information

Article

# Towards Aiding Decision-Making in Social Networks by Using Sentiment and Stress Combined Analysis

**Guillem Aguado *,†, Vicente Julián † and Ana Garcia-Fornes †**

Departamento de Sistemas Informáticos y Computación, Universidad Politécnica de Valencia, 46022 Valencia, Spain; vinglada@dsic.upv.es (V.J.); agarcia@dsic.upv.es (A.G.-F.)

* Correspondence: guiagsar@dsic.upv.es; Tel.: +36-679-109-226

† These authors contributed equally to this work.

Received: 15 February 2018; Accepted: 25 April 2018; Published: 2 May 2018

**Abstract:** The present work is a study of the detection of negative emotional states that people have using social network sites (SNSs), and the effect that this negative state has on the repercussions of posted messages. We aim to discover in which grade a user having an affective state considered negative by an Analyzer can affect other users and generate bad repercussions. Those Analyzers that we propose are a Sentiment Analyzer, a Stress Analyzer and a novel combined Analyzer. We also want to discover what Analyzer is more suitable to predict a bad future situation, and in what context. We designed a Multi-Agent System (MAS) that uses different Analyzers to protect or advise users. This MAS uses the trained and tested Analyzers to predict future bad situations in social media, which could be triggered by the actions of a user that has an emotional state considered negative. We conducted an experimentation with different datasets of text messages from Twitter.com to examine the ability of the system to predict bad repercussions, by comparing the polarity, stress level or combined value classification of the messages that are replies to the ones of the messages that originated them.

**Keywords:** multi-agent system; social networks; sentiment analysis; stress analysis; combined analysis; advice

## 1. Introduction

In the current society, we are inside an environment of online applications. One of the most present of them are Social Network Sites (SNSs). This may cause problems for the users since several situations can raise risks or negative consequences for being in an SNS [1]. In addition, teenagers face several risks at SNSs and have some characteristics that make them more vulnerable to those risks [2], and publishing a post can have negative consequences and lead to regret [3].

In [4], the effects that incidental moods, discrete emotions, integral affect, and regret can have in the decision-making process are reviewed. By incidental moods and discrete emotions, we mean affective states that are not linked directly with the task at hand (e.g., moods and emotions at the time of making a decision); integral affection is an affection that arises from the task that is being performed [4]. The researchers explained that incidental moods proved to affect decision-making by altering people's perception. Discrete emotions, integral affect, and regret were also shown to affect decision-making, and regret can have an effect in the form of anticipated regret (thinking of the bad outcome before it actually happens).

Moreover, it has been proved that a negative emotional state can lead to poor decision-making. In [5], a study demonstrates the role of cognitive bias, emotional distress and poor decision-making in the gambling disorder. More concretely, Ref. [5] demonstrates that pathological gambling correlated with negative emotional states. The more individuals have a problematic gambling involvement, the more they experience anxiety and depression.

Since stress has been observed to be associated with a concrete emotional state [6], it is suitable for building a system that analyzes the emotional state of the users.

Multi-Agent System (MAS) are systems composed of multiple software agents that interact together, and they are usually used to address computational problems that require different software entities to be independent and interactive such as online trading or complex simulations with multiple entities, like social response modeling. Our goal is to create an MAS that will be integrated into a social environment, like a social network, and incorporates different Analyzers, and a combined version of Analyzer. Those Analyzers will be able to recognize the emotional state of the users in the social environment and their stress level and to advise them to post or not by analyzing the text messages before they post them. Finally, we aim to perform an experimentation with data extracted from Twitter.com, in order to be able to discover what Analyzer is more suitable for predicting a future negative outcome in the social environment (potential negative outcome caused by negative labelled data in the text messages), and in which cases. In the experimentation, we will use the replies of the text messages to discover if the emotional state or stress level detected has propagated or not from the original messages to the direct follow-up replies, and to study which Analyzer has been able to predict better this propagation. For this purpose, we designed an MAS that includes a set of agents in charge of performing different kinds of analyses (Sentiment, Stress, and Combined), and that interacts with the users advising them at the moment of publishing a message. This system will be integrated into an SNS to perform advice to the users according to their personal state, in order to help them with the decision-making process, and for avoiding possible future bad situations that could arise as a result of posting something while suffering cognitive distress (e.g., talking to someone that is suffering cognitive distortions).

The present work deals mainly with two research areas, which are Sentiment Analysis and Stress Analysis, but it is also related to user state modeling and Multi-Agent Systems (MAS). In the following, we analyze recent works in those areas.

Sentiment Analysis is a field of research that intends to study the phenomena of opinions, sentiments, evaluations, appraisal, attitude, and emotion through different kinds of media (e.g., written messages, images, emoticons, etc.) [7]. Regarding Sentiment Analysis in written texts (which is the most common), we will find that there are four well-differentiated techniques: document-level Sentiment Analysis, sentence-level Sentiment Analysis, aspect-based Sentiment Analysis and comparative Sentiment Analysis [8]. The kind of analysis depends on the level of fine-grained Sentiment Analysis that we choose to perform, starting out from the document level Sentiment Analysis (sentiment from the entire document), to sentence level (sentiment in a sentence), and finally to the aspect based Sentiment Analysis (sentiment in concrete aspects, as sequences of words that can be one word, found in the text). Comparative Sentiment Analysis is an exception, where we use comparative sentences to learn that are the preferred entities, associated with comparative words appearing in the sentences (the sentiment words for the model) [8]. For the present study, we choose to use aspect based Sentiment Analysis on texts so we can perform a fine-grained analysis, focusing on terms that may contain sentiment and not entire sentences or documents, which may contain more than one.

