ISSN: 0976-3031

Available Online at http://www.recentscientific.com

CODEN: IJRSFP (USA)

International Journal of Recent Scientific Research
Vol. 8, Issue, 4, pp. 16698-16714, April, 2017

International Journal of Recent Scientific Research

DOI: 10.24327/IJRSR

Research Article

A SEMANTIC METADATA ENRICHMENT SOFTWARE ECOSYSTEMBASED ON MACHINE LEARNING TO ANALYZE TOPIC, SENTIMENT AND EMOTIONS

Ronald Brisebois[1]., Alain Abran[1]., Apollinaire Nadembega*[2] and Philippe N'techobo[3]

[1]École de technologie supérieure, University of Quebec, Montreal, Canada
[2]Network Research Lab., University of Montreal, Montreal, Canada
[3]École Polytechnique de Montréal, Montreal, Canada

DOI: http://dx.doi.org/10.24327/ijrsr.2017.0804.0200

ARTICLE INFO

Article History:

Received 06th January, 2017
Received in revised form 14th February, 2017
Accepted 23rd March, 2017
Published online 28th April, 2017

Key Words:

Emotion detection, natural language processing, semantic topic detection, semantic metadata enrichment, sentiment analysis, text and data mining.

ABSTRACT

In a previous paper, a semantic metadata enrichment software ecosystem(SMESE) based on a multi-platform metadata model and a hybrid machine learning model have been proposed. This work presents the SMESE V3 version enhanced with interest-based enrichments through text analysis approaches for sentiments/emotions detection and hidden topics discovery. SMESE V3 makes it possible to create and use a semantic master catalogue with enriched metadata that allows interest-based search and discovery.

This paper presents the design, implementation and evaluation of a the SMESE V3platform using metadata and data from the web, linked open data, harvesting and concordance rules, and bibliographic record authorities. The SMESE V3 platform includes three distinct engines that:

1. Identify and enrich sentiment/emotion metadata hidden within the text or multimedia structure using the proposed a new BM-Semantic Sentiment and Emotion Analysis algorithm.
2. Propose an hybrid machine learning model for metadata enrichment.
3. Generate semantic to pics by text, and multimedia content analysis using the proposed BM-Scalable Annotation-based Topic Detection algorithm.

The performance of SMESE V3is evaluated using a number of prototype simulations by comparing them to existing enriched metadata technique and classifications. The results show that the enhanced SMESE V3 and related algorithms allow greater performance for purposes of interest-based search.

INTRODUCTION

The rapid development of search and discovery engines, the sudden availability of millions of documents, and the millions upon millions of relationships to linked documents from a growing multitude of sources (e.g., online media, social media and published documents) all make it challenging for a user to find documents relevant to his or her interests or emotions.

The human brain has an inherent ability to detect topics, emotions, relationships or sentiments in written or spoken language. However, the internet, social media and repositories have expanded the number of sources, volume of information and number of relationships so fast that it has become difficult to process all this information[1]. The goal is to increase the find ability of entities matching user interest using external (outside documents) and internal (within documents) semantic metadata enrichment algorithms. While computer search engines struggle to understand the meaning of natural language, semantically enriching entities with meaningful metadata may improve those capabilities. Words themselves are often used inconsistently, having a wide variety of definitions and interpretations. Although there may beno relationship between individual words of a topic or sentiment/emotion, thesauri do express associative relationships between words, ontologies, entities and a multitude of relationships represented as triplets. Finding bibliographic references or semantic relationships in texts makes it possible to localize specific text segments using ontologies to enrich a set of semantic metadata related to topics or sentiments/emotions. The current methodology proposed by researchers for text analysis within the context of entitymeta data enrichment (EME) reduces each document in the corpus to

Corresponding author:* ***Apollinaire Nadembega*
École de technologie supérieure, University of Quebec, Montreal, Canada

a vector of real numbers where each vector represents ratios of counts. Several EME approaches have been proposed, most of them making use of term frequency-inverse document frequency (tf-idf)[2, 3]. In the tf-idf scheme, a basic vocabulary of "words" or "terms" is chosen, then for each document in the corpus, a frequency count is calculated from the number of occurrences of each word [2, 3]. After suitable normalization, the frequency count is compared to an inverse document frequency count (e.g the inverse of the number of documents in the entire corpus where a given word occurs-generally on a log scale, and again suitably normalized). The end result is a term-by-document matrix X whose columns contain the tf-idf values for each of the documents in the corpus. Thus the tf-idf scheme reduces documents of arbitrary length to fixed-length lists of numbers. For non-textual content, tools are available to extract the text from multimedia entities. Bougiatiotis and Giannakopoulos[4]propose an approach that extracts topical representations of movies based on the mining of subtitles.

In the context of this work, we focus on two research axis of the EME research field: Semantic topic detection (STD) and sentiment/emotion analysis (SEA).

On the one hand, STD helps users to efficiently detect meaningful topics. It has attracted significant research in several communities in the last decade, including public opinion monitoring, decision support, emergency management and social media modeling [5, 6]. STD is based on large and noisy data collections such as social media, and addresses both scalability and accuracy challenges. Initial methods for STD relied on clustering documents based on a core group of keywords representing a specific topic, where, based on a ratio such as tf-idf, documents that contain these keywords are similar to each other [2, 3]. Next, variations of tf-idf were used to compute keyword-based feature values, and cosine similarity was used as a similarity (or distance) measure to cluster documents. The subsequent generation of STD approaches, including those based on latent Dirichl *et al* location (LDA), shifted analysis from directly clustering documents to clustering keywords. Some examples of these advances in STD are presented in [7].

However, social media collections differ along several criteria, including the size distribution of documents and the distribution of words. One challenge is to rapidly filter noisy and irrelevant documents, while at the same time accurately clustering a large collection. Bijalwan *et al*. [8], for example, experimented with machine learning approaches for text and document mining and concluded that k-nearest neighbors(KNN), for their data sets, showed the maximum accuracy as compared to naive Bayes and term-graph. The drawback for KNN is that time complexity (i.e., amount of time taken to run) is high but it demonstrates better accuracy than others.

On the other hand, the main objective of sentiment analysis (SA) is to establish the attitude of a given person with regard to sentences, paragraphs, chapters or documents [1, 3, 9-15]. Indeed, many websites offer reviews of items like books, cars, mobiles, movies etc., where products are described in some detail and evaluated as good/bad, preferred/not preferred; unfortunately, these evaluations are insufficient for users in order to help them to make decision. In addition, with the rapid spread of social media, it has become necessary to categorize these reviews in an automated way[3].

For this automatic classification, there are different methods to perform SA, such as keyword spotting, lexical affinity and statistical methods. However, the most commonly applied techniques to address the SA problem belong either to the category of text classification supervised machine learning (SML), which uses methods like naïve Bayes, maximum entropy or support vector machine (SVM), or to the category of text classification unsupervised machine learning (UML). Also, fuzzy sets appear to be well-equipped to model sentiment-related problems given their mathematical properties and ability to deal with vagueness and uncertainty-characteristics that are present in natural languages processing. Thus, a combination of techniques may be successful in addressing SA challenges by exploiting the best of each technique. In addition, the semantic web may be a good solution for searching relevant information from a huge repository of unstructured web data [9].

One current limitation in the area of SA research is its focus on sentiment classification while ignoring the detection of emotions. For example, document emotion analysis may help to determine an emotional barometer and give the reader a clear indication of excitement, fear, anxiety, irritability, depression, anger and other such emotions. For this reason, we focus on sentiment/emotion analysis (SEA) instead of SA.

A number of algorithms or approaches are used to perform text mining, including: latent Dirichl *et al* location (LDA) [7], tf-idf[2, 3], latent semantic analysis (LSA)[16],formal concept analysis (FCA)[17], latent tree model (LTM)[18], naïve Bayes (NB)[19], support vector machine method (SVM) [19], artificial neural network (ANN)[20] based on the associated document's features.

