy	**55**	49	51

To confirm our interpretation, we calculate the average score in 3 sentiment analysis model presented in table 7. As we observe in table 7, 48% with $\mathrm{CSAM}_{\mathrm{Textblob}}$, 43% with $\mathrm{CSAM}_{\mathrm{Flair}}$ and 30% $\mathrm{CSAM}_{\mathrm{Vader}}$ score. These results are coherent with many researches that confirm that the difficulty in sentiment analysis is to define positive (happy) and negative (angry) classes contrary to neutral(others) [44]. Also, the obtained score in positive and negative classes with $\mathrm{CSAM}_{\mathrm{Textblob}}$ are the best, compared to Flair and Vader. These results confirm that our model with $\mathrm{CSAM}_{\mathrm{Textblob}}$ package present the highest score. Indeed, we conclude that aggregation with evidence theory and the integration of conditional probability with $\mathrm{CSAM}_{\mathrm{Textblob}}$ improve the result of classification conversation.

We provide some experimental evaluation of our model with cosine similarity on the one hand and without similarity on the other as shown in tables 10 and 9.

In table 9 and 10, we can observe that if we apply Textblob package in 2 cases $\mathrm{CSAM}_{\mathrm{AP}}$ (CSAM with aggregation polarity) $\mathrm{CSAM}_{\mathrm{APS}}$ (CSAM with aggregation polarity and similarity), mean f-measure score of $\mathrm{CSAM}_{\mathrm{AP}}$ increases about 0.18 compared to $\mathrm{CSAM}_{\mathrm{APS}}$. By carefully examining the data, it is found that the similarity decreases the result, so we have used cosine similarity to calculate the correspondence between terms. Cosine similarity is considered like lexical similarity based on word frequency. This similarity neglects semantics between terms. Secondly, we evaluate our model with another dataset ScenarioSA [42]. We obtain the best precision with three sentiments happy, angry and others. We use our model with Textblob. We applied CSAM model:

$$Bel(C_T) = \sum_{i=1}^{N} Prob_i(R|T) \tag{21}$$

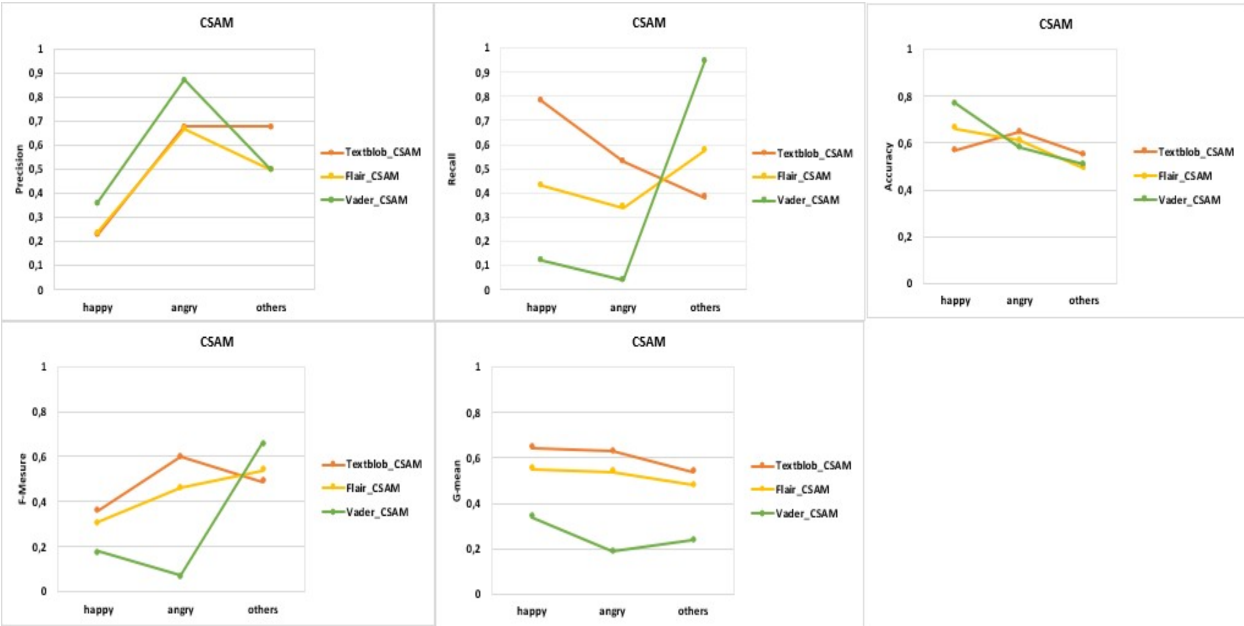

Fig.5. The average score results of CSAMTextblob: CSAM with Textblob SemEval2019

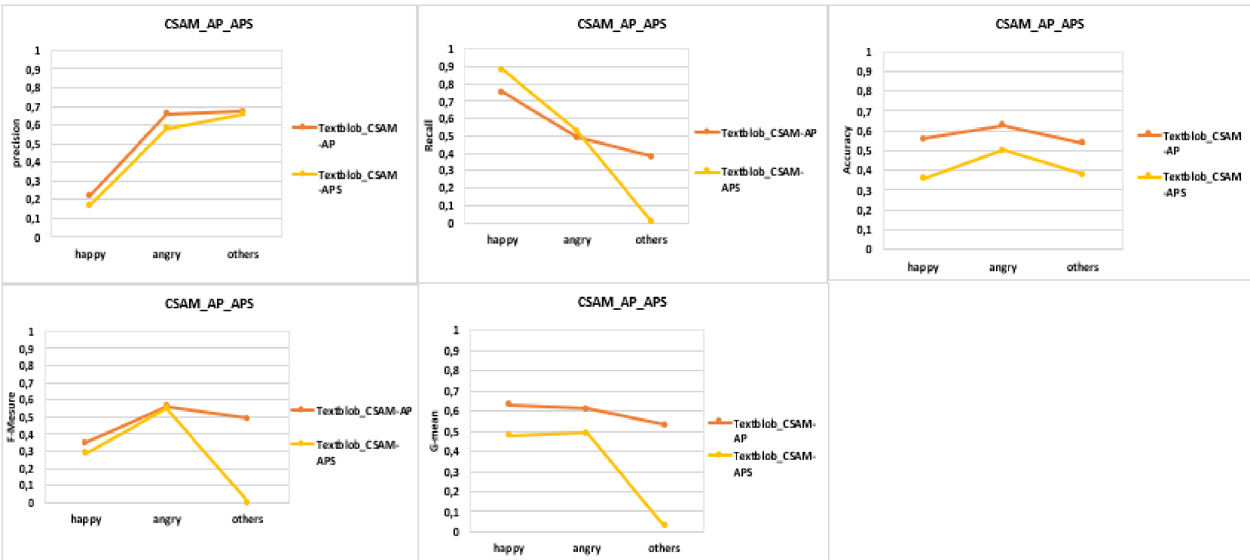

Fig.6. The average score results CSAM-APTextblob vs CSAM -APSTextblob: CSAM with aggregation polarity vs CSAM with aggregation polarity and similarity (SemEval 2019)

Table 12 present the obtained score in which we obtain high precision, recall, f-measure score with angry and others sentiment contrary to happy sentiment we obtain 63% accuracy and 67% g-mean which confirm the difficulties to analyse happy sentiment. The results confirm that our model provide the highest score with ScenarioSA dataset. Indeed, we conclude that the aggregation with Textblob improve the result of classification conversation.