Two different work lines are present in Sentiment Analysis, which sometimes are worked in hybrid approaches, and those are aspect extraction and sentiment classification [9]. For aspect detection, we can find detection through generative models (e.g., Conditional Random Fields or CRF), which use a variated set of features [10]; frequency based methods, which use the frequency of the terms in the training corpora to put them as aspects or not in the aspect set (the most frequent terms are added) [11]; and non-supervised machine learning techniques (e.g., Latent Dirichlet Allocation or LDA) [12]. We will use a frequency based method because it helps to know what aspects are the most frequently mentioned in an SNS.

In the case of sentiment classification, there are different methods like machine learning methods, which can be either supervised or not and dictionary-based methods. Machine learning methods use Support Vector Regression and other techniques to obtain the features for training the model, and non-supervised methods use other techniques like relaxation labeling [9]. Dictionary-based methods use a dictionary of aspects with a polarity assigned to them in the training step (with a method for training the aspect set), and a method for extracting polarities later from texts using the dictionary [9]. We choose a dictionary-based method because that way we will be able to have a set with sentiment aspects and another with stress aspects (sequences of words), with associated sentiment polarities or stress levels. Finally, regarding the hybrid approaches, they intend to detect aspects and assign polarities at the same time [9], but those are not used in the present work since we want to simplify the process of detecting aspects and assigning polarities and to modularize it.

TensiStrenght [6] is an algorithm derived from the SentiStrenght algorithm for sentiment strength detection, which uses a set of terms associated with stress and another set of terms associated with relaxation. Those are previously trained assigning levels of stress and relaxation to its aspect sets with an unsupervised method that use tweets annotated with stress and relaxation strengths, and then refining the values with a hill-climbing method. The sets are then used to detect stress and relaxation levels in sentences of written texts, with some improvements implemented in the algorithm such as detecting exclamation marks and boosting the strength of stress or relaxation within a sentence.

In the case of works trying to model the information of the user on a system, Rincon et al. [13] created a social-emotional model that detects the social emotion of a group of entities. They used the PAD (Pleasure, Arousal and Dominance) three-dimensional emotional space for representing the emotions of the entities and an artificial neural network to learn the emotion of the group in the context of an event that just happened; Gao et al. [14] used a model for a task of sentiment classification that computes the user and product-specific sentiment inclinations; In [15], a nearest-neighbor collaborative approach was used to train user-specific classifiers, which were finally combined with user similarity measurement in a Sentiment Analysis task.

MAS for helping or guiding users (like the one presented in the current work) have been worked on before. An MAS based system named PATRASH (Personalized Autonomous TRAansit recommendation System considering user context and History) was presented in [16]. It was designed as an application for public transit guide system, where each agent interacts with each other by exchanging messages of CSV-format (Comma-separated values file format).

The contributions of the present work are: a design of agentized versions of a sentiment Analyzer and a stress Analyzer, the design of a novel agentized combined Analyzer, the experimentation that aims to discover what agent performs better at predicting a potential future bad outcome, and in which cases, by calculating the value of the different analysis on short text messages, and finally the design of an MAS that integrates the Analyzers and use those values to advise the users of a social environment such as a Social Network. With this work, we will be able to help building systems integrated into social environments like SNSs, where there are several users interacting together. Such systems will be able to help them in their experience, by guiding their decision-making processes (e.g., warning them when they could generate a problem or be in one by means of the different analysis of data), and we will also be able to detect the cases where each analysis performs better than others.

## 2. Materials and Methods

Our system has been designed as an MAS that will analyze data from written text messages so it can give recommendations and warnings to the user for helping in the social experience. We designed the system as agent types, which are components of the MAS. These agents perform different tasks on the system and communicate with other agents in order to accomplish their tasks. They use the SPADE 2.3 multi-agent platform [17], which is distributed under a GNU Lesser General Public License v2 (LGPLv2), for their implementation.

The MAS proposed has three layers, which follow a presentation, logic and persistence layers architecture. The system is structured into diverse agent types that operate in the different layers and each one has a different task to perform. The presentation layer has an agent type to show information to the user and to get the information of the user and send it to the logic layer. The logic layer has agent types that perform the analysis and calculations of the system and generate recommendations or advice for the users, the agents get input from the presentation and persistence layers and send information to the persistence layer for storing it. Finally, the persistence layer has the agent type that stores the data into the database and provides it to the logic layer when it is needed. The architecture of the MAS can be seen in Figure 1. As we can see, the advisor agent can either get the information of sentiment polarities and stress levels from the Analyzer agents, when a user is posting on the SNS or from the database.