Our approach improves the accuracy of topic detection and sentiment/emotion discovery by semantically enriching the metadata from the linked open data and the bibliographic records existing in different formats .This paper presents the design, implementation and evaluation of anenhancedeco system, called semantic metadata enrichment ecosystem or SMESE V3. Notice that SMESE V3 is an extension of our previous work on SMESE [21].

More specifically, SMESE V3consists of engines implementing two rule-based algorithms to enrich metadata semantically:

1. BM-SATD: generation of semantic topics by text analysis, relationships and multimedia contents.
2. BM-SSEA: discovery of sentiments/emotions hidden within the text or linked to a multimedia structure through an AI computational approach.

Using simulation, the performance of SMESE V3was evaluated in terms of accuracy of topic detection and sentiment/emotion discovery. Existing approaches to enriching metadata (e.g., topic detection or sentiment/emotion discovery) were used for comparison. Simulation results showed that SMESE V3 outperforms existing approaches.

The remainder of the paper is organized as follows. Section 2 presents the related work. Section 3 describes SMESE V3 and its various algorithms while Section 4 presents the prototype of the SMESE V3 multiplatform architecture developed. Section 5presents the evaluation through a number of simulations. Section 6 presents a summary and some suggestions for future work.

RELATED WORK

Interest in entity metadata extraction was initially limited to those in the community who preferred to concentrate on manual design of on tologies as a measure of quality. Following the linked data bootstrapping provided by DB pedia, many changes ensued with a related need for substantial population of knowledge bases, schema induction from data, natural language access to structured data, and in general all applications that make for joint exploitation of structured and unstructured content. In practice, Graph-based methods, meanwhile, were incrementally entering the toolbox of semantic technologies at large.

Topic detection

In the last decade, semantic topic detection has attracted significant research in several communities, including information retrieval. Generally, a topic is represented as a set of descriptive and collocated keywords/terms. Initially, document clustering techniques were adopted to cluster content-similar documents and extract keywords from clustered document sets as the representation of topics (subjects). The predominant method for topic detection is the latent Dirichl *et al* location (LDA) [7], which assumes a generating process for the documents. LDA has been proven a powerful algorithm because of its ability to mine semantic information from text data. Terms having semantic relations with each other are collected as a topic. LDA is a three-level hierarchical Bayesian model, in which each item of a collection is modeled as a finite mixture over an underlying set of topics. Each topic is, in turn, modeled as an infinite mixture over an underlying set of topic probabilities. In the context of text modeling, topic probabilities provide an explicit representation of a document.

The literature presents two groups of text-based topic detection approaches based on the size of the text: short text [17, 22-24]such as tweets or Facebook posts, and long text [4, 5, 7, 18, 25, 26] such as a book. For example, Dang *et al*.[22] proposed an early detection method for emerging topics based on dynamic Bayesian networks in micro-blogging networks. They analyzed the topic diffusion process and identified two main characteristics of emerging topics, namely attractiveness and key-node. Next, based on this identification, they selected features from the topology properties of topic diffusion, and built a DBN-based model using the conditional dependencies between features to identify the emerging keywords. But to do so, they had to create a term list of emerging keyword candidates by term frequency in a given time interval.

Cigarran *et al*. [17] proposed an approach based on formal concept analysis (FCA). Formal concepts are conceptual representations based on the relationships between tweet terms and the tweets that have given rise to them.

Cotelo *et al*. [23], when addressing the tweet categorization task, explored the idea of integrating two fundamental aspects of a tweet: the textual content itself, and its underlying structural information. This work focuses on long text topic detection.

Recently, considerable research has gone into developing topic detection approaches using a number of information extraction techniques (IET), such as lexicon, sliding window, boundary techniques, etc. Many of these techniques [5, 18, 25, 26]rely heavily on simple keyword extraction from text.

For example, Sayyadi and Raschid [5] proposed an approach for topic detection, based on keyword-based methods, called KeyGraph, that was inspired by the keyword co-occurrence graph and efficient graph analysis methods.

In other words, KeyGraph is based on the similarity of keyword extraction from text. We note two limitations to the approach, which requires improvement in two respects. Firstly, they failed to leverage the semantic information derived from topic model. Secondly, they measured co-occurrence relations from an isolated term-term perspective; that is, the measurement was limited to the term itself and the information context was overlooked, which can make it impossible to measure latent co-occurrence relations.

Salatino and Motta [26]suggested that it is possible to forecast the emergence of novel research topics even at an early stage and demonstrated that such an emergence can be anticipated by analyzing the dynamics of pre-existing topics.

Sentiment analysis (SA)

There are three main techniques for sentiment analysis (SA)[27]:keyword spotting. lexical affinity and statistical methods. The first two methods are well known while statistical methods have to be more explored further.

Statistical methods, such as Bayesian inference and support vector machines, are supervised approaches in which a labeled corpus is used for training a classification method which builds a classification model used for predicting the polarity of novel texts. By feeding a large training corpus of affectively annotated texts to a machine learning algorithm, it is possible for the system to not only learn the affective valence of related keywords (as in the keyword spotting approach), but also to take into account the valence of other arbitrary keywords (like lexical affinity), punctuation, and word co-occurrence frequencies. Sentiment analysis can be carried out at different levels of text granularity: document [19, 28-32], sentence[1, 3, 9, 33, 34], phrase [35], clause, and word [20, 36, 37]. Sentiment analysis may be at the sentence or phrase level (which has recently received quite a bit of research attention) or at the document level.

In [11], the authors presented a survey of over one hundred studies published in the last decade on the tasks, approaches, and applications of sentiment analysis. With a major part of available worldwide data being unstructured (such as text, speech, audio, and video), this poses important research challenges. In recent year numerous research efforts have led to automated SEA, an extension of the NLP area of research.

The first five dimensions represent tasks to be performed in the broad area of SEA. For the first three dimensions (subjectivity classification, sentiment classification and review usefulness

measurement), the authors note that the applied approaches are broadly classified into three categories: machine learning, lexicon based and hybrid approaches.

Since one of our research objectives was to extract sentiment and emotion metadata from documents, the rest of this section focuses on sentiment classification, lexicon creation, and opinion word and product aspect extraction. Sentiment classification is concerned with determining the polarity of a sentence; that is, whether a sentence is expressing positive, negative or neutral sentiment towards the subject. A lexicon is a vocabulary of sentiment words with respective sentiment polarity and strength value while opinion word and product aspect extraction is used to identify opinion on various parts of a product. For the purpose of this paper, we assume that a document expresses sentiments on a single content and is written by a single author.

Cho *et al*. [30]proposed a method to improve the positive vs. negative classification performance of product reviews by merging, removing, and switching the entry words of the multiple sentiment dictionaries. They merge and revise the entry words of the multiple sentiment lexicons using labeled product reviews. Specifically, they selectively remove the sentiment words from the existing lexicon to prevent erroneous matching of the sentiment words during lexicon-based sentiment classification. Next, they selectively switch the polarity of the sentiment words to adjust the sentiment values to a specific domain. The remove and switch operations are performed using the target domain's labeled data (i.e. online product reviews) by comparing the positive and negative distribution of the labeled reviews with a positive and negative distribution of the sentiment words. They achieved 81.8% accuracy for book reviews. However, their contribution is limited to development of a novel method of removing and switching the content of the existing sentiment lexicons.

Moraes *et al*. [19] compared well-known machine learning approaches (SVM and NB) with an ANN-based method for document-level sentiment classification. Naive Bayes (NB) is a probabilistic learning method that assumes terms occur independently while the support vector machine method (SVM) seeks to maximize the distance to the closest training point from either class in order to achieve better generalization/classification performance on test data. The authors reported that, despite the low computational cost of the NB technique, it was not competitive in terms of classification accuracy when compared to SVM. According to the authors, many researchers have reported that SVM is perhaps the most accurate method for text classification. Artificial neural network (ANN) derives features from linear combinations of the input data and then models the output as a nonlinear function of these features. Experimental results showed that, for book datasets, SVM outperformed ANN when the number of terms exceeded 3,000. Although SVM required less training time, it needed more running time than ANN. For 3,000 terms, ANN required 15 sec training time (with negligible running time) while SVM training time was negligible (1.75 sec). In addition, their contribution was limited to performing comparisons between existing approaches. As in [19], Poria S. *et al*.[38] experimented with existing approaches and showed that SVM is a better approach for text-based emotion detection.