6.5. Results Analysis and Discussion

Several studies have used SemEval2019 corpus [45-47]. They apply a deep learning approach to analyze sentiment conversation. Deep learning method offers a score between [1.43, 79.59%], the average F-measure of 3 classes (happy, angry, sad) obtained is 65.99%. However, deep learning requires a large amount of training data. So, without using training data and deep learning approach, we focus on conditional probability and belief theory to aggregate tweets to get a sentiment conversation. Our CSAM model record the following results: we have recorded 48% average F-measure value and 65.99% average F-measure with deep learning. We evaluate our model with another dataset ScenarioSA in which we achieved high scores with multi-comments and different conversation length. The number of comments in conversation improve the results with ScenarioSA compared to SemEval 2019 that contains 3 turns. We compare our model to [47] and [3]. Table 13 present f-measure score achieved by our model compared to [47] in which they use Naive Bayes with emotion and N-gram features to classified conversation as happy, angry, sad and others. They reached 34.59% average F-Measure score compared to our model 49% with the SemEval 2019. Also, [3] used LR (Logistic Regression) and BERT as sentence encoder to classify conversation in SemEval 2019. They have achieved 44,95% f-measure score. In our work, we start by exploring the use of both the conditional probability and the belief function in a sentiment analysis context in the case of conversational data. The result obtained confirms that these methods can provide very good classification results, although most of the tweets are neutral. At the end of these results, we can also conclude that several interesting aspects can be further developed with the natural language used in the conversation and the presence of syntactic errors. Sentiment analysis should also be more accurate, with better use of statistical methods. It is well known that users influence each other and their response depends on previous tweets. and most of the conversations are sets of tweets with their responses which we need to distinguish users. To analyze sentiment conversation, context and microblog features are the main source of prediction. We may use contextual and social features to improve the result.

Table 7. The average score results of the proposed approach "CSAM" with state-of-the-art sentiment analysis models TextBlob, Flair and Vader with SemEval2019 dataset.

Results (in %) using all characteristic			
Measure	$\text{CSAM}_{\text{TextBlob}}$	$\text{CSAM}_{\text{Flair}}$	$\text{CSAM}_{\text{Vader}}$
Precision	53	47	**57**
Recall	**56**	45	37
F-measure	**48**	43	30
G-mean	**60**	52	25
Accuracy	**59**	58	**62**

Table 8. The average score results of the proposed approach "CSAM" compared with "CSAM -AP" and "CSAM-APS" with state-of-the-art sentiment analysis model TextBlob

Results (in %) using TextBlob (TB) characteristic			
Measure	CSAM_{TB}	$\text{CSAM-AP}_{\text{TB}}$	$\text{CSAM-APS}_{\text{TB}}$
Precision	53	51	47
Recall	56	54	47
F-measure	48	46	28
G-mean	60	59	42
Accuracy	59	57	41

Table 9. The results obtained by applying CSAM with aggregation polarity (CSAM-AP) using all characteristics with Sem Eval 2019 dataset

Results (in %) given by CSAM-AP			
Measure	Happy	Angry	Others
Precision	22	66	67
Recall	75	49	38
F-measure	35	56	49
G-mean	63	61	53
Accuracy	56	63	54

6.6. Limits of the Proposed Model

Although the cited advantages of the proposed approach, some limits are distinguished. The main objective of the proposed approach is to propose an efficient solution for twitter conversation sentiment analysis. However, despite its good performance in terms of precision, recall, f-measure, G-mean and accuracy, this solution does not take into account the semantic relationships between terms to represent information and relation between these concepts. Moreover, it does not integrate meta level features like microblog features such as hashtag. This limit penalizes the great performance of our solution and does not allow us to consider it as a perfect solution.

Table 10. The results obtained by applying CSAM with similarity (CSAM-APS) using all characteristics with Sem Eval 2019 dataset

Results (in %) given by CSAM-APS			
Measure	Happy	Angry	Others
Precision	17	58	66
Recall	88	53	0.1
F-measure	29	55	0.2
G-mean	48	49	30
Accuracy	36	50	38

Table 11. The average score results of the proposed approach CSAM with state-of-the-art sentiment analysis models TextBlob using ScenarioSA dataset

Results (in %) using all characteristics			
Measure	Happy	Angry	Others
Precision	100	100	100
Recall	57	53	67
F-measure	73	69	80
G-mean	63	94	99.6
Accuracy	67	91	99.4

