IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 9, 2018, accepted October 13, 2018, date of publication October 23, 2018,
date of current version November 19, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2877685

Few Are as Good as Many: An Ontology-Based Tweet Spam Detection Approach

BAHIA HALAWI[1], **AZZAM MOURAD**[1], **(Senior Member, IEEE), HADI OTROK**[2,3], **(Senior Member, IEEE), AND ERNESTO DAMIANI**[2,3,4]

[1]Department of Computer science and Mathematics, Lebanese American University, Beirut 1102 2801, Lebanon
[2]Department of Electrical and Computer Engineering, Khalifa University, Abu Dhabi 127788, United Arab Emirates
[3]Center for Cyber-Physical Systems, Khalifa University, Abu Dhabi 127788, United Arab Emirates
[4]Department of Computer Science, Università degli Studi di Milano, 18-20133 Milano, Italy

Corresponding author: Azzam Mourad (azzam.mourad@lau.edu.lb)

This work was supported in part by the Lebanese American University, in part by Khalifa University, and in part by the EU-Funded Project TOREADOR under Contract H2020-688797.

ABSTRACT Due to the high popularity of Twitter, spammers tend to favor its use in spreading their commercial messages. In the context of detecting twitter spams, different statistical and behavioral analysis approaches were proposed. However, these techniques suffer from many limitations due to: 1) ongoing changes to Twitter's streaming API which constrains access to a user's list of followers/followees; 2) spammer's creativity in building diverse messages; 3) use of embedded links and new accounts; and 4) need for analyzing different characteristics about users without their consent. To address the aforementioned challenges, we propose a novel ontology-based approach for spam detection over Twitter during events by analyzing the relationship between ham user tweets versus spams. Our approach relies solely on public tweet messages while performing the analysis and classification tasks. In this context, ontologies are derived and used to generate a dictionary that validates real tweet messages from random topics. Similarity ratio among the dictionary and tweets is used to reflect the legitimacy of the messages. Experiments conducted on real tweet data illustrate that message-to-message techniques achieved a low detection rate compared with our ontology-based approach which outperforms them by approximately 200%, in addition to promising scalability for large data analysis.

INDEX TERMS Twitter, meta-data, spam detection, text based analysis, event spammers, ontology.

I. INTRODUCTION

Social media platforms are widely used by different age groups for many purposes, due to their compact messages and easy to use interfaces. However, the growth of the twitter market has caused this platform to be a target for commercial spammers from all over the world. Since Twitter currently has 319 million monthly active users, that translates to nearly 48 million bot accounts, using USC's high-end estimate [28]. Nonetheless, spammers tend to exploit trending hashtags by adding annoying messages and advertisements even if unrelated [15]. In general, this makes Twitter less credible for users, researchers and business practitioners, decreasing their trust in this platform and eventually generating less revenue for all actors of the Twitter ecosystem.

In the notion of identifying Twitter spammers, many approaches search accounts for suspicious profile indicators [5], [11], [15] abnormal behavioral patterns [3], [35], [38], [39] and sometimes non-legitimate tweet messages [23], [25], [36]. However, these techniques suffer from a set of limitations which are mainly:

- Restricted access to Twitter's API and metadata which causes many statistical approaches to become costly and unpractical.
- Unavailability of the followers data which makes working with the follower/followees impossible.
- Need for analyzing characteristic and relationship user data without users' previous consent.
- Ongoing changes in spammer's techniques and strategies while approaching Twitter users.

Therefore, there is a massive need to discover Twitter event spammers through publicly available tweet messages in order to minimize spammers' abilities to pollute content and downgrade Twitter's credibility. The aforementioned limitations, as well as the need for protecting user's personal

TABLE 1. Survey of spam detection approaches.

Paper	Methodology	Classification Type	Limitations
[22]	Social honeypots for harvesting deceptive spam profiles	Statistical & behavioral	Analyzing user private info & Twitter API changes
[7]	SVM classifiers	Statistical & content	Analyzing user private info & Twitter API changes
[41]	Random forests	Statistical & content	Use of spin bots & Twitter API changes
[15]	Profile feature analysis	Statistical & behavioral	No modeling relative to time
[11]	Statistical feature profiling	Statistical	Twitter API changes
[39]	Profile characteristics analysis	Statistical	Buying other spammer's interaction (retweets/favorites/etc.)
[5]	Content entropy and profile vector characteristics	Statistical & behavioral	Hashtag hijacking & embedded links
[37]	Using Wikipedia semantics to test tweet content	Content	Use of spinbots
[23]	URL analysis	Content	Embedded links and blacklists
[25]	Message type analysis	Content	Spammers retweeting one another
[40]	Double character analysis	Behavioral	Creative tweeting techniques & Spammers retweeting one another
[3]	URL analysis for lexical features & hosts & domains	Behavioral	Embedded links
[36]	User features and social network information	Content & behavioral	Changes in behavioral strategies
[4], [6], [10], [34], [38], [44]	Ontology based approaches	Content	Targets email spam and not short text messages
Our model	Ontology based approach	Content	Spammer's creativity in expressing new ideas

and relationship data, make the spam detection problem even more challenging.

Many models try to infer about spam through content analysis techniques, mainly integrated with other major approaches [23], [25], [36]. Through our experiments, several message to message approaches have been tested (cosine similarity, NLTK, and Co-occurrence). Results explored their inefficiency when it comes to detecting spam on real tweets discussing random topics. Moreover, ontologies have been widely used mainly for detecting spam in emails such as [4], [6], [10], [33], [37], and [43]. However, these techniques only address spam in long text messages, specifically emails. To the best of our knowledge, none of these techniques were used for inferring about spam in tweets where the size of the message does not exceed 140 characters.

In this paper, we propose a novel ontology based approach for the detection of suspicious content over Twitter during occasions or events where messages are compared to ontologies of different themes to validate the similarity between tweet texts and ontologies discussing related topics. The main contribution of this work is the development of a message to ontology evaluation approach that:

- Identifies spam tweets through content analysis
- Overcomes the need for relying on private and relationship based information in order to discover spam
- Reduces the need for a high similarity overlap while comparing tweets to ontologies by exploring the fact that few are as good as many terms, hence demonstrating the scalability of our approach for large data analysis

The performed experiments show that the proposed approach is able to outperform the detection rate of current existing content analysis techniques, which we have implemented along traditional statistical, behavioral and profiling approaches in spam detection models.

The remaining sections of this paper are organized as follows. In section two, the related works are examined. In section three, the system architecture and its components are described. In section four, the ontology based analysis model is carefully illustrated while in section five the probabilistic ontology evaluation model is presented. In section six, the experimental results and proposed ideas are emphasized along with a summary of the findings. Finally, the conclusion is presented in section seven.

II. RELATED WORKS

In this section, we overview the main approaches that address the topic of detecting spam over Twitter, which are classified in three categories: statistical, content, and behavioral. Moreover, we present the major ontology based techniques used for detecting spam in emails. For convenience, we will summarize and compare the major existing models in TABLE 1.

A. STATISTICAL INDICATORS ABOUT TWITTER USERS

Chen *et al.* [11] and Fazil and Abulaish [15] deploy different statistical modeling techniques for inferring about spammers. In [15], the statistical characteristics collected from 98 social bots are used for understanding the profile features of these bots such as such as age, gender and following activities. In [11], the statistical features of a tweet are studied relative to the time domain, assuming that topics can change over time, and thus proving the inefficiency of some machine learning classifiers in inferring about spam accounts. The authors thus propose an alternative approach, called Lfun scheme, where they can discover "changed" spam tweets and incorporate them into a classifier's training process [11]. Similarly, spammers are identified in CATS [5] through a combination of behavioral pattern analysis as well as profile-based traits one. The model pays attention to the ratio between followers and followees as well as similarities between tweets of the same person, in an intent to discover very wide divergences or exploitation for trending hashtags. These characteristics, just like the use of hashtags, the number of tweets and re-tweets submitted, as well as the use of hyper-links, can all be indicators to abnormal behavior over twitter, as emphasized by [42]. In this work, the conversation strategies are analyzed through the mentioned indicators, to assess the relationship between stakeholders and re-tweeters. Similarly, the proportion of content, qualifiers, or links tweeted relative to their linear changes over the analysis time frame is another indicator studied in [13]. Non-linear patterns are the main targets for indicating abnormal content dissemination in such scenarios.

Analyzing statistical attributes related to Twitter users is indeed beneficial for detecting abnormal characteristics. However, these approaches still suffer from many limitations since spammers can add non-realistic information about them in order to deceive other users. In addition to that, spammers can work in groups to support one another and gain credibility.

B. SHORT TEXT MESSAGE ANALYSIS

An important line of research in spam detection relies on analyzing the tweet content, as shown in [23] and [36] where suspicious use of hashtags or URLs is traced. The main objective in [36] is to study the semantics of short texts or messages in contrast with a set of Wikipedia text pages which are modeled and used as an aggregation of entities. The work presented in [23] stresses on the need for efficient URL detection schemes utilizing different features such as lexical ones and dynamic behaviors. For this purpose, a URL detection system for twitter, called WarningBird is presented. Furthermore, this system investigates the correlations of URL redirect chains [23], commonly practiced by different spam bots along with the frequency of redirecting. Makice [25] use statistical parameters about message content for deciding about legitimacy of Twitter users. They also explain how a language model is used in assessing the results along with a tracker for divergence among different language models.