Emotion analysis

Emotions are also associated with mood, temperament, personality, outlook and motivation [27, 39, 40]. However, sentiments are differentiated from emotions by the duration in which they are experienced. The SWAT model was proposed to explore the connection between the evoked emotions of readers and news headlines by generating a word-emotion mapping dictionary. For each word *w* in the corpus, it assigns a weight for each emotion *e*, i.e., P(e|w) is the averaged emotion score observed in each news headline *H* in which *w* appears. The emotion-term model is a variant of the NB classifier and was designed to model word-emotion associations. In this model, the probability of word *wj* conditioned on emotion *ek* is estimated based on the co-occurrence count between word *wj* and emotion *ek* for all documents. The emotion-topic model is combination of the emotion-term model and LDA.

A system for text-based emotion detection is proposed by Anusha and Sandhya [41]which uses a combination of machine learning and natural language processing techniques. They used the Stanford CoreNLP toolkit to create the dependency tree based on word relationships. Phrase selection is done using the rules on dependency relationships that gives priority to the semantic information for the classification of a sentence's emotion. Next, they used the Porter stemming algorithm for stemming, and stop words removal and tf-idf to build the feature vectors.

Cambria *et al*.[42]explored how the high generalization performance, low computational complexity, and fast learning speed of extreme learning machines can be exploited to perform analogical reasoning in a vector space model of affective common-sense knowledge. After performing TSVD on Affect Net, they used the Frobenius norm to derive a new matrix. For the emotion categorization model, they used the Duchenne smile and the Klaus Scherer model.

Conclusion

Some of our key findings from the related work on sentiment and emotion analysis are:

1. Traditional sentiment analysis methods mainly use terms and their frequency, part of speech, rule of opinions and sentiment shifters. Semantic information is ignored in term selection, and it is difficult to find complete rules.
2. Most of the recent contributions are limited to sentiment analysis elaborated in terms of positive or negative opinion and do not include the analysis of emotions.
3. Existing approaches do not take into account the human contribution to improve accuracy.
4. Existing approaches do not combine sentiment and emotion analysis.
5. Lexicon and ontology based approaches provide good accuracy for text-based sentiment and emotion analysis when applying SVM techniques. In our work, it is more important to identify the sentiment and emotion of a book taking into account all the books of the collection. For example, assuming that book A has 90% fear and 80% sadness while the emotion which has the best weight of book B is 40% fear; can it be said that fear is the emotion of book B as in book A?

6. Existing approaches do not take into account document collections. In terms of granularity, most of the existing approaches are sentence-based.
7. These approaches do not take into account the context around the sentence and in this way, it is possible to miss the real emotion.

As a general conclusion to the literature review on topic detection, sentiment and emotion analysis, 95% of the work focused on features of the documents (e.g., sentence length, capitalized words, document title, term frequency, and sentences position) to perform text mining and generally make use of existing algorithms or approaches (e.g., LDA, tf-idf, VSM, SVD, LSA, TextRank, PageRank, LexRank, FCA, LTM, SVM, NB and ANN) based on their features associated to documents.

Table I compares the most known text mining algorithms (e.g., AlchemyAPI, DBpedia, Wikimeta, open calais, Bitext, AIDA, TextRazor) with our algorithms proposed in SMESE V3by keyword extraction, classification, sentiment analysis, emotion analysis and concept extraction.

RULE-BASED SEMANTIC METADATA INTERNAL ENRICHMENT ENGINE

This section presents an overview and the details of the proposed a rule-based semantic metadata internal enrichment engine, a Machine Learning Engine (MLE), including two different algorithms (BM-SATD and BM-SSEA).

MLE is part of the SMESE V3platform architecture as shown in Fig. 1. The main goal of SMESE V3 is to enhance the SMESE platform through text analysis approaches for topics, sentiment/emotion and semantic relationships detection. SMESE V3 allows one to create a semantic master catalogue with enriched metadata that enables the search and discovery interest-based engines. To perform this task, the following tools are needed:

1. Topics are a controlled set of terms designed to describe the subject of a document. While topics do not necessarily include relationships between terms, we include relationships as triplets (Entity - Relationship - Entity).
2. A multilingual thesauri and ontology to provide hierarchical relationships as well as semantic relationships between topics.

Table I Summary of attribute comparison of existing and proposed algorithms

Existing algorithms	Keyword extraction	Classification	Sentiment analysis	Emotion analysis	Concept extraction
AlchemyAPI (http://www.alchemyapi.com/)	x	x	x	x	x
DBpedia Spotlight (https://github.com/dbpedia-spotlight)					x
Wikimeta (https://www.w3.org/2001/sw/wiki/Wikimeta)					x
Yahoo! Content Analysis API (out of date) (https://developer.yahoo.com/contentanalysis/)		x			x
Open Calais (http://www.opencalais.com/)	x	x			x
Tone Analyzer (https://tone-analyzer-demo.mybluemix.net/)			x	x	
Zemanta (http://www.zemanta.com/)					x
Receptiviti (http://www.receptiviti.ai/)			x	x	
Apache Stanbol (https://stanbol.apache.org/)					x
Bitext (https://www.bitext.com/)			x		x
Mood patrol (https://market.mashape.com/soulhackerslabs/moodpatrol-emotion-detection-from-text)				x	
Aylien (http://aylien.com/)	x	x	x		
AIDA (http://senseable.mit.edu/aida/)					x
Wikifier (http://wikifier.org/)					x
TextRazor (https://www.textrazor.com/)					x
Synesketch (http://krcadinac.com/synesketch/)				x	
Toneapi (http://toneapi.com/)			x	x	
SMESE V3	x	x	x	x	x

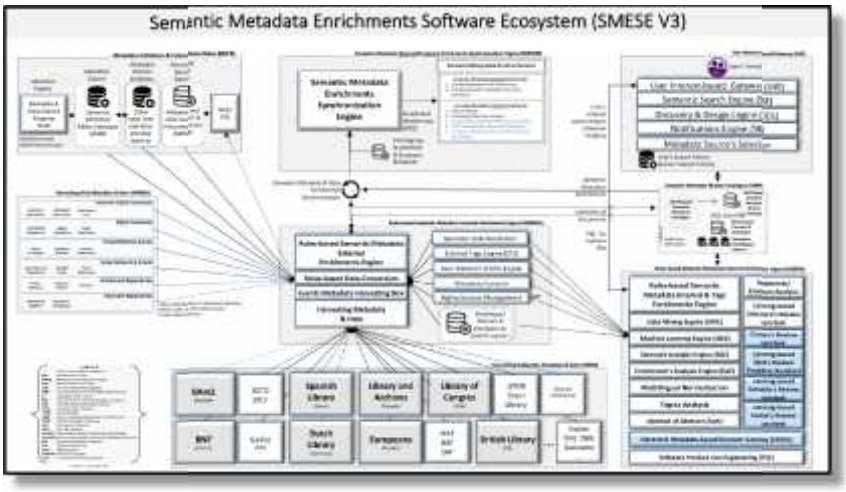

Fig 1 SMESE V3-Semantic Metadata Enrichment Software Ecosystem

3. Anontology to provide a representation of knowledge with rich semantic relationships between topics. By breaking content into pieces of data, and curating semantic relationships to external contents, metadata enrichments are created dynamically.

In Fig. 1, the V3 improvements to the SMESE platform from this work and its implementation are presented in blue.

The following sub-sections present the terminology and assumptions, the necessary pre-processing and details of the two algorithms proposed and implemented.

Terminology and assumptions

In this section the following terms are defined:

1. A word or term is the basic unit of discrete data, defined to be an item from a vocabulary indexed by {1, …,V}. Terms are presented using unit-basis vectors. Thus, the i^{th} term in the vocabulary is represented by an I-vector *w* such that $w^i = 1$ and $w^j = 0$ for $i \neq j$.
2. A lineis a sequence of N terms denoted by *l*.
3. A document is a sequence of N lines denoted by D = (w_1,w_2; …,w_N), where w_i is the i^{th} term in the sequence coming from the lines. D is represented by its lines as D = (l_1, …l_i,…,l_K).
4. A corpus is a collection of *M* documents denoted by C={D_1, D_2, …, D_M}.
5. An emotion word is a word with strong emotional tendency or a probabilistic distribution.