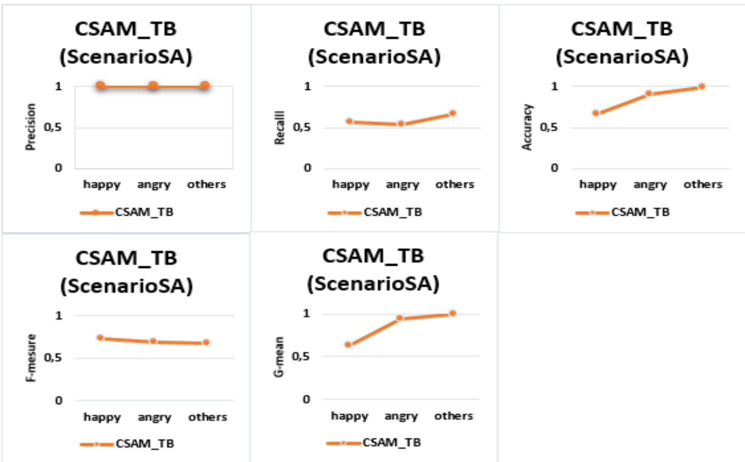

Fig.7. Precision CSAMTextblob : CSAM with Textblob ScenarioSA Dataset

Table 12. The average score results of the proposed approach CSAM with state-of-the-art sentiment analysis models TextBlob, with ScenarioSA dataset

Results (in %) using all characteristics						
	ScenarioSA			SemEval2019		
Measure	Happy	Angry	Others	Happy	Angry	Others
Precision	**100**	**100**	**100**	23	68	68
Recall	57	53	67	**78**	53	38
F-measure	73	69	**80**	36	60	49
G-mean	63	94	**99.6**	64	63	54
Accuracy	67	91	**99.4**	57	65	55

Table 13. Comparison of the proposed approach "CSAM" with [47] and [3], using SemEval2019 dataset.

Results (in %) using all characteristic			
Measure	$CSAM_{TextBlob}$	[47]	[3]
F-measure	**49**	34.94	44.51

7. Conclusion and Prospects

7.1. Summary

In this paper, we have presented a new model of calculating sentiment conversation based on belief function and conditional probability. We use three sentiment analysis model, Textblob, Flair and Vader. To evaluate our method, we use two datasets SemEval2019 and ScenarioSA. First, we test our model with SemEval2019 dataset. We compare our model using Textblob, Flair and Vader as a sentiment analysis technique. The results show that the use of Textblob with our model outperforms Flair and Vader. Second, we evaluate our model with another dataset ScenarioSA in which we achieved high scores with multi-comments and different conversation length. We have conducted different standard measures such as precision, recall, F-measure and G-mean. The founded results analysis confirms the feasibility of our model and its performance.

7.2. Prospects

As future work, we plan forth a new direction. The first direction is to work with other data collections and details through a more comparative study to resolve the problem of Conversational Sentiment Analysis in tweets. We hope to confirm the performance of our model via this study. As a second direction, several studies confirm conversation. This combination could improve the performances and the results. Third, since the functionality of ontology and the importance of semantics to understand the meaning of information. We propose to include ontology in the importance of deep learning and possibility theory. Thus, we plan to use the possibility to calculate sentiment deep learning model. The fourth direction consists in making our model more efficient by taking into account (i) the semantics between term to represent information and relation between these concepts and (ii) the integration of other meta level features like microblog features such as hashtag, which will certainly improve the results.

References

[1] Pang et al. Opinion mining and sentiment analysis. Foundations and Trends in Information Retrieval, 2:1– 135, 01 2008.

[2] Zhang et al. Scenariosa: a dyadic conversational database for interactive sentiment analysis. IEEE Access, 8:90652–90664, 2020.

[3] Winata et al. Caire_hkust at semeval-2019 task 3: Hierarchical attention for dialogue emotion classification. arXiv preprint arXiv:1906.04041, 2019.

[4] Wei et al. Modeling conversation structure and temporal dynamics for jointly predicting rumor stance and veracity. arXiv preprint arXiv:1909.08211, 2019.

[5] Mohamed Nazih Omri. Possibilistic pertinence feedback and semantic networks for goal's extraction. Asian Journal of Information Technology, 3(4):258– 265, 2004.

[6] Mohamed Nazih Omri. Relevance feedback for goal's extraction from fuzzy semantic networks. Asian Journal of Information Technology, 3(6):434–440, 2004.