The content of a tweet is analyzed for classifying it among spam or ham. Unfortunately, those solutions become less efficient because spammers often learn to react to many detection techniques. They can embed new links in their messages to avoid URL honey-pots while other spammers tend to use spinbots to reshape a certain phrase or idea.

C. BEHAVIORAL ANALYSIS

Other directions adopted in detecting Twitter spammers focus on discovering traits or patterns that best describe the spammer's behavioral profile. In such works like [39], the main contribution is to determine deceptive double characters for one user profile. This is done by analyzing non-verbal behavior variables as a function of time such as follows and retweets. Also, Sumner *et al.* [38] follow a similar technique. Direct approaches to checking up the user's portfolio include, but are not limited to, the notion of having no profile photo/biography/personal tweets or a suspiciously high/low number of followers/followees. In typical scenarios, a Twitter user is expected to have a reasonable ratio between people who she follows and people who follow her back. That's why, approaches within this scope search Twitter for suspicious profile characteristics or profile-based behavioral patterns. Examples of different profile-based behavioral analysis activities are demonstrated in [3] and [35].

Although analyzing user profiles is among the most trending techniques today, this approach becomes less efficient when spammers work in groups to support one another. This makes ratio calculations inaccurate. Similarly, it makes inferences relative to inactivity or lack of user-related attributes error prone. In addition to that, recent constrains placed over Twitter's API hinders access to many essential user-related metadata.

D. ONTOLOGY BASED APPROACHES

During the past few years, the number of email users increased dramatically, leading to the tracing of an unprecedented volume of 269 billion spam emails, according to [8]. As spammers always try to uncover a way to bypass existing filters, new techniques need to be developed. Balakumar and Vaidehi [6], Cao *et al.* [10], and Shahi *et al.* [34] proposed a tool to help detecting spam messages based on the semantics of their content. The idea behind this approach is to trace emails that claim to be about a topic but contain no term belonging to the vocabulary of that topic.

In the following, we review few approaches targeting spam detection in emails. Shoaib and Farooq [37] introduce the design of a system that uses ontologies to model features that are extracted from a user;s profile. The features are given to machine learning classifiers J48 and Naive Bayes that learn a user centric model of Good Spam or Bad Spam. In [43], two levels of ontology spam filters are implemented: a first level global ontology filter and a second level user-customized ontology filter. The user-customized ontology filter was created based on the specific user's background as well as the filtering mechanism used in the global ontology filter creation. In parallel, Alsmadi and Alhami [4] examine a large set of personal emails, from Gmail mainly, in order to properly classify subjects, targetting Arabic and English languages using k-means clustering algorithm [4].

The results of major ontology based approaches are very efficient when it comes to detecting spam in emails. However,

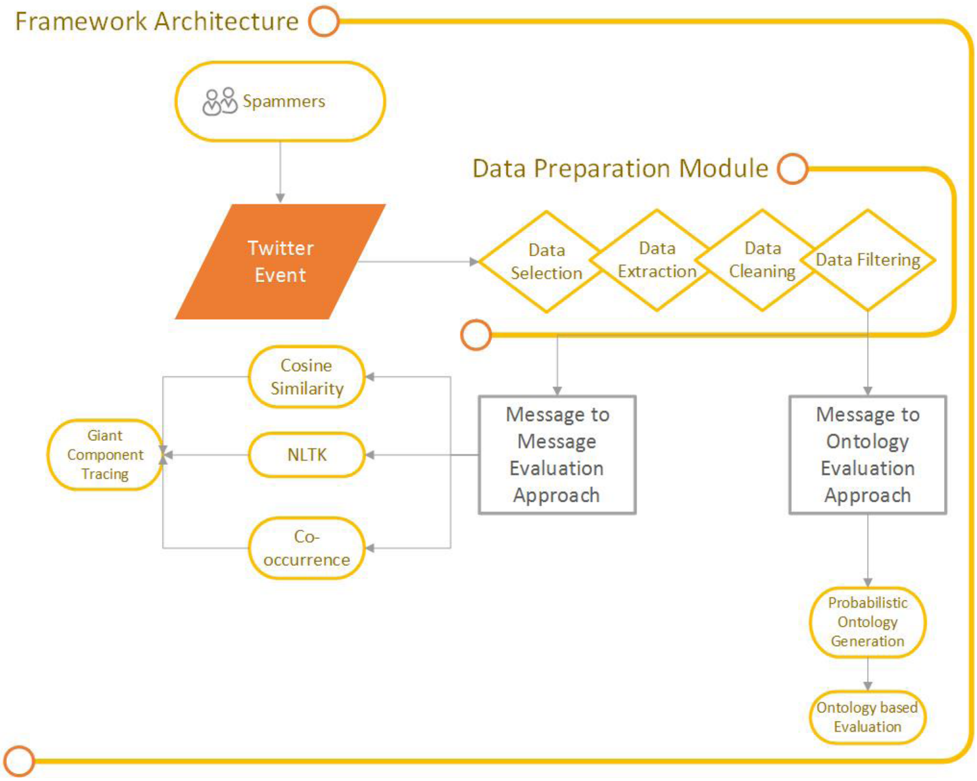


FIGURE 1. System architecture.

to the best of our knowledge, these approaches focus on the detection of spammers in emails and none of them targeted short messages, particularly Twitter. Our work will look more into the content sent by users rather than by their relationships with one another. In particular, we will study the divergence between content sent by legitimate users against content sent by spammers.

III. SYSTEM ARCHITECTURE AND DATA PREPARATION

Spammers today can make metamorphic representations to the same piece of content or information, making spam detectors confused. Also, spammers can create many accounts associated with different emails and spread the same ideas through different platforms. In parallel, it has become trendy for groups of spammers working together to bond and retweet one another to make their tracing harder. Finally, Twitter's new API has made using traditional approaches for spam detection like the follower/followee network analysis more costly. Here, we overcome those hurdles by elaborating a text based approach for spam detection in Twitter events. Our approach embeds a data preparation process where we deploy different data driven techniques from the data pipeline to extract raw data and prepare the datasets needed in testing as indicated in [41] where different steps for extracting and preprocessing tweets are explained before sending the cleaned tweets into the classification tool based on a latent Dirichlet allocation technique. Once tweets are ready, we run the second phase to inspect the validity of the deployed evaluation modules. Figure 1 illustrates the overall framework of our approach, which will be explained step-by-step in the next sections.

A. DATA PREPARATION MODULE

This unit is responsible for setting up the needed resources to run the rest of the computation. In this section, we discuss data preparation and address the need for scaling the system to handle big data.

1) DATA SELECTION

We use an online archive for tweets that dates back to a random collection of events and trends. These data sets are raw and unstructured. The time frame to which the data sets belong to ranges from 05-2013 to 08-2013 where the files are compressed in: [1].

2) DATA EXTRACTION

In this phase, we are concerned with the systematic collection of tweets that we can study and experiment with. We downloaded 4 tweet files from archive.org, structured them into tabular formats, cleaned the redundancies, and filtered them according to content and hashtags.

3) DATA CLEANING

In Twitter, each user can select the language of preference, through which the overall settings and display of Twitter will appear in. However, a user with Arabic settings, can still send English tweets. Therefore, we cannot rely on the preference settings as an attribute for inferring about user's tweet language. As a result, we had to design a script to extract only English tweets by studying the language of the tweet and identifying the highest similarity between its tokens and language axioms. In addition to that, we have paid attention to eliminating duplicate tweets by checking their keys and maintaining at most one instance of each.

4) DATA FILTERING

The collected tweets contain different topics and different themes as they are generated by random users under different hashtags and different time zones. We have a timeline of tweets that arranges tweets by the timing of their posting. Furthermore, our intent is to perform our experiments based on events, so we relied on hashtags to filter each set of tweets discussing a topic. Accordingly, we prepared the data sets that contain tweets where each group is clustered together according to the hashtags they mention and having the attributes of each tweet assigned to it. On the other hand, just like any language, English has a lot of stop words that pollute the tweet text when trying to analyze it along with its metadata. These stop words are common and found in any language with no significant meaning when presented solely [29]. In our experiment, we remove all the stop words and symbols as well as the # and 'http' links as the objective behind the simulations we conducted is to assess tweet legitimacy based on content only. Other techniques, can then be used to evaluate trustworthiness of http links. Hyper-links trustworthiness is a separate and large domain that is outside the scope of our paper yet the integration of http evaluation techniques with our approach can give even further improvements to the accuracy of the detection, which was emphasized in [2] when both content features and the hyperlink structure are used. Our list of stop words is minimal with only determiners (i.e. tend to mark nouns where a determiner will be usually followed by a noun) or determiners with prepositions (i.e. express temporal or spatial relations) or just coordinating conjunctions (i.e. connect words, phrases, and clauses) depending on the needs of the application [29]. Then, we create a vector from each tweet to use it in our analysis.