To implement the BM-SATD and BM-SSE Aalgorithms, machine learning models have been used to perform metadata enrichments (see Fig.2):

3. A Machine Learning Engine allows to use a combination of supervised and unsupervised and allows to generate a predictive model
4. A feedback processing allows to the Machine Learning Engine to learn.
5. New texts or documents who are converted into Metadata vectors use the predictive model generated in 3.

Document pre-processing

The objective of the pre-processing is to filter noise and adjust the data format to be suitable for the analysis phases. It consists of stemming, phase extraction, part-of-speech filtering and removal of stop-words. The corpus of documents crawled from specific databases or the internet consists of many documents. The documents are pre-processed into a basket dataset C, called the document collection. C consists of lines representing the sentences of the documents. Each line consists of terms, i.e. words or phrases. More specifically, a pre-processing including tokenization, lower casing and stemming of all the terms using the Porter stemmer[43] is performed.

Scalable annotation-based topic detection: BM-SATD

The aim of BM-SATD is to build a classifier that can learn from already annotated contents (e.g., documents and books) and infer the topics of new books. Traditional approaches are typically based on various topic models, such as latent Dirichl *et al* location (LDA) where authors cluster terms into a topic by mining semantic relations between terms. However, co-occurrence relations across the document are commonly neglected, which leads to detection of incomplete information.

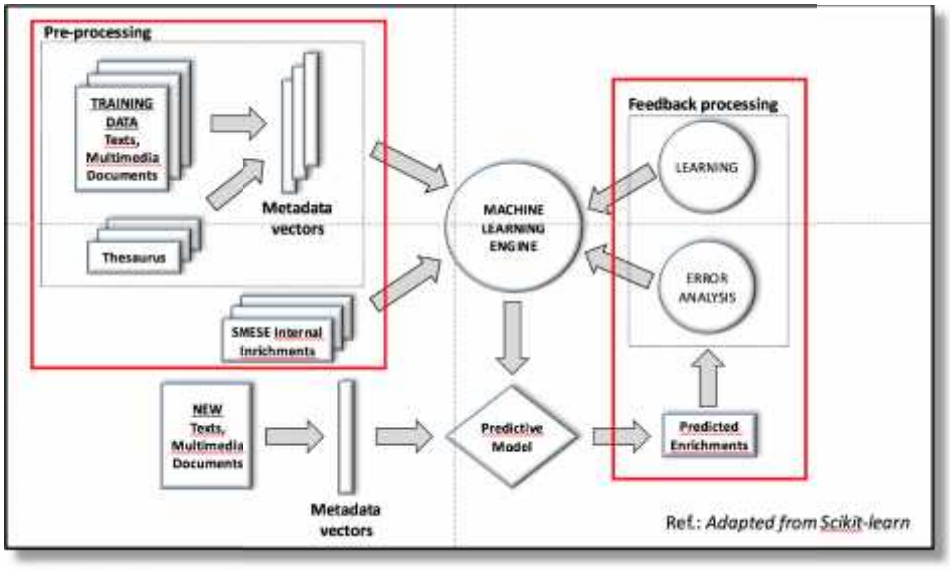

Fig. 2 Supervised Learning applied to Metadata Enrichments

1. There is a pre-processing using Training Data.
2. One or multiple thesaurus are available. A thesaurus contains a list of words with synonyms and related concepts. This approach uses synonyms or glosses of lexical resources in order to determine the emotion or polarity of words, sentences and documents.

Furthermore, the inability to discover latent co-occurrence relations via the context or other bridge terms prevents important but rare topics from being detected. BM-SATD combines semantic relations between terms and co-occurrence relations across the document making use of document annotation. In addition, BM-SATD includes:

1. A probabilistic topic detection approach, called semantic topic model (BM-SemTopic).
2. A clustering approach that is an extension of KeyGraph, called semantic graph (BM-SemGraph).

BM-SATD is a hybrid relation analysis and machine learning approach that integrates semantic relations, semantic annotations and co-occurrence relations for topic detection. More specifically, BM-SATD fuses multiple relations into a term graph and detects topics from the graph using a graph analytical method. It can detect topics not only more effectively by combining mutually complementary relations, but it can also mine important rare topics by leveraging latent co-occurrence relations. The following sub-sections present the details of the five phases of the BM-SATD model.

Relevant and less similar documents selection

A filtering process is performed to avoid using a large corpus of documents that are similar or not relevant. It is not necessary to compare a new document of a collection with two other documents of the collection that are similar in order to know whether this new document is similar to each of the other documents. This strategy merely increases computation time. Here, only documents that are already annotated by topic are considered.

Not annotated documents semantic term graph generation

The semantic term graph is a basis for detecting topics automatically. The BM-SemGraph has one node for each term in the vocabulary of the document. Edges in a BM-Sem Graph represent the co-occurrence of the corresponding keywords and are weighted by the count of the co-occurrences. Note that, in contrast to existing graph-based approaches, the co-occurrence between A and B is different from the co-occurrence between B and A. This difference allows one to retain the semantic sense of co-occurrence terms.

Step 1: Co-occurrence clusters generation

For the co-occurrence graph, the assumption is that terms that have a close relation to each other may be linked by the co-occurrence link. The relation between two terms W_i and W_j is measured by their conditional probability. Let D be a document and $V_D = (w_1, w_2; \ldots, w_N)$ be the terms of D and L_D be the number of lines of D.

The conditional probability $p(\overline{W_i}, \overline{W_j}^{\varepsilon})$ of $\overline{W_i}, \overline{W_j}^{\varepsilon}$ is computed using equation (1)where:

1. ε denotes the minimum distance between W_i and W_j
2. The distance between two terms is the number of terms that appear between them for a given line
3. ε is a parameter determined by experimentation.

$$p\left(\overline{W_i}, \overline{W_j}^{\varepsilon}\right) = \sum_{l=1}^{L_D} \frac{N^{line\ l}\left(\overline{W_i}, \overline{W_j}^{\varepsilon}\right)}{\frac{N(line\ l)}{\varepsilon}} \quad (1)$$

where $N^{line\ l}\ \overline{W_i}, \overline{W_j}^{\varepsilon}$ denotes the number of times that W_i and W_j co-occur with a minimum distance ε and where W_i appears before W_j, and *N(line l)* denotes the number of terms of the line l.

To formally define a relation between two terms W_i and W_j, their frequent co-occurrence measured by the conditional probability $p(\overline{W_i}, \overline{W_j}^{\varepsilon})$, needs to exceed the co-occurrence threshold. The co-occurrence threshold is also determined by experimentation. Note that frequent co-occurrence is oriented. This allows one to retain the semantic orientation of the links between terms. Next, the oriented links are transformed into simple links without losing the semantic context.

Step 2: Cluster optimization

To improve quality, clusters should be pruned, such as by removing weak links or partitioning sparse cluster into cohesive sub-clusters. Clusters are pruned according to their connectedness. The link *e* is pruned when no path connects the two ends of *e* after it is pruned. The link between the black node and the green node should be pruned. Secondly, cliques are identified. Let C be the clique and W_i and W_j be the nodes of C that are linked to another node. The weight between W_i and W_j is given by equation (2):

$$w\ W_i, W_j = \max_{\substack{W_k \in C \\ W_s \in C}} w\ W_k, W_s \quad (2)$$

Step 3: Key term extraction

To extract key terms, the relation between a term and a cluster is measured. It is assumed that the weight of a term in a given cluster may be used to determine the importance of this term for the cluster. Let R be the set of nodes of the cluster C where the node $W_i$is inside. The weight of W_i in the cluster C is given by equation (3):

$$f\ W_i = \sum_{W_j \in R} w\ W_i, W_j \quad (3)$$

To identify a term as a key term, a sort of terms is performed based on their weights regardless of the clusters that they are in. Next, the Num KeyTerm terms that have the largest weights are selected as Key Terms. NumKeyTerm is a parameter.