[7] Bing Liu. Opinion mining and sentiment analysis. In Web Data Mining, pages 459–526. Springer, 2011.

[8] Sailunaz et al. Emotion and sentiment analysis from twitter text. Journal of Computational Science, 36:101003, 2019.

[9] Lian et al. Smin: Semi-supervised multi-modal interaction network for conversational emotion recognition. IEEE Transactions on Affective Computing, 2022.

[10] Park et al. Plusemo2vec at semeval-2018 task 1: Exploiting emotion knowledge from emoji and# hashtags. arXiv preprint arXiv:1804.08280, 2018.

[11] Barbosa et al. Robust sentiment detection on twitter from biased and noisy data. In Coling 2010: Posters, pages 36–44, 2010.

[12] Song et al. Sacpc: A framework based on probabilistic linguistic terms for short text sentiment analysis. Knowledge-Based Systems, 194:105572, 2020.

[13] Görmez et al. Fbsem: a novel feature-based stacked ensemble method for sentiment analysis. International Journal of Information Technology and Computer Science, 6:11–22, 2020.

[14] Azer et al. Credibility detection on twitter news using machine learning approach. International Journal of Intelligent Systems and Applications, 13(3):1–10, 2021.

[15] Shobana et al. Twitter sentimental analysis. International Journal of Recent Technology and Engineering (IJRTE), 7, 2018.

[16] Pang et al. Thumbs up?: sentiment classification using machine learning techniques. In Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10, pages 79–86. Association for Computational Linguistics, 2002.

[17] Das et al. Yahoo! for amazon: Extracting market sentiment from stock message boards. In Proceedings of the Asia Pacific finance association annual conference (APFA), volume 35, page 43. Bangkok, Thailand, 2001.

[18] Zhang et al. Combining lexicon-based and learningbased methods for twitter sentiment analysis. HP Laboratories, Technical Report HPL-2011, 89, 2011.
[19] Kaur et al. Twitter sentiment analysis of the indian union budget 2020. 2020.
[20] Kunal et al. Textual dissection of live twitter reviews using naive bayes. Procedia computer science, 132:307–313, 2018.
[21] Golam Mostafa, Ikhtiar Ahmed, Masum Shah Junayed, "Investigation of Different Machine Learning Algorithms to Determine Human Sentiment Using Twitter Data", International Journal of Information Technology and Computer Science(IJITCS), Vol.13, No.2, pp.38-48, 2021. DOI: 10.5815/ijitcs.2021.02.04
[22] Alvarez-Melis et al. Topic modeling in twitter: Aggregating tweets by conversations. In Proceedings of the International AAAI Conference on Web and Social Media, volume 10, 2016.
[23] Naouar et al. Information retrieval model using uncertain confidence's network. International Journal of Information Retrieval Research, 7(2):33–49, 2017.
[24] Boughammoura et al. Information retrieval from deep web based on visual query interpretation. International Journal of Information Retrieval Research, 2(4):45–59, 2012.
[25] Amer et al. Recherche de conversations dans les réseaux sociaux: modélisation et expérimentations sur twitter. 2015.
[26] Meester et al. A new look at conditional probability with belief functions. Statistica Neerlandica, 73(2):274–291, 2019.
[27] Hall et al. Cross-validation and the estimation of conditional probability densities. Journal of the American Statistical Association, 99(468):1015–1026, 2004.
[28] Van Campenhout et al. Maximum entropy and conditional probability. IEEE Transactions on Information Theory, 27(4):483–489, 1981.
[29] Glenn Shafer. A mathematical theory of evidence, volume 42. Princeton university press, 1976.
[30] Silva et al. Predicting misinformation and engagement in covid-19 twitter discourse in the first months of the outbreak. arXiv preprint arXiv:2012.02164, 2020.
[31] Loria et al. Textblob: simplified text processing. Secondary TextBlob: simplified text processing, 3, 2014.
[32] Hutto et al. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In Proceedings of the International AAAI Conference on Web and Social Media, volume 8, 2014.
[33] Akbik et al. Contextual string embeddings for sequence labeling. In Proceedings of the 27th international conference on computational linguistics, pages 1638–1649, 2018.
[34] Felbo et al. Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. arXiv preprint arXiv:1708.00524, 2017.
[35] Jianqiang et al. Deep convolution neural networks for twitter sentiment analysis. IEEE Access, 6:23253– 23260, 2018. [36] Pak et al. Twitter as a corpus for sentiment analysis and opinion mining. In LREc, volume 10, pages 1320– 1326, 2010.
[36] Carvalho et al. On the evaluation and combination of state-of-the-art features in twitter sentiment analysis. Artificial Intelligence Review, pages 1–50, 2020.
[37] Carvalho et al. An assessment study of features and meta-level features in twitter sentiment analysis. In Proceedings of the Twenty-second European Conference on Artificial Intelligence, pages 769–777, 2016.
[38] Agarwal et al. Sentiment analysis of twitter data. In Proceedings of the workshop on language in social media (LSM 2011), pages 30–38, 2011.
[39] Heck et al. Deep learning of knowledge graph embeddings for semantic parsing of twitter dialogs. In 2014 IEEE Global Conference on Signal and Information Processing (GlobalSIP), pages 597–601. IEEE, 2014.
[40] Philippe Smets. The combination of evidence in the transferable belief model. IEEE Transactions on pattern analysis and machine intelligence, 12(5):447–458, 1990.
[41] Zhang et al. Scenariosa: a dyadic conversational database for interactive sentiment analysis. IEEE Access, 8:90652–90664, 2020.
[42] Chatterjee et al. Semeval-2019 task 3: Emocontext contextual emotion detection in text. In Proceedings of the 13th International Workshop on Semantic Evaluation, pages 39–48, 2019a.
[43] Alswaidan et al. A survey of state-of-the-art approaches for emotion recognition in text. Knowledge and Information Systems, pages 1–51, 2020.
[44] Ke et al. Lightgbm: A highly efficient gradient boosting decision tree. Advances in neural information processing systems, 30:3146–3154, 2017.
[45] Devlin et al. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018.
[46] Peters et al. Deep contextualized word representations. arXiv preprint arXiv:1802.05365, 2018.
[47] Chatterjee et al. Understanding emotions in text using deep learning and big data. Computers in Human Behavior, 93:309–317, 2019.