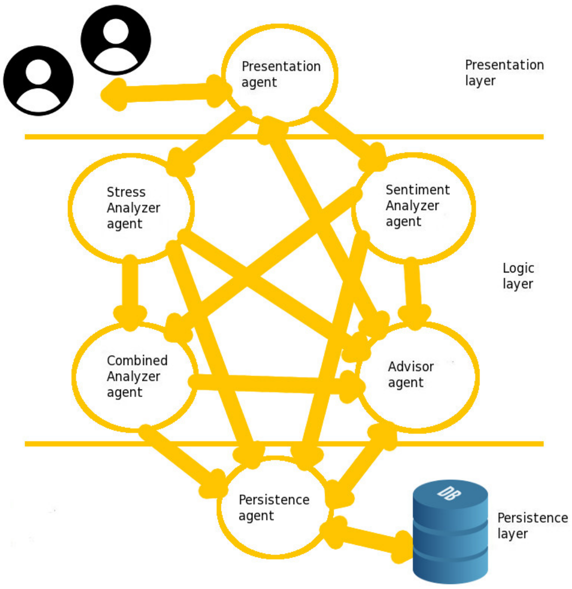

**Figure 1.** Architecture of the system built as a Multi-Agent System (MAS) with three layers.

As we can see in Figure 1, there are three different analyzer agents, which correspond to the Sentiment Analyzer agent, the Stress Analyzer agent and the Combined Analyzer agent. Each of those agents will be performing the three different kinds of analyses on the system. The Combined Analyzer agent will be interacting with the two other agents to perform a Combined Analysis. Other auxiliary artifacts have been built to pre-process and translate tweets and to extract data from Twitter.com We will explain in more detail the agent types of the MAS in the next subsections.

### 2.1. Presentation Agent

The presentation agent type uses different widgets for getting information from the users on the SNS, so it can get the text messages or other data, and send it to the Analyzer agents in the logic layer. It also can get information about advice or warnings from the Advisor agent and show a warning, telling the user to be careful.

### 2.2. Sentiment Analyzer Agent

The Sentiment Analyzer agent type is the one in charge of taking the short text messages as input (e.g., from a social network), analyzing it using a previously trained aspect set, and giving, as a result, its sentiment polarity (positive, negative or neutral). He sends this polarity to other agents for storing it in the database or for using it in more calculations and generating advice or warnings for the users. For designing it, various decisions have been made:

1. Aspect-based Sentiment Analysis: The kind of Sentiment Analysis chosen for the system was aspect-based. This type of Sentiment Analysis, as explained in the previous section, performs an analysis based on concrete aspects found in the sentences of texts, creating the model as an aspect set with associated polarities and later using it to perform the classification on text messages. We used an annotated dataset with polarities assigned to short written messages (tweets), extracted from diverse variated topics (e.g., politics) for training the model. This dataset is extracted from the TASS (Taller de Análisis de Sentimientos) experimental evaluation workshop [18,19].
2. Aspect extraction: We selected a frequency-based method for performing the aspect extraction, where we create aspects as the terms found in the training corpora, which are unigrams. We select then the terms or aspects with a higher frequency of appearance in the corpora to constitute the aspect set.
3. Sentiment classification: Since we have an annotated corpora of data with sentences labeled with a polarity, we classified the aspects of the aspect set using those labels, assigning to them a polarity as the one with a major appearance on the training labeled corpora (the corpora assigns polarities to sentences, so we took those polarities as associated with the terms appearing in the sentence), which means that we use a Bayesian classifier.
4. Sentence classification: For using the model, we perform a classification of short written texts as follows: All the possible n-grams of the message are compared with each aspect of the aspect set, and, if an aspect is found, we store that information. Finally, when all the aspects of the aspect set are compared, we determine the sentiment of the message as the most predominant polarity found from the previous exploration—either positive, negative or neutral.

### *2.3. Stress Analyzer Agent*

The Stress Analyzer agent type is similar to the Sentiment Analyzer agent type, but it assigns levels of stress to the aspects of the aspect set instead of sentiment polarities. In that manner, we can find low stress level, normal stress level and high stress level associated with an aspect. The dataset used to train the model is also a dataset of messages written in a context in which stress is normally present. This dataset was composed of stress-related tweets annotated with stress strengths, coming from the work on TensiStrength [6], extracted from Twitter.com monitoring a set of stress and relaxation keywords.

### *2.4. Combination Analyzer Agent*

For designing this agent, we use the combined values of sentiment and stress from the text messages. We determined that, when stress is in low or normal levels, we assign the polarity of the message as the polarity of the Sentiment Analysis, but, when the stress levels are high, we directly assign the polarity of the message being analyzed with this combined model as negative. This is done in this way to determine the effect of high levels of stress in the repercussion of a message in an SNS, and we do not experiment with normal levels of stress as considering them negative because those are present in a multitude of situations for different reasons and we choose not to consider a medium level of stress as a negative state. Instead, we study the negative bad repercussions and user negative states as high levels of stress or negative sentiment polarity.

### *2.5. Advisor Agent*

The Advisor agent type performs a calculation based on the information returned by the Analyzer agents, calculated at the moment when a user is posting a message in the system. It has been designed to work as follows: first, when the user posts a message, it is sent to the logic layer by the presentation agent, so it is sent to every Analyzer. The Analyzers perform their analysis on the data and send the result to the database and to the Advisor agent. Finally, this agent, taking into consideration the output of every analyzer and the current domain of application (stress present environment or not), will decide whether to send a warning to the user or not if the Analyzers have indicated that the message may be showing a bad or negative state of the user that wrote it. For deciding what to take into account, regarding the output of the Analyzers, we aim to discover what agent works best in each situation, so we conducted an experimentation that has this purpose, which will be shown in the next section.