Step 4: Semantic topic generation

Semantic topic generation combines a correlated topic model (CTM) [44]and a domain knowledge model (DKM) [45], called BM semantic topic model (BM-SemTopic), to build the real semantic topic model. In LDA, a topic is a probability distribution over a vocabulary. It describes the relative frequency each word is used in a topic. Each document is regarded as a mixture of multiple topics and is characterized by a probability distribution over the topics. A limitation of LDA is its inability to model topic correlation. This stems from the use of the Dirichlet distribution to model the variability among topic proportions. In addition, standard LDA does not consider domain knowledge in topic modeling.

To overcome these limitations, BM-SemTopic combines two models:

1. A correlated topic model (CTM)[44] that makes use of a logistic normal distribution.
2. A domain knowledge model (DKM)[45] that makes use of the Dirichlet distribution.

BM-SemTopic uses a weighted sum of CTM and DKM to compute the probability distribution of term W_i on the topic *z*. The sum is defined by equation (4):

$$h(W_i|z) = \omega CTM(W_i|z) + (1-\omega)\, DKM(W_i|z) \quad (4)$$

where ω is used to give more influence to one model based on the term distribution of topics.

When the majority of terms are located in a few topics, this means the domain knowledge is important and ω must be small. BM-SemTopi*c* develops the CTM where the topic proportions exhibit a correlation with the logistic normal distribution and incorporates the DKM. A key advantage of BM-SemTopicis that it explicitly models the dependence and independence structure among topics and words, which is conducive to the discovery of meaningful topics and topic relations.

CTM is based on a logistic normal distribution. The logistic normal is a distribution on the simplex that allows for a general pattern of variability between the components by transforming a multivariate normal random variable. This process is identical to the generative process of LDA except that the topic proportions are drawn from a logistic normal distribution rather than a Dirichlet distribution.

DKM is an approach to incorporation of such domain knowledge into LDA. To express knowledge in anontology, BM-SemTopicuses two primitives on word pairs: Links and Not-Links. BM-Sem Topic replaces the Dirichlet prior by the Dirichlet Forest prior in the LDA model. Then, BM-Sem Topic sorts the terms for every topic in descending order according to the probability distribution of the topic terms. Nextit picks up the high-probability terms as the feature terms. For each topic, the terms with probabilities higher than half of the maximum probability distribution are picked up.

Step 5 Semantic term graph extraction

To discover semantic relations between the semantic terms, the semantic aspect is included making use of Word Net::Similarity [46]. Based on the structure and content of the lexical database Word Net, Word Net::Similarity implements six measures of similarity and three measures of relatedness. Measures of similarity use information found in a hierarchy of concepts that quantify how much concept A is like concept B.

When the semantic terms are identified, the semantic value of each topic's candidates is computed. The semantic value of each term W_i, is given by equation (5):

$$SEM(W_i|z) = TP - ITP(W_k|z) = h(W_i|z) * \log\left(\frac{|Z|}{\sum_{t \in Z} h(W_i|t)}\right) \quad (5)$$

where Z denotes the set of semantic topics. TP-ITP is inspired by the tf-idf formula, where TP is term probability and ITP inverse topic probability.

Semantic links between semantic terms for the term graph are constructed using the vector measure, one of the measures of relatedness of Word Net::Similarity [46].The vector measure creates a co-occurrence matrix for each word used in Word Net glosses from a given corpus, and then represents each gloss/concept with a vector that is the average of these co-occurrence vectors.

Let W_i and W_j be semantic terms of the synsets A and B, respectively. Let $\vec{A} = (a_1, \ldots, a_q)$ and $\vec{B} = (b_1, \ldots, b_q)$ be the co-occurrence vectors of A and B, respectively. Let V_z be the set of semantic terms of the semantic topic Z. The weight of the link between W_i and $W_j$is computed by equation (6):

$$Dis(W_i, W_j|z) = \frac{SEM(W_i|z) + SEM(W_j|z)}{\frac{\sum_{W_k \in V_z} SEM(W_k|z)}{n}} \times \sum_{l=1} (a_l - b_l)^2 \quad (6)$$

To discover a semantic relation between two terms, the semantic distance is computed. The semantic distance between two terms is the shortest path between the terms using equation (7):

$$SEMDis(W_i, W_j|z) = \min_{pa \in P} \sum_{W_k \in pa} Dis(W_i, W_k|z) \quad (7)$$

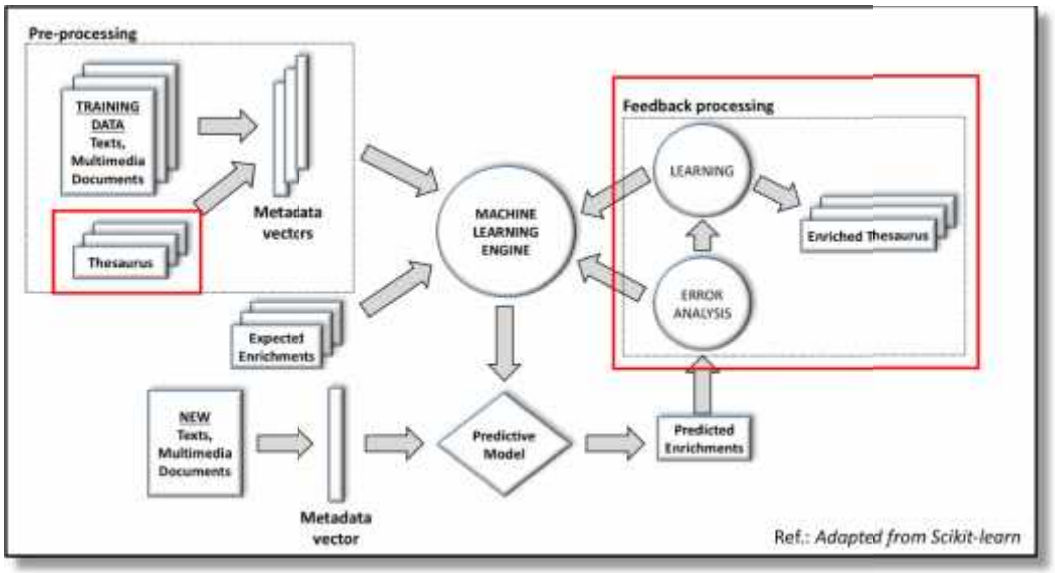

Fig. 3 Supervised Learning applied to Metadata Enrichments

where *pa*, W_k, and *P* denote a path between W_i and W_j in the thesaurus, a term on a path *pa* and the set of path s*pa* between W_i and W_j, respectively. See Fig. 3, in the pre-processing phase, we can notice the usage of thesaurus. At the end of the machine learning process, an enriched thesaurus is generated to be part of the input of the machine learning process.

To formally define a semantic relation between two terms W_i and W_j, the semantic distance $SEMDis(W_i,W_j'|z)$must not exceed the semantic threshold. The semantic threshold is determined by experimentation.

The last process to generate the semantic term graph BM-SemGraph is amerging of the term graph and the semantic graph. The term graph and semantic graph are merged by coupling the co-occurrence relation and the semantic relation. New terms are added as semantic terms and new links are added as semantic links if they do not appear in the term graph. For each link between two nodes W_j and W_k of the merged graph, the weight, called the BM Weight (BMW), for a given topic $t_i$is computed using equation (8):

$$BMW(W_j',W_k|\,t_i) = \frac{\lambda}{SEMDis(W_j',W_k\,|t_i)} + (1-\lambda) \times w(W_i,W_j') \quad (8)$$

where λdetermined by experimentation.