Authors' Profiles

Imen Fadhli is currently a Ph.D. student at the Higher Institute of Computer Science and Telecom (ISITCom), University of Sousse, Tunisia. She received her Masters degree from the Higher Institute of Management, University of Sousse, Tunisia; in computer science in 2017. Her current research interests include the detection of the credible conversation in Twitter microblogs. She is member of the research Laboratory MARS (Modeling of Automated Reasoning Systems).

Lobna Hlaoua In 2007, she obtained her Ph.D. in Computer Science from the University Toulouse-III Paul-Sabatier in France. Since September 2008, she has been an associate professor of computer science at the University of Sousse in Tunisia, and since 2011, she has been a member of MARS (Modeling of Automated Reasoning Systems) Research Laboratory. Her research group focuses on information retrieval, social network analysis, and data mining. She co-supervised 5 Ph.D. and 10 master theses. She is reviewer of the Journal of the Association for Information Science and Technology (JASIST).

Mohamed Nazih Omri In 1994, he received his Ph.D. in Computer Science from the University of Jussieu in Paris, France. He is a computer science professor at the University of Sousse in Tunisia. He has been a founding member of MARS (Modeling of Automated Reasoning Systems) Research Laboratory since January 2011. His research group focuses on information retrieval, data bases, knowledge bases, and web services. He supervised more than 20 Ph.D. and Master students in various areas of computer science. He is a reviewer for many international journals, including Information Fusion, Psihologija, and many international conferences, including AMIA, ICNC-FSKD, AMAI, and others.

How to cite this paper: Imen Fadhli, Lobna Hlaoua, Mohamed Nazih Omri, "Sentiment Analysis CSAM Model to Discover Pertinent Conversations in Twitter Microblogs", International Journal of Computer Network and Information Security(IJCNIS), Vol.14, No.5, pp.28-46, 2022. DOI:10.5815/ijcnis.2022.05.03