B. SPAM DETECTION APPROACHES

Detecting spam messages based on their content is not an easy task, especially when considering fragmented text, URLs and slang phrases. In this context, we have implemented three traditional statistical message to message models (cosine vector similarity, NLTK and co-occurrence models) in order to estimate their usefulness as mentioned in papers [5], [21], and [36]. Our experimental results prove that trying to identify spam based on message similarity yields unsatisfactory results. At first, experimenting with cosine vector similarity yields to around 25% correctness in optimal scenarios. Attempts to enhance the results through the NLTK model allowed a modest enhancement to 28% correctness rate. Later, the deployment of the co-occurrence model makes the accuracy close to 30% as depicted in figure 2. In the cosine vector similarity based experiments, we realized that people expressing the same ideas using different terms will eventually cause the results to be inaccurate. The search for similarity relies heavily on the existence of the same terms. Thus, different expressions of similar ideas will give a low similarity value for related tweets, making the false positive tweets number very high in the spam clustering process. Similarly, in the NLTK based approach, topic trees and semantic distances didn't give accurate results either. The reason is that people tend to be brief over Twitter, using abbreviations and special terms that are not adopted as they are from the English dictionaries as well as other terms like links and emoji icons. That's why, we tried to use a co-occurrence based model that would increase the similarity between terms based on their co-mentioning in the same tweet text, yet high similarity in this scenario is only attained when people are discussing the same ideas with previously mentioned terms. Discussing new ideas under the same theme will give a low similarity rate. To enhance the accuracy of the message to message detection approaches, we have elaborated in this work an ontology based evaluation technique that analyzes tweet text messages for detecting spam. The technical details as well as the experimental results and findings are further discussed in the next sections. It is important to mention also that after performing the calculations in each experiment, we had attempts to trace the giant component, with the hope of eliminating outliers to a certain topic or context of speech, indicating the presence of spam. However, unlike the high correctness of the giant component tracing for the network of followers and followees, the content clustering based on cosine vector similarity, NLTK and co-occurrence does not yield to acceptable results. Now to benchmark our results, we have referred to paper [21], where clustering of tweets is attempted based on cosine similarity scores, yielding to satisfactory results, as compared to other clustering algorithms like DBSCAN and K-means where the results were unacceptable with very high rates of false positive reaching around 83%. However, the acceptance level attained from the cosine-based similarity experiments, as reflected in the results of this paper, seem to cluster tweets that only have

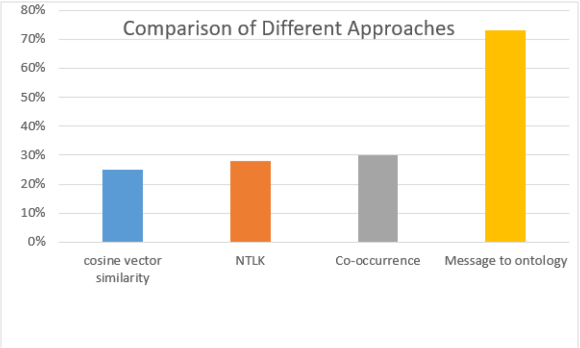


FIGURE 2. Comparison of different approaches at their ideal state.

common terms or tokens very efficiently. Yet the authors discuss the need for extending the work to analyze synonyms to overcome the static term problem.

IV. MESSAGE TO ONTOLOGY EVALUATION APPROACH

An ontology is an implicit representation of knowledge through a set of concepts and relationships that assist in understanding a field. An example of the ontology's representation is illustrated in Figure 3 where a set of hierarchal interconnected concepts are built. We propose in this section a new model that relies mainly on comparing tweet messages against ontologies in order to infer about spam. The idea is that ontologies extracted about a certain theme or topic should contain a large segment of terms that cover the studied topic. These terms are what people who discuss sub-topics in this theme use or mention. By generating a dictionary of such terms and traversing them token by token, we seek to spot the same terms in the tweet text. Optimally, the witnessing of one or more terms in the tweet text among dictionary terms gives the tweet more credibility and less likelihood of being categorized as spam. Differences among topics and hashtags are an additional expectation from the technique.

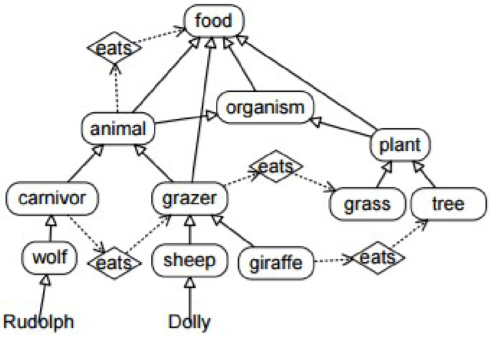


FIGURE 3. Ontology structure.

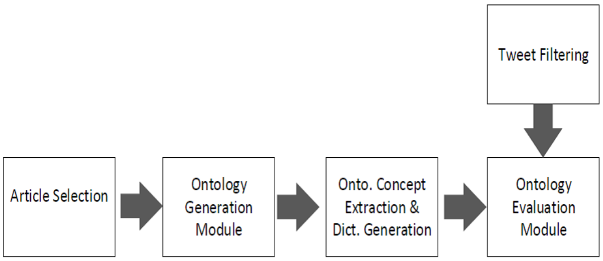

Tweet Filtering
Article Selection
Ontology Generation Module
Onto. Concept Extraction & Dict. Generation
Ontology Evaluation Module

FIGURE 4. Ontology based analysis module.

Our ontology generation models adopt a common direction, including the following steps: (1) Domain terminology extraction, (2) Concept discovery, (3) Non-taxonomic relations learning, (4) Rule discovery, (5) Ontology population, (6) Concept hierarchy extension and (7) Frame and event detection. Figure 4 outlines the architecture of our proposed approach. The ontology generation module acts as a container for the steps mentioned above. Thus, they allow the generation of an ontology based on any theme or genre. For this sake, it is mandatory to start with a set of textual files. After selecting a set of articles, they are cleaned from different figures and are then inserted into the model. At this stage, users can interact and select the intended objectives such extracting concepts. The model systematically performs the steps described above in order to extract and populate the ontology along with all the relationships between the corpus elements. After collecting the list of concepts in that ontology, we transform it into an array of terms that acts as a dictionary and is used in comparing the tweets against. This takes place in the evaluation phase, where tweets are modeled as sentences composed of tokens. The terms from the tweet are traversed and compared against the terms from the dictionary thus yielding to an evaluation indicator that suggests the likelihood of being a spam tweet to that particular topic/discussion. In the sequel, we illustrate the technical details of every step in the model presented in Figure 4.

A. ARTICLE SELECTION

In order to extract the ontologies associated with each theme, we feed the ontology generation platform with textual documents that contain the most commonly used terms, hence composing the theme's taxonomy. We used documents with minimal intersections for the sake of covering the widest range of terms or ideas in the ontology extraction process. As we are working with limited resources, we only used few documents/articles to achieve this target.

B. ONTOLOGY GENERATION MODULE

After selecting the articles, we send those discussing the same theme or topic to our ontology generation module. In this step, the objective is to extract (1) the linguistic procedure and (2) the adjustments through the statistical ones. The technical details of the probabilistic ontology model and algorithms are presented in chapter 5. The corresponding algorithms are executed to identify all inheritance relationships. At this point, we have the list of attributes being generated. We also obtained the concepts and entities to be used in the comparison later on. These concepts are the main terms or key nouns that can be found in any textual piece that addresses a topic or a subtopic.

C. ONTOLOGY CONCEPT EXTRACTION AND DICTIONARY GENERATION

The ontology generation module, used during the extraction of the ontologies, generates the list of concepts from articles belonging to different themes. We have used the politics, soccer and technology topics in this paper in order to illustrate our proposition, as presented in figure 4. Moreover, the model is fed with articles belonging to the relative domain. For instance the generation of a politics ontology involves only politics articles. So the generation of a politics ontology involved only politics articles. The generated ontology, which contains the concepts we are relying on, is to be used as the test benchmark against tweets discussing a politics hashtag.

D. TWEET FILTERING

In the validation approach adopted by the ontology based algorithm, we seek terms or nouns in the tweet that are mentioned in the ontology list of concepts. A spammer can exploit this by tweeting about an irrelevant topic and just mentioning one or more valid hashtags. In this way, the spam content is equivocal to the algorithm, as it is able to overcome it by achieving the minimal relevancy required by containing that term or hashtag. As we are interested in measuring content relevance against the overall content related to that theme, we disregard hashtags. Therefore, tweets composed of hashtags only are treated as a spam to prevent users infusing meaningless hashtags from overcoming the algorithm. In other cases, users of our model might be interested in measuring the frequency of tweets to gather insights regarding various matters like supporting sports teams in a certain zone or assessing voters from different locations. Such scenarios entail accepting hashtags and weighing their presence in a tweet to take it into consideration while comparing against the ontology concepts. Even if the tweet contains no actual content or new messages, tweeting a certain hashtag is a must for cumulatively summing up users' counts and mentions. Therefore, inclusion of hashtags in the evaluation process becomes relative to the scenario of use. Sentiment analysis projects that aim to reflect opinions and accurate emotions should completely overlook tweets made up of hashtags while number evaluations for human activity prediction such as political support can be inferred through such tweets.

E. ONTOLOGY EVALUATION MODULE

The ontology evaluation module is the core component in our ontology based tweet spam detection approach. In this phase, we use the extracted concepts from the generated ontology as white lists or dictionaries to test for content similarity with the tweets. After the clustering of random tweets according to topics and depending on their hashtags, the hashtags are removed according to the tweet filtering mechanism explained in section 4.4. Then, we scan through the tweet tokens to find the terms that match with our white lists. The more similar tokens we detect, the higher is the credibility of the tweet. Figure 5 illustrates the three categories we used in order to demonstrate the usability our approach. For instance, the technology ontology is used for evaluating the technology tweets. Similarly, each ontology is used while testing the tweets that discuss the same topic in order to make the evaluation process more accurate and topic tailored. In our platform, communication among different phases across the evaluation model is done automatically. We have achieved that using scripts designed to structure input and output so that results can be inserted into various segments of the model.