Topic detection

Topics that may be associated with a new document are detected based on the BM-SemGraph. Note that the BM-SemGraph is obtained using a collection of documents. In this case, the likelihood of detecting topics among a collection of documents is high and must be computed. To accomplish this, the feature vector of each topic based on the clusters of BM-SemGraphis computed. The feature vector of a topic is calculated using the BMRank of each topic term. Let A be the set of nodes of BM-SemGraph directly linked to term W_j in the topic t_i.The score forterm $W_j$is given by equation (9):

$$BMRank(W_j'|t_i) = \frac{\sum_{W_k \in A} BMW(W_j',W_k|\,t_i)}{|A|} \quad (9)$$

The term with the largest BMRank is called the main term of the topic;the other terms are secondary terms. The same processes are used to obtain the BM-SemGraph of an individual document *d* and the feature vectors of topics t_j^d. Next, the similarity between each topic t_i and the topics t_j^d of document *d* is computed in order to detect document topics.

Training

The training process establishes a terms graph based on the relevant and less similar documents for a given topic t_i. To form the terms graph for a given topic, the pre-processing of its relevant and less similar documents is first carried out, a set of linesis obtained where each line is a list of terms, and the co-occurrence of these terms is then computed.

Topics refining

The architecture overview of the topic refining process phase in BM-SATD is presented in Fig. 4, this process refines the detected topics making use of relevant documents already annotated by humans based on existing or known topics. Following this process, three lists of topics are obtained: a list of new topics, a list of similar existing topics and a list of not similar existing topics.

The list of existing topics that match new document detected topics is identified based on the new document detected topics and annotated documents by topic (existing topics). Then, the clusters of terms by to pic are identified based on the collection of relevant and less similar documents. Note that each topic is a cluster of terms graph. Therefore, in this case, a graph matching technique is a good candidate to perform topic similarity detection.

Next, using our graph matching technique, the clusters of terms by topics of relevant and less similar collection of annotated documents which match with CTGare identified, for each cluster of terms graph by topic (CTG) of the new document. The matching score between two clusters is then computed. Let:

1. H be the new document terms graph and G be the terms graph obtained by a training process applied on the collection of relevant and less similar documents annotated by topics.
2. C_j^d be a cluster of H associated to topic t_j^d and C_i be a cluster of G associated with topic t_i.
3. W_i and W_j be two terms of cluster C_j^d; the link matching function g($\overrightarrow{W_i'W_j'}$) between W_i and W_j is defined by equation (10):

$$g: C_j^d \times C_j^d \rightarrow IR$$
$$g(W_i'W_j') = \begin{cases} MinHopClusterOfti(W_i,W_j) & if\ path\ between\ W_i,W_j \\ 1+MaxHopClusterOfti & if\ not\ path\ between\ W_i,W_j \end{cases} \quad (10)$$

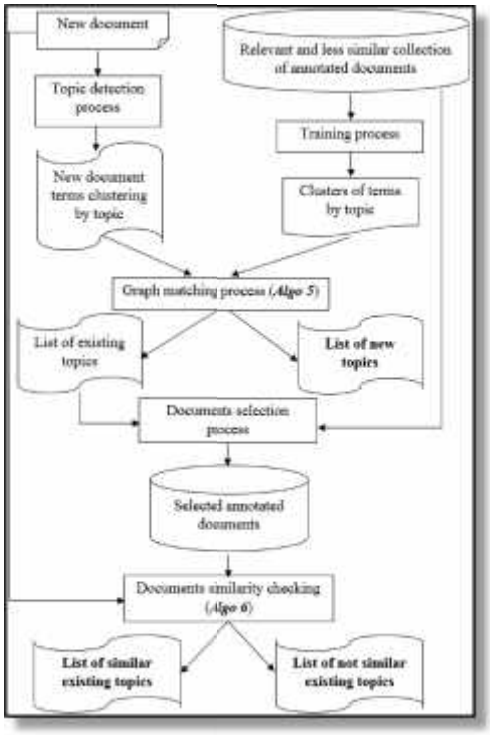

Fig 4 BM-SATD Topic refining process phase-Architecture overview

For a direct link $\overrightarrow{W_iW_j}$ (only one hop between W_i and W_j) of cluster C_j^d, the process checks whether there is a path between W_i and W_j in the cluster C_i, regardless of the number of hops. Using the link matching function, the matching score between two clusters C_i^d and $C_i$is given by equation (11):

$$o \square H \times G \rightarrow 0;1$$
$$o\ C_j^d, C_i\ = \frac{C_j^d}{\sum_{W_i,W_j \square C_j^d} g(\overrightarrow{W_iW_j})} \quad (11)$$

where C_j^d is the number of links inclusters C_i^d.

Semantic sentiment and emotion analysis: BM-SSEA

The BM-SSEA goal is to classify the corpus of documents taking emotion into consideration, and to determine which sentiment it more likely belongs to. A document can be a distribution of emotion $p\ e|d\ \ e \square E$ and a distribution of sentiment $p\ s|d\ \ s \square S$.BM-SSEA is a hybrid approach that combines a keyword-based approach and a rule-based approach.BM-SSEA is applied at the basic word level and requires an emotional keyword dictionary that has keywords (emotion words) with corresponding emotion labels. To refine the detection, BM-SSEA develops various rules to identify emotion. Rules are defined using an affective lexicon that contains a list of lexemes annotated with their affect.

The emotional keyword dictionary and the affective lexicon are implemented in a thesaurus. BM-SSEA is a knowledge-based approach that uses an AI computational technique. The purpose of BM-SSEA is to identify positive and negative opinions and emotions.

For affective text evaluation, BM-SSEA uses the SS-Tagger (a part-of-speech tagger)[47] and the Stanford parser[48]. The Stanford parser was selected because it is more tolerant of constructions that are not grammatically correct. This is useful for short sentences such as titles. BM-SSEA also uses several lexical resources that create the BM-SSEA knowledge base located in the thesaurus.The lexical resources used are: WordNet, WordNet-Affect, SentiWordNet and NRC emotion lexicon. WordNet is a semantic lexicon where words are grouped into sets of synonyms, called synsets. WordNet-Affect is a hierarchy of affective domain labels that can further annotate the synsets representing affective concepts.

The NRC emotion lexicon is a thesaurus that associates for a word, the value one or zero for each emotion. This association is made of binary vectors. The disadvantage of this thesaurus is that since the values are binary, all words belonging to an emotion have the same weight for that emotion. To address this problem, the NRC emotion lexicon thesaurus was combined with the Word Net, WordNet-Affect and SentiWord Netthesaurus. This associates a feelings score with each word-POS. Where POS_1 are grammatical categories used to classify words in dimensions such as adjectives or verbs. Senti Word Net associates with each couple a valence score that can be either negative or positive with respect to the sense of the word in question. The word death, for example, is likely to have a negative score. BM-SSEA also relies on shifter valences.

For example, take the phrase "I am happy" witha score of 1 for the joy emotion. For the phrase "I am **very** happy", 'very' is a valence intensifier that will change the joy emotion score to 2. In the case, "I am **not** happy" the modifier 'not' will change the emotion joy to the contrary emotion sadness.

The main component of BM-SSEA is the thesaurus, called BMemotion word model (BMEmoWordMod). BMEmoWord Mod is an emotion-topic model that provides the emotional score of each keyword by taking the topic into account.

BMEmoWordMod introduces an additional layer (i.e., latent topic) into the emotion-term model such as SentiWordNet.BM-SSEA is composed of three phases: BMEmoWordMod generation process phase, sentiment and emotion discovery process phase and third sentiment and emotion refining process phase. The following sub-sections describe the three phases of the BM-SSEA model used to discover sentiment and emotion.

BMEmoWordMod generation process phase

A training set from the original corpusis created. The most relevant and discriminative documents are selected automatically. In the second step, each word is tagged with a POS and the combination of word and POS used as the essential feature. Finally, BMEmoWordMod is generated using the extracted features, which can then be used to discovery the sentiments and emotions of new documents. Many steps have to be completed: (1) Training set selection, (2) Intermediate lexicon generation and (3) Sentiment and emotion lexicon generation.

Sentiment and emotion discovery

This phase identifies the sentiments and emotions that are likely associated with a given new document by using the sentiment and emotion semantic lexicon BMEmo WordMod generated in the previous section. After preprocessing, the term vector of the new document is defined using TF-IDF.