### *2.6. Persistence Agent*

The persistence agent type has functions for storing polarities, text messages and other data in the database. This agent also has functions to extract information from the database when an agent from the logic layer requests it and can either get or send information from or to the agents on the logic layer.

### *2.7. Design of the Experiment*

In this section, we will explain the experimentation conducted and the corpora of data that we used for performing it. We will also show the metrics used and, in the next section, discuss the results of all the experiments.

We have taken corpuses of data extracted from the popular SNS Twitter.com, composed of text messages of real people from all around the world and characterized for having a thematic on each corpora (e.g., political, cultural, etc.). The corpora has been created using the Twitter.com API (Application Programming Interface) for streaming tweets, and it has been processed using a function to clean them up for the Sentiment, Stress, or Combined Analyzer, which searches possible sources of error for the future analysis and corrects them. We have two corpuses for the main experimentation:

1. Podemos (A political corpora made of messages related to the politic party 'Podemos'). This is a very large corpora (about 1.9 millions of tweets).
2. Star Wars (A leisure corpora about the famous franchise of films Star Wars, which gathers tweets of people from all around the world). It is a very large corpora (about 12 millions of tweets), but only a small part of it is messages in Spanish, and we need them to be in that language because the aspect sets of the Analyzers are in Spanish only.

We also used the annotated corpora Stompol [18,19] for the calculation of the recall of the Analyzers. STOMPOL (corpora of Spanish Tweets for Opinion Mining at aspect level about POLitics) is a corpora of Spanish tweets prepared for the research in the task of opinion mining at aspect level.

In this work, we will try to determine the effect that the messages detected as negative or dangerous by the hand of the Sentiment and Stress Analyzers have in the messages that are a repercussion of them (we used the replies of the messages in this case).

We also want to determine the effect that tweets analyzed by our combined model (which uses different analyzers) have on the replies as well, and compare this effect to the effect that we observed in the previous stage (using only one analyzer at a time). With this information, we aim to determine whether it is more useful or informative to use only one analyzer or both combined, and in what situations.

We coded a function that reads and loads a tweet in JSON format and then analyzes it for knowing if it is a Spanish tweet (since, as mentioned before, our aspect sets of the Analyzers are in Spanish only), and if it is a reply of another tweet. If it passes the two filters, then we proceed to look at a dictionary where we store the analyzed replies, and if the original tweet that generated a reply is not present, then we search that tweet using the Twitter.com API. We calculate the sentiment and stress of both messages and store those values in the dictionary (if the original message was already present, we only calculate the ones of the reply).

When we have all the corpora analyzed, for all the tweets that generated replies, we do as explained in the following:

1. Calculate its combined value using both the sentiment and stress value in the way we explained in the Section 2.4.
2. Calculate the mode of the sentiment polarities in the replies of that tweet.
3. Calculate the mode of the stress levels found in the replies of the tweet.
4. Calculate the combined value of the replies using both modes previously calculated.

With this done, we finally proceed to calculate which tweets correspond to its replies in terms of comparing their calculated values (using Sentiment, Stress, and Combined values), with the calculated final values in the replies (using mode of sentiment, mode of stress levels and combined values on the replies). If it is the same value (positive, negative or neutral for sentiment, stress levels or combined value), we conclude that this tweet has generated a repercussion, according to what the model predicted.

Finally, we just accumulate the percentage of tweets that are in line (have the same emotional polarity associated, stress level or combined value), with the output of the Analyzers for the replies (using Sentiment, Stress, and Combined values), and store it as the result of the experiment.

### 2.7.1. Metrics of the Experimentation

We performed an experiment with an annotated corpora with tweets associated with a sentiment polarity, in order to discover the recall of the analyzers. For calculating the recall, we took the tweets that were classified as negative by the classifier and the total amount of tweets annotated as negative (or annotated as having a stress level associated as negative by the classifier). We used the following metrics for calculating the result of the experiments:

1. For Sentiment Analysis, percentage of concordance sentiment (PCsen):
   tweetsConc = Amount of tweets with the same emotional polarity than the mode in its replies.
   tweetsTotal = Amount of total tweets with replies analyzed:
   $$\mathrm{PCsen} = \frac{\mathrm{tweetsConc}}{\mathrm{tweetsTotal}}.$$
2. For Stress Analysis, percentage of concordance stress (PCstr):
   tweetsConc = Amount of tweets with the same stress levels than the mode in its replies.
   tweetsTotal = Amount of total tweets with replies analyzed:
   $$\mathrm{PCstr} = \frac{\mathrm{tweetsConc}}{\mathrm{tweetsTotal}}.$$
3. For the Combined Analysis, percentage of concordance combined (PCcomb):
   tweetsConc = Amount of tweets with the same value, combining emotional polarity and stress levels than the one calculated with the mode of the sentimental polarities and stress levels in its replies.
   tweetsTotal = Amount of total tweets with replies analyzed:
   $$\mathrm{PCcomb} = \frac{\mathrm{tweetsConc}}{\mathrm{tweetsTotal}}.$$
4. Recall for the Sentiment Analyzer (RecallSA):
   NegativeTweetsDetected = amount of tweets considered negative that the analyzer detected.
   NegativeTweets = Amount of tweets considered negative in the corpora:
   $$\mathrm{RecallSA} = \frac{\mathrm{NegativeTweetsDetected}}{\mathrm{NegativeTweets}}.$$