V. PROBABILISTIC ONTOLOGY MODEL AND ALGORITHMS

In this section, we present the probabilistic ontology framework implemented within our model. Also, we emphasize its two major components, the probabilistic extraction phase and the ontology generation phase, along with the algorithms deployed in each of them.

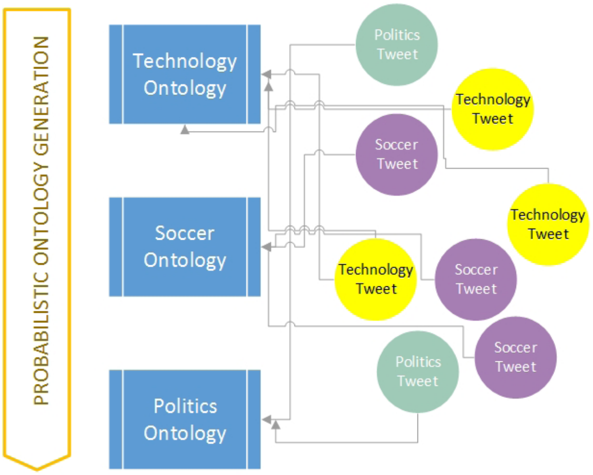


FIGURE 5. Overall ontology based evaluation model.

A. PROBABILISTIC ONTOLOGY EXTRACTION ALGORITHMS

Each algorithm used in the ontology generation module is needed to generate or assist in the generation of a particular modeling primitive [14]. It is also important to note that these tools contain private libraries which produce declarative primitives, thus providing extensibility, flexibility and translation abilities. The modeling primitives are the following:

- Concepts (class)
- Concept inheritance (subclass of)
- Concept instantiation (instance of)
- Properties or relations (relation)
- Domain and range restrictions (domain/ range)
- Mereological relations
- Equivalence

1) CONCEPTS (CLASS)

Concepts or classes are an assessment for the relevance of a certain term with respect to the corpus in question. In order to perform this, three logical phases, Relative Term Frequency (RTF), TFIDF (Term Frequency Inverted Document Frequency) and Entropy and C-value/NC-value methods are implemented.

2) CONCEPT INHERITANCE (SUBCLASS-OF)

In the concept inheritance class, we made use of the hypernym structure of Word Net and Hearst patterns as well as linguistic heuristics to trace sub-class of relationships.

3) CONCEPT INSTANTIATION (INSTANCE-OF)

In the similarity based approach, the algorithm extracts context vectors for instances and concepts from the text collection and assigns instances to the concept corresponding to the vector with the highest similarity. Mere logical relations are among the relations examined in the implementation.

4) MEREOLOGICAL RELATIONS

Part of relations are the main focus here in the exploration. The algorithm here counts the occurrences of certain patterns that assist in the identification of the part of relation among terms. After that, the probability of collecting this value is also validated. Word Net is used for comparing the results and highlighting major differences. In particular, JAPE expressions, which are rules and regulations relative to a particular language are used for the purpose of discovering mereological (part-of) relations. This is done through an algorithm that counts the occurrences of sequences that reflect a part-of relation between any two terms.

5) GENERAL RELATIONS

In order to extract relations across textual data, subcategories as well as frequencies and arguments related to transitive, intransitive and complement sentence structures are emphasized.

6) EQUIVALENCE RELATIONSHIPS

In order to feature equivalence relationships, we use the intuition where terms or concepts are equivalent to the extent to which they share similar syntactic contexts. The algorithm thus mainly focuses on contextual features derived from the language axioms. Values generated are later on used as the probability for the equivalence of the concepts in question.

B. PROBABILISTIC GENERATION OF ONTOLOGIES

The probabilistic model assures that ontologies are attached along the generated results to allow the tracking of changes in the attained corpus. Moreover, we map the discovered variations incrementally into the probabilistic ontology model, rather than doing it from scratch. These changes can be easily noticed and analyzed over time. While trying to extract an ontology, different tools tend to adopt either the machine learning techniques or linguistic ones. On the contrary, the use of a probabilistic approach helps in modeling primitives rather than in a concrete knowledge representation language. To achieve this, a controller is core to the adopted architecture, supporting in the relative initialization of different algorithms, which are responsible for processing data, learning orders and applying the probabilistic model. Each algorithm passes through three execution phases: The notification process where changes are tracked and then the computation phase where witnessed changes are mapped to the generated knowledge. Finally, in the result generation phase, the corpus gets finalized and the probabilistic model gets updated. Our probabilistic ontology model consists of a set of modeling primitives, regardless of the ontology representation language being used such as OWL, RDFS and F-Logic. The probabilistic ontology model acts as a bag containing learnt elements. Here, probabilities are deployed in order to enhance results, allowing a more precise decision on the inclusion or exclusion of a certain object. A modeling primitive library is deployed in order to allow for defining new primitives in a declarative fashion. As a result, knowledge is easy to get described and represented. These modeling primitives allow for the translation of any type of knowledge needed. Each sentence is associated with a probability relative to its entities. The statement can exist with a probability that is calculated based on the following formula:

$$P(S^{(m)};\theta) = \frac{exp(\theta^T f(e^{(m)}, t^{(m)}))}{\sum_e \exp(\theta^T f(e, t^{(m)}))} \tag{1}$$

where $P(S^{(m)};\theta)$ is the probability for each sentence, θ is the log likelihood of a corpus D in this ontology, s is the sentence represented as a parse tree and t is a unary pattern. Here $e^{(m)} = (e^{(m1)}, \ldots, e^{(mn)})$ is a vector of entities. Different entities are looked at as a categorical random variable which has a domain as all the noun phrases (PNPs and CNPs) in the corpus. Through the probabilistic ontology model, the results of the system are associated with the relative probabilities. This is a collection of instantiated modeling primitives which are independent of a concrete ontology representation language. In the sequel, we present the remaining two underlying algorithms that are used for completing the generation phase of the probabilistic ontology model: data driven discovery and natural language processing.

1) DATA DRIVEN DISCOVERY

The main objective in data driven discovery is to actually build up implicit specifications by analyzing the ontology variations across data. Initially, three different approaches to discovering changes can be outlined : (i) structure-driven, (ii) usage driven, and (iii) data driven. The data driven method to discovery is used in our model as it is highly connected to the underlying data or text. So changes are expected once modification to texts occur. Moreover, change strategies are also tracked, helping thus in measuring influence across that ontology. This takes place prior to formally mapping out knowledge diagnosed about concepts, instances, and relations as well as knowledge about how these aspects change as depicted in Figure 6. Implicit mandatory points are calculated here, allowing for bottom up modifications in behaviors and respectively in the model used for discovery. This model is specifically crucial for tracing all changes and modifications taking place and mapping it into the whole mathematical system being calculated. Figure 7 illustrates the logic functions.

2) NATURAL LANGUAGE PROCESSING

We extend the flexibility of the Gate framework (https://gate.ac.uk/) in running new linguistic algorithms along with annotating the results through regular expressions. Before initially running any algorithm, we process files, tokenize them and separate sentences from one another. Later, the tagger places the terms in the suitable category. In parallel, a morphological analyzer is used to lemmatize the text and after that the stemmer is used to stem them respectively. At that stage, the textual material becomes ready to be used. A Jape transducer is responsible for matching patterns across different

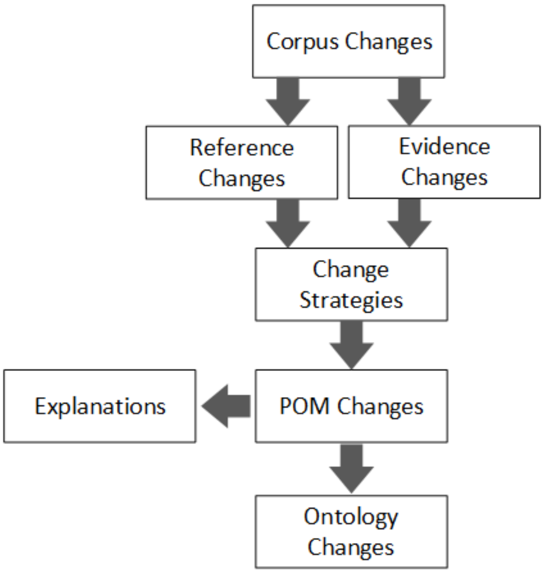


FIGURE 6. Change management high-level architecture.

```
Rule: NounPhrase (
 (
  {Token.category == DT}
  {SpaceToken.kind == space}
 )?
 (
  ({Token.category == JJ}|{Token.category == VBG})
  {SpaceToken.kind == space}
 )*
 (
  (
   ({Token.category == NN} | {Token.category == NNS})
   {SpaceToken.kind == space}
  )*
  ({Token.category == NN} | {Token.category == NNS})
 ):np_head
):np
-->
:np.NounPhrase = { rule = "NounPhrase" },
:np_head.Head = { rule = "NounPhrase" }
```

FIGURE 7. Rule discovery logic.

ontology learning algorithms. For the sake of fulfilling the following steps, we use text2onto, which is an open source key technology [14] for semantics-driven modeling, mainly supporting users in order to construct ontologies from a given set of textual data. We will use text2onto because it helps us in combating a set of problems that alternative tools suffer from, mainly the flexibility in collecting modeling primitives instead of just representing knowledge bases semantically according to a significant number of scientific researchers.