Let ND be the new document and $W_{ND} = \{W_1, \ldots, W_z\}$ the set of distinct terms occurring in the corpus of documents. To obtain the z-dimensional term vector that represents each document in thecorpus, the tf-idf of each term of W_z is computed. The result of this computation establishes the term vector $\bar{t}_{ND} = (\text{tfidf}\ W_1, ND\ , \ldots , \text{tfidf}\ W_z, ND\ )$.

Using vector $\bar{t}_{ND}$, $T_{ND}=\{t_p , \ldots, t_q\}$ obtained using BM-SATD and BMEmoWordMod, the sentiment and emotion vector of new document

$\bar{E}_{f_j,ND} =$
$(E\ f_j, \text{ND}, e_1\ , \ldots , E\ f_j, \text{ND}, e_E\ , E\ f_j, \text{ND}, s_P\ , E\ f_j, \text{ND}, s_N\ )$ is given by equation (12):

$$E\ f_j, \text{ND}, e_i\ = \frac{\text{tfidf}\ W_j, ND}{\sum_{l=1}^{z} \text{tfidf}\ W_l, ND} \times \sum_{t_k \square T_{ND}} \text{BMEmoWord}(f_j, e_i, t_k) \quad (12)$$

where $\text{BMEmoWord}(f_j, e_i, t_k)$ denotes the emotion probability of emotion e_i for the feature word f_j giving the topic t_k.$\text{BMEmoWord}(f_j, e_i, t_k)$ is selected in BMEmoWordMod.

The weight of emotion e_i for document ND is computed with equation (13):

$$W_E(ND, e_i) = \sum_{w_j \in W_{ND}} E(f_j, ND, e_i) \quad (13)$$

Equation (29) yields the emotional vector of new document ND

$V_{ND} = (W_E(ND, e_1), \ldots, W_E(ND, e_i), \ldots, W_E(ND, e_E), W_E(ND, s_P), W_E(ND, s_N))$.

Next, the new document ND emotion and sentiment is inferred using a fuzzy logic approach and the emotional vector V_{ND}. The weight of emotion is transformed into five linguistic variables: very low, low, medium, high, and very high. Then, using these variables as input to the fuzzy inference system one obtains the final emotion for the new document.

Sentiment and emotion refining

The refining process validates discovered sentiment and emotion after the document analysis. Similarity is computed between new documents and documents in the corpus rated over E emotions. First, the term vectors of each document are defined using the tf-itf of each term, tf-itf is then computed using equation (1).Note that the terms extracted from the corpus of documents rated over E emotions are those employed by users. To measure the similarity between two documents, the cosine similarity of their representative vectors is computed. Two documents d1 and d2 are similar when the similarity $SimCos(\bar{t}_{d1}, \bar{t}_{d2})$ of these two documents is less than the similarity threshold β.

EVALUATION USING SIMULATIONS

This section presents an evaluation of BM-SATD and BM-SSEA performance using simulations. To perform these simulations, an experimental environment was developed to provide a simulator to prototype the different algorithms of SMESE V3.

Dataset and parameters

To evaluate BM-SATD and BM-SSEA, real datasets from different projects that have digital and physical library catalogues were used. These datasets, consisting of 25,000 documents with a vocabulary of 375,000 words, were selected using average TF-IDF. The documents covered 20 topics and 8 emotions. The number of documents per topic or emotion was approximately equal. The average number of topics per document was 7 while the average rating emotion number per document was 4.15,000 documents of the dataset were used for the training phase and the remaining 10,000 other documents used for the test. Note that the 10,000 documents used for the tests were those that had more annotated topics or a higher rating over emotions.

To measure the performance of topic detection (sentiment and emotion discovery, respectively) approaches, comparison of detected topics (the discovered sentiment and emotion, respectively) with annotation topics of librarian experts (user ratings) were carried out. Table II presents the values of the parameters used in the simulations. The server characteristics for the simulations were: Dell Inc. Power Edge R630 with96 Ghz (4 x Intel(R) Xeon(R) CPU E5-2640 v4 @ 2.40GHz, 10 core and 20 threads per CPU) and 256 GB memory running VMWare ESXi 6.0.

Table II Simulation parameters

Parameter	Value	Parameter	Value
ε	3	α	100
NumKeyTerm	8	co-occurrence threshold	0.75
ω	0.5	semantic threshold	1
β	0.7	term cluster matching threshold	0.45
λ	0.6		

Performance criteria

The performance of BM-SATD and BM-SSEA performance was measured in terms of running time[18] and accuracy[25][5]. Note that in the library domain, the most important criteria was precision while resource consumption was important for the software providers.

The running time, denoted by Rt, was computed as follows:

$$Rt = Et - Bt$$

where *Et* and denotes the time when processing is completed and *Bt* the time when it started.

To compute the accuracy, let $T_{annotated}$ and $T_{detected}$ be the set of annotated topic and the set of detected topics by BM-SATD for a given document d. The accuracy of topics detection, denoted by A_d^t, was computed as follows:

$$A_d^t = \frac{2.|T_{annotated} \cap T_{detected}|}{|T_{annotated}| + |T_{detected}|}$$

The same formula was applied to compute the accuracy of the sentiment and emotion discovery measurement. E_{rating}(resp. $E_{discovered}$) that denotes the set of rating over emotion (resp. the set of discovered emotion by BM-SSEA) was used instead of $T_{annotated}$ (resp. $T_{detected}$).

Simulation results were averaged over multiple runs with different pseudorandom number generator seeds. The average accuracy, *Ave_acc*, of multiple runs was given by:

$$Ave\_acc = \frac{\sum_{x=1}^{I} \frac{\sum_{d \in TD} A_d^t}{|TD|}}{I}$$

where TD denotes the number of tests documents and I denotes the number of test iterations.

The average running time, *Ave_run_time*,was given by:

$$Ave\_run\_time = \frac{\sum_{x=1}^{I} Rt}{I}$$

Topic detection approaches performance evaluation

BM-SATD performance was evaluated in terms of running time and accuracy. The dataset and parameters mentioned above were applied.BM-SATD performance was compared to the approaches described in [25], [5], [7] and [18], referred to as LDA-IG (probabilistic and graph approach), KeyGraph (graph analytical approach), LDA (probabilistic approach)and HLTM (probabilistic and graph approach), respectively.LDA-IG, Key Graph, LDA and HLTM were selected because they are text-based and long text approaches.

Comparison approaches

Table III presents the characteristics of the comparison approaches for topic detection.

Table III Topic detection approaches for comparison

Approach	Granularity	Description	Training phase	Refining	Semantic	Topic correlation	Domain knowledge
LDA-IG [25]	Document	Probabilistic and graph based	Yes	No	No	No	No
KeyGraph [5]	Document	Graph based	Yes	No	No	No	No
LDA [7]	Document	Probabilistic based	No	No	No	No	No
HLTM [18]	Document	Probabilistic and graph based	Yes	No	No	No	No
BM-SATD	Configu-rable as desired	Semantic, probabilistic and graph based	Yes	Yes	Yes	Yes	Yes

Our proposed approach BM-SATD is the only one that is really semantic and takes into account the correlated topic and domain knowledge. The parameters for the comparison approaches used where those which provided the best performance.

Results analysis

Fig. 5presents the average running time of the detection phase when the number of documents used for the tests were varied. Training times were excluded as this phase was performed only one time. However, the BM-SATD training phase required more time than the other approaches. This was justified by the fact that BM-SATD identifies the relevant and less similar documents used for training phase. Fortunately, the new generation of data center equipment offers sufficient resources to reduce significantly the training delay. Only the time required to detect new document topics was measured.

Fig. 5 also shows that the average running time increased with the number of test documents. Indeed, the bigger the number of test documents, the longer the time to perform detection and, ultimately, the higher the average running time.