5. Recall for the Stress Analyzer (RecallStr):
NegativeTweetsDetected = amount of tweets considered negative that the analyzer detected (which in this case is associated with the stress level considered negative).
NegativeTweets = Amount of tweets considered negative in the corpora (again it is associated with the stress level considered negative):

$$\text{RecallStr} = \frac{\text{NegativeTweetsDetected}}{\text{NegativeTweets}}.$$

6. Recall for the Combined Analyzer (RecallCombined):
NegativeTweetsDetected = amount of tweets considered negative that the analyzer detected.
NegativeTweets = Amount of tweets considered negative in the corpora:

$$\text{RecallCombined} = \frac{\text{NegativeTweetsDetected}}{\text{NegativeTweets}}.$$

### 2.7.2. Plan of the Experiments

We will explain in this subsection how many experiments we launched, of what kind and with what corpora of data. As stated above, we launched an experiment with a corpora of annotated tweets called Stompol for calculating the recall of the analyzers. We used the number of tweets classified as negative and that actually had a negative polarity label coded by a human, and the total amount of tweets coded as negative. The fraction of negative tweets detected by the analyzer (NegativeTweetsDetected), and negative tweets in the corpora (NegativeTweets), are shown following for the three analyzers:

RecallSA = 79.12% ; RecallStr = 0.012% ; RecallCombined = 79.12%.

The Stress Analyzer agent has a very low recall, and this may be caused because we only use the high levels of stress to determine whether a user has a dangerous stress level or not. Nevertheless, it has proved to make a difference in the tests over a large amount of data (which will be shown in the following). The low recall of this Analyzer makes it less useful to be used alone because, even if it has a good concordance with the replies (PCstr), it may only detect a small number of negative stress states. Thus, trying to combine it with the Sentiment Analyzer may make it more useful.

The recall for the Sentiment Analyzer resulted in being the same as the Recall Combined. This may be caused because the Stompol corpora, which was used for the calculation of the recall of the analyzers is a small corpora, and the small number of detections from the Stress Analyzer were also detected (mostly or completely) by the Sentiment Analyzer:

1. Experimentation with the corpora Podemos: We prepared an experimentation with the corpora Podemos in the following way: We partitioned this corpora and, since it is a very large corpora, we decided to make six different partition sizes, doing four different experiments for each partition size. This was done in this way because the largest partition size was 1/4 of the corpora replies, and the maximum amount of parts that we could perform without using a tweet more than one time was four. We performed each experiment using the three different analyzers. The first partition is 1/128 of the total replies of the corpora for each experiment (around 1700 replies); the second partition is 1/64 of the replies; in this same way, the following four partitions are of 1/32, 1/16, 1/8 and 1/4 of the total replies, and the final results of the experimentation can be seen in Table 1.
2. Experimentation with the corpora Star Wars: we prepared the experimentation for the Star Wars corpora as shown in the following: we made partitions of the corpora with four different partition sizes and with three different experiments for each one, and we performed the partitions in this

way because even when the corpora is a large corpora, the amount of tweets in Spanish is not high, resulting in a modest amount of replies in Spanish (22,543 replies). Remember that, since the aspect sets of the Analyzers are built with aspects in Spanish, we can only analyze Spanish tweets with them. Again, the maximum amount of different experiments that could be performed with the biggest partition size, without using data in more than one different experiment, was used for all the partition sizes (in this case 3). For each experiment, we used the three analyzers, and the number of replies for each partition size were 1/3, 1/6, 1/12 and 1/24 of the total replies of the corpora. The final results of this experimentation are shown in Table 2.

**Table 1.** Experimentation with the corpora Podemos.

| *Partition Size* | *Experiment* | *PCsen* | *PCstr* | *PCcomb* |
|---|---|---|---|---|
| 1/128 of replies | 1 | 0.5975 | 0.9752 | 0.5944 |
| | 2 | 0.5594 | 0.9752 | 0.5644 |
| | 3 | 0.5881 | 0.9611 | 0.5943 |
| | 4 | 1.0 | 1.0 | 1.0 |
| 1/64 of replies | 1 | 0.5789 | 1.0 | 0.5789 |
| | 2 | 0.4583 | 1.0 | 0.4583 |
| | 3 | 0.5680 | 0.9813 | 0.5697 |
| | 4 | 0.4706 | 1.0 | 0.4706 |
| 1/32 of replies | 1 | 0.5 | 0.9833 | 0.5 |
| | 2 | 0.5682 | 1.0 | 0.5682 |
| | 3 | 0.5261 | 0.9799 | 0.5281 |
| | 4 | 0.5824 | 0.9780 | 0.5824 |
| 1/16 of replies | 1 | 0.5132 | 0.9737 | 0.5 |
| | 2 | 0.5156 | 0.9778 | 0.52 |
| | 3 | 0.5616 | 0.9726 | 0.5616 |
| | 4 | 0.5375 | 0.95 | 0.525 |
| 1/8 of replies | 1 | 0.5508 | 0.9786 | 0.5508 |
| | 2 | 0.5546 | 0.9738 | 0.5611 |
| | 3 | 0.5493 | 0.983 | 0.5511 |
| | 4 | 0.5864 | 0.978 | 0.5864 |
| 1/4 of replies | 1 | 0.5591 | 0.9694 | 0.5577 |
| | 2 | 0.5948 | 0.9752 | 0.6020 |
| | 3 | 0.5638 | 0.9741 | 0.5618 |
| | 4 | 0.5674 | 0.9787 | 0.5686 |