VI. IMPLEMENTATION AND EXPERIMENTAL RESULTS

In this section, we present the experiments completed over three main themes of tweets which are sports, technology and politics. We first run the experiments, clarify findings and discuss the related insights.

The objectives of the experiments we conducted are:

1) Evaluate the performance of the ontology based algorithm implemented in differentiating between spam tweets against relevant ones.
2) Compare the correctness of the results upon changing the similarity token threshold values.
3) Infer about the relationship between the theme of the tweet and the threshold selected for the comparison.

For the sake of fulfilling the mentioned experiments, we have used an HP computer with the following specifications: Intel Core I5 2.3Ghz, 8GB Ram and a 5400 rpm hard disk. In order to assess the behavior of the ontology model over different topics and relative to varied token similarity threshold values, we conduct a set of independent experiments. Table 2 outlines the different data sets used for this purpose, including the size of each, the theme, and the abbreviation used in the figures. The threshold value represents the minimal similarity accepted in validating the legitimacy of a tweet. Each tweet is cleaned against stop words and irrelevant terms are disregarded from its context. Then, tweets are iteratively tokenized. A threshold here represents the percentage of words/tokens needed minimally to accept a tweet into the legitimate category. A 0.1 threshold for instance, mandates the existence of 10% of the tweet tokens among the words in the respective ontology being used for the comparison. As the threshold increases, more tokens become required for accepting the tweets into the legitimate (i.e. not spam) category. Six different thresholds are used in testing, ranging from 0.05 to 0.5. As the domain of values is relatively small, we have not tried to set it based on background information. Rather, we have tried different values to find the impact and relevance of each threshold. We determined the threshold to work with based on the impact of threshold selection on the correctness of the results collected in evaluating spam legitimacy. Note that when the threshold is below 0.05, it is as if we are accepting any tweet and treating it as legitimate content. In other terms, in this case we are not mandating the presence of any similarity between legitimate tweets tokens and tokens of non-legitimate ones. That is why we haven't tested values below 0.05 as they will validate any tweet as legitimate making their evaluation useless. To evaluate the effect of selecting the right threshold relative to the topic being evaluated and the tweets being processed, the same

TABLE 2. Datasets used in the experiments.

Name of Tweet set	**Theme**	**Size**
Basketball	Sports	253
Tech Event	Technology	254
Politics	Politics	297
Politics 1	Politics	253
Politics 2	Politics	297
Soccer	Sports	115
Strata Event 1	Technology	392
Strata Event 2	Technology	385
Spam	Spam tweets	35

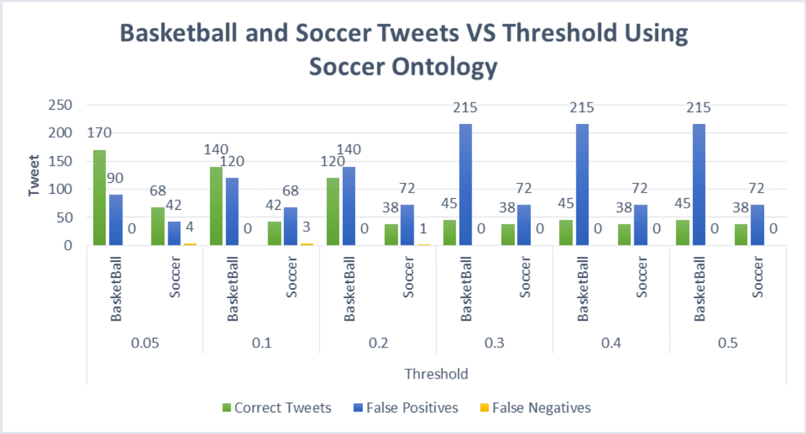


FIGURE 8. Basketball and Soccer tweets vs. Threshold using a soccer ontology.

sets of tweets will be used in the execution. Changing the threshold is compulsory for tracing changes in evaluation patterns. The displayed results will imply the benefits and drawbacks for adopting each threshold. Thus, users of the proposed model will have the ability to decide on the best threshold relevant to the scenario and flexibility in tolerating spam. Strictness in detecting spam tweets will lead to compromises at the level of false positives being detected and therefore over all correctness.

Subsections 6.1, 6.2 and 6.3 are divided into three sub-subsections as follows:

- Accuracy of classifying tweets: We present the results of testing the ontology approach against a set of random tweets relative to different token similarity threshold values. Findings will reflect the approach accuracy in terms of classifying tweets among correct, false positive and false negative.
- Efficiency in recognizing spam tweets: We present the results of testing the ontology approach against a set of spam tweets relative to different token similarity threshold values. Findings will reflect the approach accuracy in terms of detecting different types/patterns of spam.
- Discussion: We present concluding remarks and comparisons across the different performed experiments.

A. EVALUATING A SET OF BASKETBALL AND SOCCER TWEETS AGAINST A SOCCER ONTOLOGY

1) ACCURACY OF CLASSIFYING TWEETS

Fig 8 represents the results of testing a set of basketball tweets (from random NBA games) against a soccer ontology. Six different thresholds (0.05, 0.1, 0.2, 0.3, 0.4, and 0.5) are used for experimenting in order to measure the impact of modifying the similarity threshold on the classification accuracy. By analyzing the above results, we notice that our approach is reaching a 63% classification accuracy. We also notice that a lower threshold (0.05 and 0.1) yields to more accuracy in terms of detecting spam content among tweets. For instance, at a threshold 0.05, the number of correctly recognized tweets is 170 while the number of false positive tweets is 90. When the threshold is increased to 0.1, the number of correctly inferred tweets decreases to 140 while the number of false positives increases to 120. The results are relatively better when it comes to false positives that seem to increase as the threshold increases. Another observation is that after the third threshold (0.2), the results seem to converge. Although result accuracy decreases, more false positives are being traced, and the results among the final three thresholds are exactly the same. After the third threshold (0.2), the algorithm returns a spam indicator for majority of tweets being executed.

Moreover, we tested a set of soccer tweets against the same soccer ontology. Tweets of this data set discuss a soccer game and conversations for tweeters about it. While observing the scores of Figure 8 more thoroughly, we realize that the lower a threshold, the more accurate are the results. At a threshold of 0.05, the percentage of correctly evaluated tweets exceeds 66% whereas the rate decreases to 40% at a threshold of 0.1. Larger thresholds yield to relatively lower correctness results (around 33%). Moreover, the results after the third threshold (0.2) remain the same but with a high false positive rate (67%).

2) EFFICIENCY IN RECOGNIZING SPAM TWEETS

In order to measure the effect of changing the threshold on spam recognition only, we repeat the execution of the same ontology based technique for the same six thresholds. In Fig. 9, we observe that at a lower threshold (0.05 and 0.1), in general, yields a higher result accuracy yet spam recognition is less efficient using those thresholds. As the threshold is increased, the accuracy of recognizing spam only becomes more efficient than detecting it at a relatively higher threshold (0.3 and above). As the threshold seems to increase, the false negatives become nonexistent and the detection of spam tweets becomes complete.

3) DISCUSSION

While experimenting with different sets of sports tweets against a soccer ontology, we realize that the ontology based

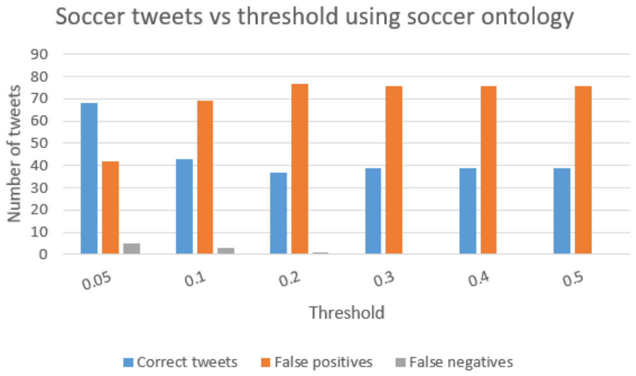


FIGURE 9. Results of experimenting with spam tweets against a soccer ontology.

technique is powerful in detecting legitimate tweets and correctly classifying it. Moreover, we notice that, the lower a threshold, the more flexible is the detection, meaning that the algorithm allows in more suspected tweets for the sake of not considering legitimate tweets as spam. At the lower thresholds we witness higher overall efficiency in segmenting different types of tweets but minimal accuracy in correctly recognizing spam tweets. As the threshold increases, we can trace the trade off, overall correctness eventually decreases but effectiveness in realizing various spam patterns increases to become ideal. The optimal threshold therefore lies at an intermediate level, most probably around 0.2, where the false positive rate is acceptable and spam detection is accurate. More experiments will be needed however, on different topics, to validate these findings. Another conclusion is that ontology based approaches preserve a minimal ability in detecting spam tweets, even when tweets do not belong to the same theme which the ontology being used for the comparison discusses. The third conclusion in this scope is the fact that results are being stable for all high thresholds. In particular, the last three thresholds (0.3, 0.4, and 0.5) have converging results. This becomes of large importance when handling large data sets as we can reduce the effort needed in checking for similarity. Rather than checking for 50% similarity to accept a tweet, it is feasible to check for 40% and even 30% similarity.

B. EVALUATING A SET OF TECHNOLOGY TWEETS AGAINST A TECHNOLOGY ONTOLOGY

1) ACCURACY OF CLASSIFYING TWEETS

Figure 10 depicts the results obtained when running the ontology based algorithm against a random group of technology tweets discussing some technology events. Again, the same thresholds (0.05, 0.1, 0.2, 0.3, 0.4, and 0.5) are used in implementing the results, giving us the ability to compare among the efficiencies of each. By having a closer look at the obtained numbers, we realize that lower thresholds (0.05 and 0.1) are better in detecting spam content, compared to higher thresholds. If we take the first two thresholds for instance (0.05 and 0.1), the lower threshold among both allows tracing 160 correct tweets and 80 false positive ones (around 62% of accuracy) while the higher threshold returns 155 correct tweets and 85 false positive ones. False positives obviously increase with the increase in threshold. Moreover, results obtained at the third threshold (0.2) and above (0.3, 0.4 and 0.5) are almost the same. On the other hand, accuracy declines after the third threshold as more false positives are noticed. Just like the previous model where we compared spam tweets against the soccer ontology, after the third threshold (0.2) the results are exactly the same and the algorithm returns a spam answer for almost all tweets being checked.

2) EFFICIENCY IN RECOGNIZING SPAM TWEETS

Figure 11 illustrates the variations of the result efficiency in terms of identifying spam tweets. When running the set of spam tweets that contains different spam patterns, changing the threshold produces different results. The ontology based method is executed against the same six thresholds used in the rest of the experiments. Results reveal that lower thresholds (0.05 and 0.1) cause the algorithm to miss some spam tweets

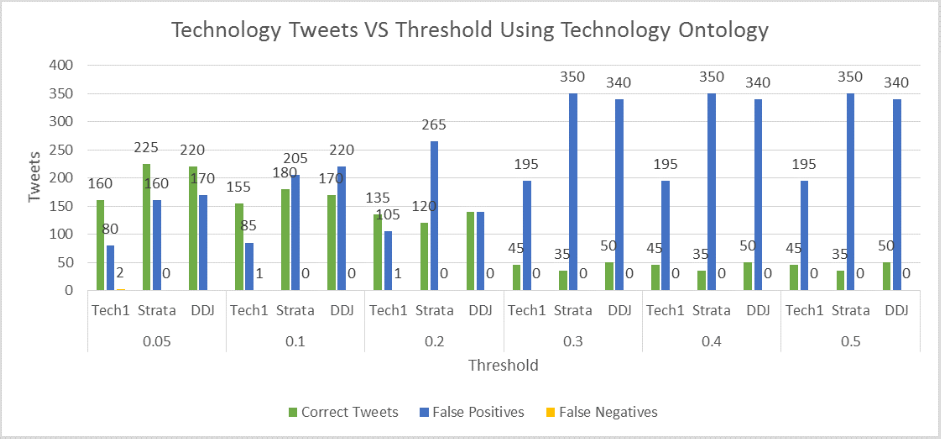


FIGURE 10. Results of experimenting with technology tweets against a Technology Ontology.

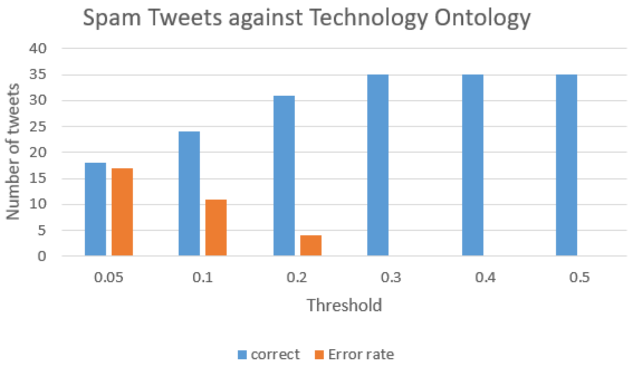


FIGURE 11. Results of experimenting with spam tweets against a technology ontology.

while higher thresholds (0.2 and above) reflect more accuracy when it comes to recognizing spam tweets. At the false negatives level, a higher threshold has a better impact on the accuracy of handling this type.

Another independent execution of the ontology based approach using a set of tweets that discuss strata conference event against a technology ontology is completed in Fig.10. When the threshold is small (0.05), the detection accuracy exceeds 57% over a set of almost 400 tweets. As the threshold increases, the accuracy decreases gradually to reach 30% after the third threshold (0.2). Results after that converge and the accuracy is very low for all three thresholds (0.3, 0.4, and 0.5).

A similar execution of technology related tweets are run using the same thresholds and results match with the previous experiments. The lower a threshold, the higher a correctness rate and the increase in threshold yields to higher false positive rates. Stability of results is achieved after a 0.2 threshold.

3) DISCUSSION

Just like the findings of the sports based experiments, results reassure the conclusion that lower thresholds are better in the overall assessment of results but are less accurate in tracing spam tweets. On the other hand, larger thresholds become strict, classifying legitimate tweets as spam ones and thus the overall results decline. The compromise among both suggests using an intermediate threshold, and setting a lower threshold depending on the theme and nature of tweets. In parallel, eliminating the need for checking for 50% similarity by checking for 30% only is also verified in these experiments, particularly, as the experiments prove that results among the last three thresholds converge.

C. EVALUATING A SET OF POLITICS TWEETS AGAINST A POLITICS ONTOLOGY

1) ACCURACY OF CLASSIFYING TWEETS

In Figure 12, a politics ontology is used to test a group of tweets that discuss different election topics. The smaller thresholds (0.05 and 0.1) have higher correctness rates with smaller rates of false positive tweets being labeled. On the contrary, the false positive rates increase as the threshold increases. At a threshold of 0.05 for instance, 199 correct tweets were recognized while 50 were false positives and only 1 false negative (accuracy rate exceeds 70%). As the threshold is increased to 0.2, the number of correctly classified tweets decreases to 150 while the number of false positive tweets increases to 139. After the third threshold (0.2), we notice that the results look the same for the false positives and correct tweets tested.

2) EFFICIENCY IN RECOGNIZING SPAM TWEETS

Figure 13 reflects the results collected upon testing the ontology approach against a group of spam tweets that

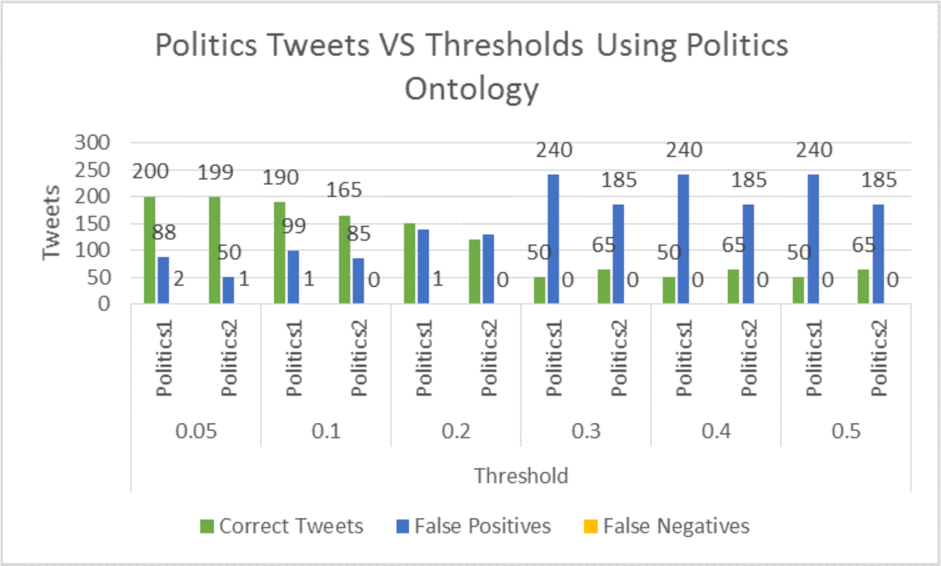


FIGURE 12. Results of experimenting with politics tweets against a politics ontology.

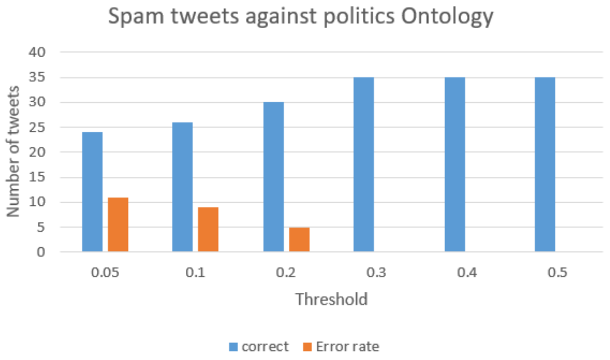


FIGURE 13. Results of experimenting with spam tweets against a politics ontology.

contain different patterns used by spammers. Low thresholds (between 0.05 and 0.1) have less accuracy when differentiating between spam tweets and legitimate ones. At a threshold of 0.05, the number of correctly recognized spam tweets is 25 while 10 spam tweets are not recognized. Once we increase the threshold to 0.1, 29 spam tweets are recognized and at a threshold of 0.2, 32 out of 35 spam tweets are detected. Optimal results are obtained after the third threshold (0.2) where all the spam tweets get collected. At lower thresholds, accuracy seems to increase as the threshold increases.