**Table 2.** Experimentation with the corpora Star Wars.

| *Partition Size* | *Experiment* | *PCsen* | *PCstr* | *PCcomb* |
|---|---|---|---|---|
| 1/3 of replies | 1 | 0.6107 | 0.9905 | 0.6069 |
| | 2 | 0.6 | 1.0 | 0.6 |
| | 3 | 0.6373 | 0.9707 | 0.6327 |
| 1/6 of replies | 1 | 0.6075 | 0.9791 | 0.6045 |
| | 2 | 0.6209 | 0.9783 | 0.6137 |
| | 3 | 0.6275 | 0.9902 | 0.6373 |
| 1/12 of replies | 1 | 0.6075 | 0.9794 | 0.6075 |
| | 2 | 0.6391 | 0.9699 | 0.6165 |
| | 3 | 0.6142 | 0.9864 | 0.6142 |
| 1/24 of replies | 1 | 0.6061 | 0.9865 | 0.6044 |
| | 2 | 0.52 | 0.98 | 0.5133 |
| | 3 | 0.625 | 0.875 | 0.625 |

We show the results of the experiments launched for the corpora Podemos in Figures 2 and 3, and the values of the experiments for each corpora size have been represented as one single point as the average of all the experiments launched for that corpora size. We separated the information about the Stress Analyzer experiments from the others because the percentage of concordance of this analyzer (PCstr) is very high and made it difficult to appreciate well the results of the others when they were shown in the same figure. Finally, we show the results for the experimentation with the corpora Star Wars in Figures 4 and 5, in the same way that we did in the case of the experimentation of the corpora Podemos. In the following two figures, the legend stands for:

1. SA and Stress A: Sentiment Analysis combined with Stress Analysis.
2. SA: Only Sentiment Analysis.
3. Stress A: Only Stress Analysis.

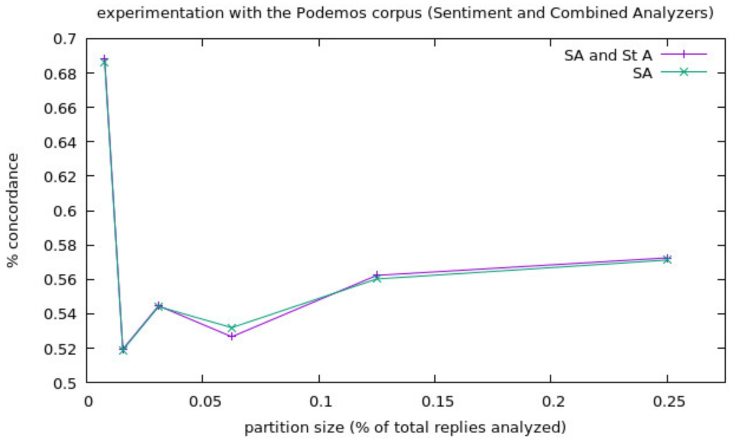

**Figure 2.** Results of the experiments with the corpora Podemos for the Sentiment Analyzer and the Combined Analyzer.

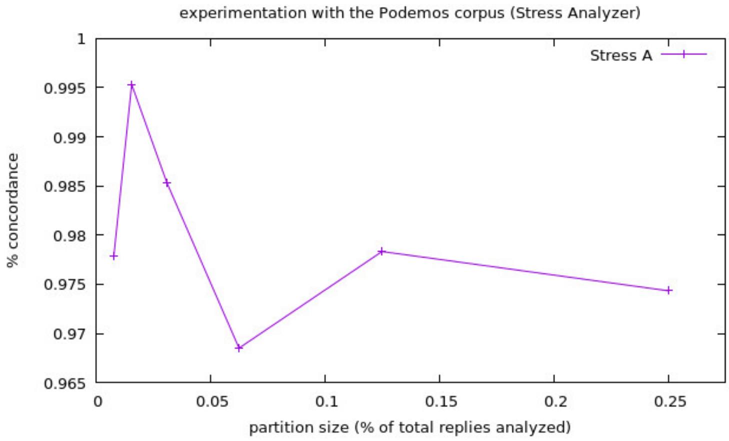

**Figure 3.** Results of the experiments with the corpora Podemos for the Stress Analyzer.

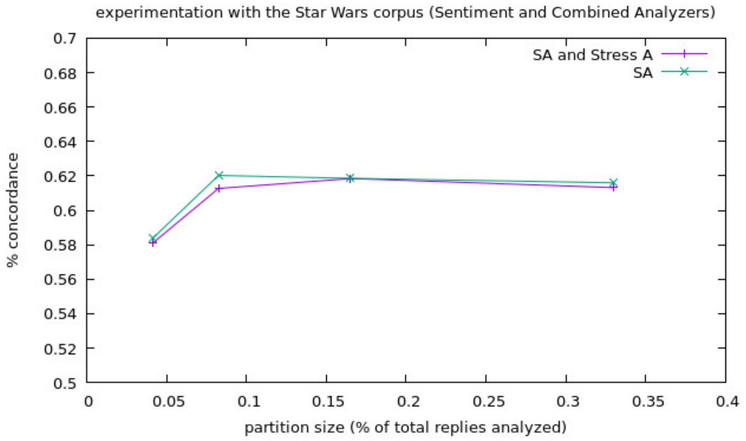

**Figure 4.** Results of the experiments with the corpora Star Wars for the Sentiment Analyzer and the Combined Analyzer.