3) DISCUSSION

Experiments run with politics tweets are very insightful for reassuring the previous two conclusions. In terms of selecting the best thresholds, which seems to be 0.2, experimenting with different topics returns similar output. This threshold has the most logical trade off among the six tested thresholds. It provides us with the ability to recognize spam while not getting too strict in falsely labeling legitimate tweets as spam. Moreover, as conversations in politics tweets look the closet to regular sentences (as compared to technology and sports), the overall accuracy of the politics experiment are the highest. Again, results prove that we do not have to check for at least 50% similarity between the tweet and the ontology being used. Thus, this can reduce our effort in checking actually for 30% similarity only across the last three thresholds that carry the same results over all tweet themes.

D. EFFECT OF CHANGING THE NUMBER OF ARTICLES USED IN EXTRACTING THE ONTOLOGY

Figure 14 illustrates the variation of result accuracy as we manipulate the size of the ontology being used for extraction, and eventually using the extracted concepts for comparing against tweet tokens. As we have explained before, the comparison we perform is against a set of concepts that belong to an ontology. This ontology is extracted by traversing a set of articles that discuss a common theme. In the first experiment, 3 articles are used for the extraction of concepts. Correctness rate starts around 130, 150 and 180 for a threshold of 0.05 for 3, 5, and 6 articles respectively. Result accuracy continues to decrease over all three experiments as we increase the threshold. In the experiment whose ontology was extracted through 3 articles, the rates are less accurate relative to the other two experiments over all thresholds. We do witness an enhancement in results as the number of articles increases. Ideally, the correctness rate attained while relying on 3 articles during the ontology extraction phase is around 50%, it rises to 55% as we increase the number of articles to 5 and it approaches 70% as we use 6 articles instead. Thus, in the most optimal cases, the marginal change can reach 15% . However, exact

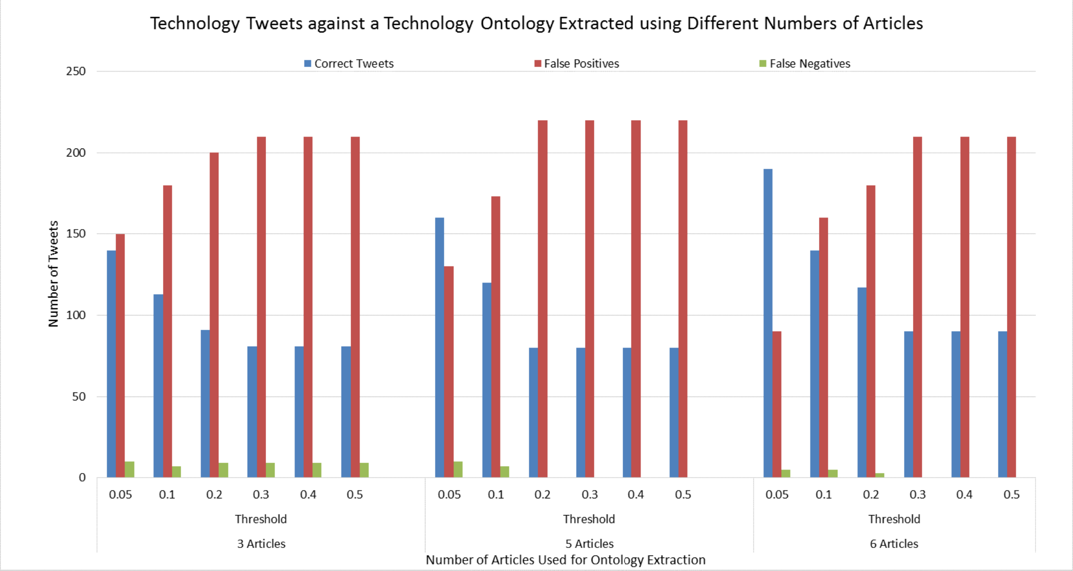


FIGURE 14. Results of experimenting with technology tweets against a technology ontology extracted using different numbers of articles.

inferences regarding the achieved enhancement remains an issue to be further investigated.

E. FINDINGS

In this section, we present the following insights that are generated from the aforementioned exhaustive experiments:

- **Token Similarity Threshold Selection and Adaptation:** Lower thresholds help in achieving better false positive rates, as compared to larger thresholds. However, this takes place at the price of accuracy in detecting actual spams. Accordingly, the most acceptable results occur at an intermediate level, while tolerating a compromise at the level of accuracy.
- **Few are as Good as Many:** After a certain threshold (mainly 30% similarity rate), few terms become as good as many terms while deciding on legitimacy of tweets. We noticed during the experiments that when the results cross a threshold of 0.3, majority of classified tweets (between spam and non-spam) converge. Of course, this becomes of big importance for scalability when the data sets being tested get larger. By adopting the smaller threshold (0.3 instead of 0.5 for instance), we reduce a big part of the overhead and collect results at a faster pace. A lower cardinality, indicating the need for a lower overlap is in this case satisfactory for detecting a real and legitimate tweeting style or content.
- **Comparison between Different Themes:** Different themes produce varied results in terms of accuracy of spam detection. For instance, sports related topics contain a lot of slang, abbreviations and misleading terms. Tweets in this scope are also shorter than tweets in alternative topics. That is why, tracing spam content in these tweets is quite challenging, even with the ontology based approach. On the other hand, politics tweets have a better structuring. Some of them are even complete sentences. Also, the formal sense in those tweets helps in writing longer tweets to complete the sentence. This makes it more relevant to the ontology based approach while examining the tweets content. Therefore, topics can play a role in helping throughout the evaluation phase and this has been already examined in [24].
- **Effect of Using a Larger Ontology:** Ontologies in our case are acting as a white list or dictionary of acceptable terms. Nonetheless, this dictionary includes terms that are frequently mentioned in a group of articles or discussions relative to a topic. That is why, larger numbers of textual documents used in extracting the ontology yielded to enhanced marginal utility. Exact marginal changes as well as the stability rate attracts attention and requires more investigation. The addition of articles is eventually expanding our set of terms and making our dictionary richer which achieves more accuracy in testing.

VII. CONCLUSION

In this paper, we first elaborated on the challenges met while trying to detect spam over Twitter, where restrictions on Twitter's API and constraints to many data attributes have been placed. Moreover, we implemented a set of message to message techniques for detecting spams which showed inefficiency in terms of classification accuracy. In this regard, we proposed an alternative ontology based tweet spam detection approach that identifies spams through content analysis solely. Our proposition overcomes the need for relying on private and user-relationship data, which majority of current spam detection techniques require. Experimental results illustrate that our approach outperforms existing message to message spam detection techniques by around 200% in terms of detection rate due to reduction in false positives and false negatives. Finally, the proposed techniques emphasize the few are as good as many notion by which we witnessed a reduction in the needed overlap to test for token similarity due to result convergence at high thresholds, hence demonstrating the scalability of our approach for large data analysis.

REFERENCES

[1] (May 2013). *Twitter-Stream*. [Online]. Available: archive.org/download/archiveteam-twitter-stream-2013-05

[2] J. Abernethy, O. Chapelle, and C. Castillo, "Web spam identification through content and hyperlinks," in *Proc. 4th Int. Workshop Adversarial Inf. Retr. Web (AIRWeb)*, New York, NY, USA, 2008, pp. 41–44.

[3] B. Alghamdi, J. Watson, and Y. Xu, "Toward detecting malicious links in online social networks through user behavior," in *Proc. Int. Conf. Web Intell. Workshops (WIW)*, 2016.

[4] I. Alsmadi and I. Alhami, "Clustering and classification of email contents," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 27, no. 1, pp. 46–57, 2015.

[5] A. A. Amleshwaram, N. Reddy, S. Yadav, G. Gu, and C. Yang, "CATS: Characterizing automation of Twitter spammers," in *Proc. Int. Conf. Commun. Syst. Netw. (COMSNETS)*, 2013, pp. 1–10.

[6] M. Balakumar and V. Vaidehi, "Ontology based classification and categorization of email," in *Proc. Int. Conf. Signal Process., Commun. Netw.*, 2008, pp. 199–202.

[7] F. Benevenuto, T. Rodrigues, J. Almeida, M. Goncalves, and V. Almeida, "Detecting spammers and content promoters in online video social networks," in *Proc. IEEE INFOCOM Workshops*, Apr. 2009, pp. 1–2.

[8] F. Bentley, N. Daskalova, and N. Andalibi, "If a person is emailing you, it just doesn't make sense: Exploring changing consumer behaviors in email," in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, 2017, pp. 85–95.

[9] M. Bradonjić, A. Hagberg, and A. G. Percus, "Giant component and connectivity in geographical threshold graphs," in *Proc. Int. Conf. Algorithms Models Web-Graph*. San Diego, CA, USA: Springer, 2007, pp. 209–216.

[10] L. Cao, G. Nie, and P. Liu, "Ontology-based spam detection filtering system," in *Proc. Int. Conf. Bus. Manage. Electron. Inf. (BMEI)*, 2011, pp. 282–284.

[11] C. Chen, Y. Wang, J. Zhang, Y. Xiang, W. Zhou, and G. Min, "Statistical features-based real-time detection of drifted Twitter spam," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 4, pp. 914–925, Apr. 2017.

[12] C. Chen, J. Zhang, Y. Xiang, W. Zhou, and J. Oliver, "Spammers are becoming smarter on Twitter," *IT Prof.*, vol. 18, no. 2, pp. 66–70, 2016.

[13] C. Chew and G. Eysenbach, "Pandemics in the age of Twitter: Content analysis of tweets during the 2009 H1N1 outbreak," *PLoS ONE*, vol. 5, no. 11, p. e14118, 2018.

[14] P. Cimiano and J. Völker, "Text2Onto: A framework for ontology learning and data-driven change discovery," in *Proc. 10th Int. Conf. Natural Lang. Process. Inf. Syst. (NLDB)*. Berlin, Germany: Springer-Verlag, 2005, pp. 227–238.

[15] M. Fazil and M. Abulaish, "Why a socialbot is effective in Twitter? A statistical insight," in *Proc. 9th Int. Conf. Commun. Syst. Netw. (COMSNETS)*, 2017, pp. 564–569.

[16] W. Feng *et al.*, "STREAMCUBE: Hierarchical spatio-temporal hashtag clustering for event exploration over the Twitter stream," in *Proc. 31st Int. Conf. Data Eng.*, Apr. 2015, pp. 1561–1572.

[17] A. Gelbukh, "Natutal language processing: Perspective of CIC-IPN," in *Proc. Int. Conf. Adv. Comput., Commun. Informat. (ICACCI)*, 2013, pp. 2112–2121.

[18] P. A. Grabowicz, M. Babaei, J. Kulshrestha, and I. W. Weber. (2016). "The road to popularity: The dilution of growing audience on Twitter." [Online]. Available: https://arxiv.org/abs/1603.04423

[19] A. Guille and C. Favre, "Mention-anomaly-based event detection and tracking in Twitter," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining (ASONAM)*, Aug. 2014, pp. 375–382.

[20] A. Gün and P. Karagöz, "A hybrid approach for credibility detection in Twitter," in *Proc. Int. Conf. Hybrid Artif. Intell. Syst.* Salamanca, Spain: Springer, 2014.

[21] N. Kaur and C. M. Gelowitz, "A tweet grouping methodology utilizing inter and intra cosine similarity," in *Proc. IEEE 28th Can. Conf. Electr. Comput. Eng. (CCECE)*, Halifax, NS, Canada, May 2015, pp. 756–759.

[22] K. Lee, J. Caverlee, and S. Webb, "The social honeypot project," in *Proc. 19th Int. Conf. World Wide Web (WWW)*, 2010, pp. 1139–1140.

[23] S. Lee and J. W. Kim, "WarningBird: A near real-time detection system for suspicious URLs in Twitter stream," *IEEE Trans. Dependable Secure Comput.*, vol. 10, no. 3, pp. 183–195, May/Jun. 2013.

[24] D. Leprovost, L. Abrouk, and D. Gross-Amblard, "Discovering implicit communities in Web forums through ontologies," *Web Intell. Agent Syst., Int. J.*, vol. 10, no. 1, pp. 93–103, 2018.

[25] K. Makice, *Twitter API: Up and Running: Learn How to Build Applications with the Twitter API.* Newton, MA, USA: O'Reilly Media, Inc., 2009.

[26] J. Martinez-Romo and L. Araujo, "Detecting malicious tweets in trending topics using a statistical analysis of language," *Expert Syst. Appl.*, vol. 40, no. 8, pp. 2992–3000, 2013.

[27] G. McDonald, C. Macdonald, I. Ounis, and T. Gollins, "Towards a classifier for digital sensitivity review," in *Proc. Eur. Conf. Inf. Retr.* Amerterdam, The Netherlands: Springer, 2014, pp. 500–506.

[28] M. Newberg, "As many as 48 million Twitter accounts aren't people, says study," CNBC, Fort Lee, NJ, USA, Tech. Rep., Mar. 2017.

[29] B. Patel and D. Shah, "Significance of stop word elimination in meta search engine," in *Proc. Int. Conf. Intell. Syst. Signal Process. (ISSP)*, Mar. 2013, pp. 52–55.

[30] J. Perkins, *Python Text Processing With NLTK 2.0 Cookbook.* Birmingham, U.K.: Packt, 2010.

[31] S. J. Purewal, "How to keep your Twitter following authentic," CNET, Tech. Rep., Mar. 2015.

[32] H. Sarvari, E. Abozinadah, A. Mbaziira, and D. Mccoy, "Constructing and analyzing criminal networks," in *Proc. Secur. Privacy Workshops (SPW)*, 2014, pp. 84–91.

[33] A. M. Shahi, B. Issac, and J. R. Modapothala, "Enhanced intelligent text categorization using concise keywords analysis," in *Proc. Int. Conf. Innov. Manage. Technol. Res. (ICIMTR)*, Malacca, Malaysia, May 2012, pp. 574–579.

[34] J. Shapiro, *Kids Don't Read Books Because Parents Don't Read Books.* New York, NY, USA: Forbes, May 2014.

[35] H. Shen and X. Liu, "Detecting spammers on Twitter based on content and social interaction," in *Proc. Int. Conf. Netw. Inf. Syst. Comput.*, Jan. 2015, pp. 413–417.

[36] M. Shirakawa, K. Nakayama, T. Hara, and S. Nishio, "Wikipedia-based semantic similarity measurements for noisy short texts using extended naive Bayes," *IEEE Trans. Emerg. Topics Comput.*, vol. 3, no. 2, pp. 205–219, Jun. 2015.

[37] M. Shoaib and M. Farooq, "USpam—A user centric ontology driven spam detection system," in *Proc. 48th Int. Conf. Syst. Sci. Hawaii (HICSS)*, 2015, pp. 3661–3669.

[38] C. Sumner, A. Byers, R. Boochever, and G. J. Park, "Predicting dark triad personality traits from Twitter usage and a linguistic analysis of tweets," in *Proc. 11th Int. Conf. Mach. Learn. Appl. (ICMLA)*, 2012, pp. 386–393.

[39] M. Tsikerdekis and S. Zeadally, "Multiple account identity deception detection in social media using nonverbal behavior," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 8, pp. 1311–1321, Aug. 2014.

[40] A. H. Wang, "Don't follow me: Spam detection in Twitter," in *Proc. Int. Conf. Secur. Cryptogr. (SECRYPT)*, Jul. 2010, pp. 1–10.

[41] D. Wang, A. Al-Rubaie, S. Stinčić, Clarke, and J. Davies, "Real-time traffic event detection from social media," *ACM Trans. Internet Technol.*, vol. 18, no. 1, 2018, Art. no. 9.

[42] R. D. Waters and J. Y. Jamal, "Tweet, tweet, tweet: A content analysis of nonprofit organizations' Twitter updates," *Public Relations Rev.*, vol. 37, no. 3, pp. 321–324, 2011.

[43] S. Youn, "SPONGY (spam ontology): Email classification using two-level dynamic ontology," *Sci. World J.*, vol. 2014, Aug. 2014, Art. no. 414583.

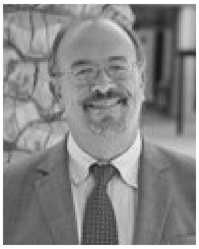
[44] S. Youn and D. McLeod, "Spam email classification using an adaptive ontology," *J. Social Work*, vol. 2, no. 3, pp. 43–55, 2007.

[45] J. C. Zemla, Y. N. Kenett, K.-S. Jun, and J. L. Austerweil, "U-INVITE: Estimating individual semantic networks from fluency data," in *Proc. 38th Annu. Meeting Cognit. Sci. Soc.*, 2016, pp. 1907–1912.

BAHIA HALAWI received the B.S. and M.Sc. degrees (Hons.) in computer science from Lebanese American University in 2012 and 2016, respectively. Her research interests include social network analysis, fog computing, and machine learning.

AZZAM MOURAD (SM'15) received the Ph.D. degree in electrical and computer engineering from Concordia University, Montreal, Canada. He is currently an Associate Professor of computer science with Lebanese American University and an Affiliate Associate Professor with the Software Engineering and IT Department, École de Technologie Supérieure, Montreal. He has served/serves as an Associate Editor for the IEEE COMMUNICATIONS LETTERS, a General Co-Chair for WiMob2016, and a track chair, a TPC member, and a reviewer for several prestigious conferences and journals.

HADI OTROK (SM'14) received the Ph.D. degree in electrical and computer engineering (ECE) from Concordia University, Montreal, Canada. He is currently an Associate Professor with the Department of ECE, Khalifa University, and an Affiliate Associate Professor with the Concordia Institute for Information Systems Engineering, Concordia University, and with the Electrical Department, École de Technologie Supérieure, Montreal. He co-chaired several committees at various IEEE conferences. Moreover, he is a TPC member of several conferences and a reviewer of several highly ranked journals. He is an Associate Editor of *Ad Hoc Networks* (Elsevier) and the *IEEE Communications Letters*.



ERNESTO DAMIANI is currently a Full Professor at the Università degli Studi di Milano, where he leads the SESAR Research Lab, and the Leader of the Big Data Initiative at the EBTIC/Khalifa University, Abu Dhabi, United Arab Emirates. He is a Principal Investigator of the H2020 TOREADOR Project. He was a recipient of the Chester-Sall Award from the IEEE IES Society (2007). He was named ACM Distinguished Scientist (2008) and received the IFIP TC2 Outstanding Contributions Award (2012).