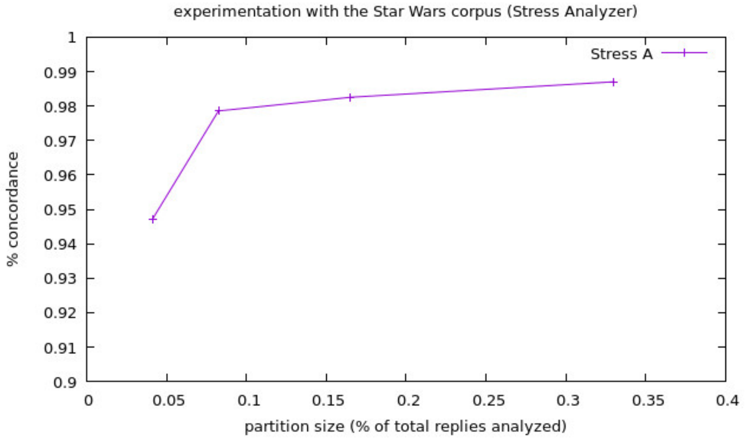

**Figure 5.** Results of the experiments with the corpora Star Wars for the Stress Analyzer.

## 3. Results

In this section, we will discuss the results of all the experimentation with the data from Twitter.com and the different Analyzers. In order to get rid of any ambiguity before we proceed with the analysis of the experimental results, we remind the reader that the Analyzers, and, thus, the system, predict the propagation of a sentiment or stress level in general (e.g., if the sentiment in the original tweet is positive, will this propagate to the replies? and, if negative, will it?). It does so in the experiments as well, where what we compare is the sentiment or stress level of the replies to the one in the original message, in order to know if this sentiment or stress level has propagated to the direct follow-up replies.

Then, the probability of a negative sentiment arising from positive and vice-versa, and all the combinations with different sentiment or stress levels, are just the inverse probability of the one calculated in the experiments, which is the probability of the propagation of the sentiment or stress level from the original to the replies (amount of times that the sentiment or stress level propagated/number of total comparisons).

### *Results of the Experimentation*

We discovered that all the analyzers separately (Sentiment, Stress and Combined) are successfully able to predict a bad outcome through bad emotional states, high levels of stress or negative combined value in the text messages. This can be seen in the experiments with all the corpuses. Regarding the Stress Analyzer, despite having a general tendency of high concordance with the replies (PCstr), we have to remember that it has a very small recall (RecallStr) so it may be less suitable than the other analyzers in a variety of cases.

In the case of the Podemos corpora, we can see that there is a big variation at the smallest sized experiments (1/128), but the results are considerably more stable at the big partition size experiments, starting to get more variation again when it comes to the biggest size experiment. This could be caused by the excessive amount of information when more and more replies are added to the analysis. As we can see in Figure 2, there is a general tendency (except for one case, which are the 1/16 partition size experiments), of the Combined Analyzer (PCcomb) to perform better than the Sentiment Analyzer alone (PCsen), from where we can conclude that at least at the domains where there is stress involved (such as politics in this case), the Combined Analyzer performs better than just the Sentiment Analyzer. We can see a general tendency of the Stress Analyzer to fluctuate in the 90–100% range of concordance (PCstr).

Regarding the case of the experiments with the Star Wars corpora, we can see a big change from 58–61% approximately for the Combined Analyzer, from 58–62% for the Sentiment Analyzer and from 94.7–97.8% for the Stress Analyzer, between the experiments with the smallest partition size (1/24), and the second smallest one (1/12), but stable results in the rest of the experiments. This is in line with the rest of the experimentation because of the number of replies analyzed. We can see that there is no clear difference between the Combined Analyzer and the Sentiment Analyzer, but the latter has shown to be slightly better or equal in all the experiments. This shows that, in the case of domains where stress is not normally present, the Stress Analyzer may only add noise to the results of the calculation of the user state that already considers the output of the Sentiment Analyzer; thus, the system must select to use the Combined Analyzer or not depending on the domain of application. The Stress Analyzer by itself continues with a general tendency to be in the 90–100% range of concordance.

## 4. Discussion

In this work, we have addressed the topic of Sentiment, Stress, and Combined Analysis in the Social Network domain, and we have discovered that sentiment, stress and a combination of both found in a written message are good indicators that this polarity, stress level or combined value will propagate to the future messages influenced by the current one. We discovered that the Combined Analysis works well at least in the domains where stress is present, slightly less well than the Sentiment Analysis alone in domains where stress is not normally present, like in the Star Wars corpora. For integrating the Analyzers as a helping or aiding system for users in an SNS, we designed an MAS that incorporates agents for the Sentiment and Stress Analysis and a novel Combined Analysis. This system will be able to analyze the sentiment polarity and the stress levels in the data that a user posts in an SNS, in order to perform a combination of the two mentioned analyses, and to decide whether to advise the user or not depending on those values and the concrete case.

For future lines of work, we will be performing a deeper analysis with more data, and we will aim to create new agents capable of performing new types of analyses (e.g., with other media like images, sounds, keystroke dynamics, etc), and discovering what domains and situations are more suitable for each of them for being able to help the user even better in his or her social experience. We will also integrate our MAS in an SNS and evaluate it with human users.

**Author Contributions:** G.A.S., V.J.I. and A. G.-F. conceived and designed the experiments; G.A.S. performed the experiments; G.A.S., V.J.I. and A. G.-F. analyzed the data; G.A.S., V.J.I. and A. G.-F. contributed with references, tools and technologies; G.A.S., V.J.I. and A. G.-F. wrote and revised the paper.

**Acknowledgments:** This work was supported by the project TIN2017-89156-R of the Spanish government.

**Conflicts of Interest:** The authors declare no conflict of interest. The founding sponsors had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, and in the decision to publish the results: the Spanish government.

## References

1. Vanderhoven, E.; Schellens, T.; Vanderlinde, R.; Valcke, M. Developing educational materials about risks on social network sites: A design based research approach. *Educ. Technol. Res. Dev.* **2016**, *64*, 459–480. [CrossRef]
2. Vanderhoven, E.; Schellens, T.; Valcke, M. Educating Teens about the Risks on Social Network Sites. An intervention study in Secondary Education. *Comunicar* **2014**, *22*, 123–131. [CrossRef]
3. Christofides, E.; Muise, A.; Desmarais, S. Risky Disclosures on Facebook: The Effect of Having a Bad Experience on Online Behavior. *J. Adolesc. Res.* **2012**, *27*, 714–731. [CrossRef]
4. George, J.M.; Dane, E. Affect, emotion, and decision-making. *Organ. Behav. Hum. Dec. Process.* **2016**, *136*, 47–55. [CrossRef]
5. Ciccarelli, M.; Griffiths, M.D.; Nigro, G.; Cosenza, M. Decision making, cognitive distortions and emotional distress: A comparison between pathological gamblers and healthy controls. *J. Behav. Ther. Exp. Psychiatry* **2017**, *54*, 204–210. [CrossRef] [PubMed]
6. Thelwall, M. TensiStrength: Stress and relaxation magnitude detection for social media texts. *Inf. Process. Manag.* **2017**, *53*, 106–121. [CrossRef]
7. Liu, B. Sentiment Analysis and Opinion Mining. *Ser. Synth. Lect. Hum. Lang. Technol.* **2012**, *5*, 1–167. [CrossRef]
8. Feldman, R. Techniques and Applications for Sentiment Analysis. *Commun. ACM* **2013**, *56*, 82–89. [CrossRef]
9. Schouten, K.; Frasincar, F. Survey on Aspect-Level Sentiment Analysis. *IEEE Trans. Knowl. Data Eng.* **2016**, *28*, 813–830. [CrossRef]
10. Jakob, N.; Gurevych, I. Extracting opinion targets in a singleand cross-domain setting with conditional random fields. In Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, Cambridge, MA, USA, 9–11 October 2010; pp. 1035–1045.
11. Hu, M.; Liu, B. Mining opinion features in customer reviews. In Proceedings of the 19th National Conference on Artifical Intelligence, San Jose, CA, USA, 25–29 July 2004; pp. 755–760.
12. Blei, D.M.; Ng, A.Y.; Jordan, M.I. Latent dirichlet allocation. *J. Mach. Learn. Res.* **2003**, *3*, 993–1022.
13. Rincon, J.A.; de la Prieta, F.; Zanardini, D.; Julian, V.; Carrascosa, C. Influencing over people with a social emotional model. *Neurocomputing* **2017**, *231*, 47–54. [CrossRef]
14. Gao, W.; Yoshinaga, N.; Kaji, N.; Kitsuregawa, M. Modeling user leniency and product popularity for sentiment classification. In Proceedings of the IJCNLP, Nagoya, Japan, 14–18 October 2013.
15. Seroussi, Y.; Zukerman, I.; Bohnert, F. Collaborative inference of sentiments from texts, User Model. In *Proceedings of the 18th International Conference on User Modeling, Adaptation and Personalization*; Springer: New York, NY, USA, 2010; pp. 195–206.
16. Nakamura, H.; Mise, S.; Mine, T. Personalized Recommendation for Public Transportation Using User Context. In Proceedings of the 5th IIAI International Congress on Advanced Applied Informatics (IIAI-AAI), Kumamoto, Japan, 10–14 July 2016; pp. 224–229.
17. Gregori, M.E.; Cámara, J.P.; Bada, G.A. A Jabber-based Multi-agent System Platform. In Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems, Hakodate, Japan, 8–12 May 2006; pp. 1282–1284.
18. Villena-Román, J.; Lana-Serrano, S.; Martínez-Cámara, E.; González-Cristóbal, J.C. TASS—Workshop on Sentiment Analysis at SEPLN. *Procesamiento del Lenguaje Natural* **2013**, *50*, 37–44.
19. Villena-Román, J.; García-Morera, J.; Lana-Serrano, S.; González-Cristóbal, J.C. TASS 2013—A Second Step in Reputation Analysis in Spanish. *Procesamiento del Lenguaje Natural* **2014**, *52*, 37–44.