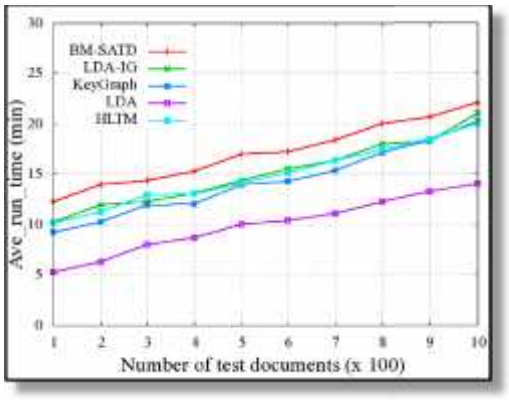

Fig. 5 Topic detection -Average running time versus number of documents for test phase

It was also observed that LDA outperforms the other approaches.LDA produced an average of 1.37 sec per document whereas BM-SATD produced an average of 2.62 secper document.

The average relative improvement (defined as [Aver._runtime of BM-SATD −Aver._runtime of LDA]) of LDA compared with BM-SATD was approximately1.25 sec per document.

The short run times of LDA were due to the fact that LDA did not perform a graph treatment. Graph processing algorithms are very time consuming. Other approaches also outperformed BM-SATD on the running time criteria since BM-SATD performed topic refining in order to increase accuracy.

Fig. 6 shows the average accuracy when varying the number of detected topics. For the five approaches, the average accuracy decreased with the number of detected topics. The increase in the number of subjects to detect led to decreased accuracy. However, in terms of accuracy, BM-SATD outperformed the approaches used for comparison. BM-SATD produced an average accuracy of 79.50% per topic while LDA-IG, the best among the approaches used for comparison, produced an average of 61.01% per topic.

The average relative improvement in accuracy (defined as [Ave_acc of BM-SATD −Ave_acc of LDA-IG]) of BM-SATD compared to LDA-IG was 18.49% per topic. The performance of BM-SATD is explained as follows:

1. BM-SATD used the relevant documents for training phase.
2. BM-SATD refined its detection topic results by measuring new document similarity with relevant and less similar annotated documents.
3. BM-SATD combined correlated topic model and domain knowledge model instead of LDA.

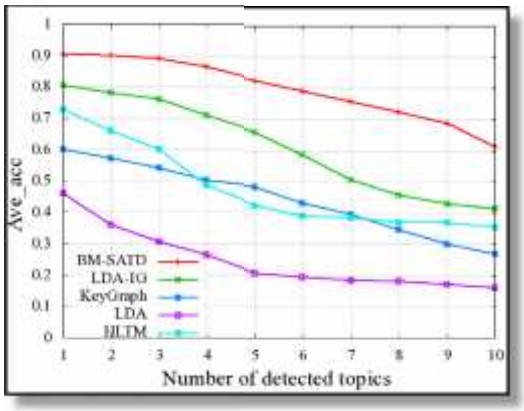

Fig. 6 Accuracy for number of detected topics for 5 comparison approaches

Fig. 6 also shows that BM-SATD produced an average accuracy of 90.32% for one detected topic and 61.27% for ten detected topics compared to 80.29% and 41.01% respectively for LDA-IG. The gap between BM-SATD accuracy and LDA-IG accuracy was 10.03% for one detected topic and 20.26% for ten detected topics. This meant that BM-SATD was by in large more accurate than LDA-IG in detecting several topics.

The Fig. 7presents the average accuracy when varying the number of training documents of the learning phase. LDA was not included in the scenario since not raining phase was performed. Fig. 7 shows that the average accuracy increased with the number of training documents. The larger the number of training documents, the better the knowledge about word distribution and co-occurrence and, ultimately, the higher the detection accuracy. However, the accuracy remained largely stable for very high numbers of training documents. When the number of documents of a collection was larger, the number of vocabulary words remained constant, and the term graph did not change. It also shows that HLTM was the approach whose detection accuracy was the first to reach stability at 10,000 training documents. HLTM builds a tree instead of a graph as the other approaches and its tree has less internal roots to identify topics. However, BM-SATD and LDA-IG outperformed HLTMin terms of accuracy.

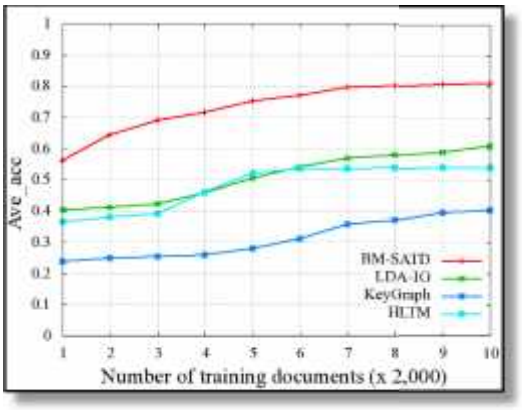

Fig. 7 Topic detection - accuracy for number of training documents

Fig. 7 also shows that BM-SATD outperformed LDA-IG on the accuracy criteria. For example, BM-SATD demonstrated an average accuracy of 73.49% per 2,000 training documents while LDA-IG produced an average accuracy of 50.86% per 2,000 training documents.The average relative improvement of BM-SATD compared to LDA-IG was 22.63% per 2,000 training documents. The better performance of BM-SATD followed from its use of a domain knowledge model. BM-SATD did not requirea large number of documents for the training phase. In conclusion, the 1.25 sec running time per document increase was a small price to pay for the larger average accuracy of topic detection (18.49%).

Sentiment and emotion analysis performance evaluation

BM-SSEA performance was also evaluated in terms of accuracy and running time. Simulations used the dataset and parameters mentioned previously. The performance of BM-SSEA was compared to the approaches described in [49] and [41], referred to as ETM-LDA and AP, respectively. ETM-LDA and AP were selected because they were document-based rather than phrase-based.

Comparison of approaches with BM-SSEA

Table IV shows the characteristics of the sentiment and emotion approaches used for comparison with BM-SSEA.

BM-SSEA was the only entirely semantic approach taking into account the rules for inferring emotion. In addition, BM-SSEA used a semantic lexicon. Several approaches used semantic lexicon, but these were limited to phrases rather than documents. The best performance approaches used were AP and ETM_LDA.

Results analysis

Fig. 8presents the average running time when varying the number of detected emotions. Training times were excluded because this phase was performed only once. The BM-SSEA training phase took more time than the other approaches due to lexicon aggregation and enrichment by users. The average running time increased with the number of test documents. This is normal, as the larger the number of test documents the longer the average running time to perform the sentiment and emotion discovery. Fig. 8 shows that ETM-LDA and AP outperformed BM-SSEA on the running time criteria. ETM-LDA required an average of 1.53 sec per document whereas BM-SSEA required an average of 1.74 sec per document. The average relative improvement of ETM-LDA compared with BM-SSEA was approximately 0.21 sec per document. The poorer performance of BM-SSEA resulted from refining sentiment and emotion to increase accuracy.

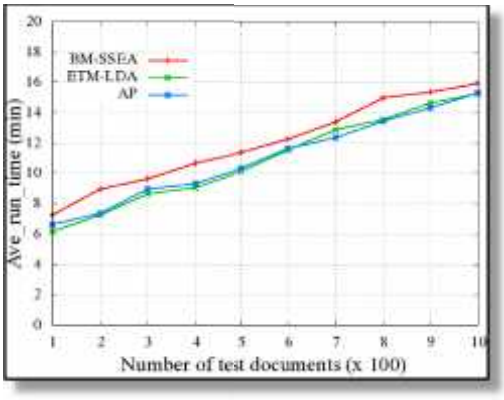

Fig. 8 Emotion discovery -Average running time versus number of documents for test phase

Table IV Sentiment and emotion approaches for comparison

Approach	Granularity	Approach	Training phase	Refining	The saurus	Topic modeling	Emotions number
AP [41]	Document	Learning based	Yes	No	5	No	8
ETM-LDA [49]	Document	Keyword based	Yes	No	6	Yes	8
BM-SSEA	Configurable as desired	Keywordand rule based	Yes	Yes	1,2,3, and 4	Yes	8

1-WordNet; 2-WordNet-Affect; 3-SentiWordNet; 4-NRC Emotion Lexicon; 5- Stanford Core NLP; 6-Gibbs sampling.

Fig. 9 presents the average accuracy when varying the number of discovered emotions. Positive and negative sentiments were not considered in the accuracy measurement. Fig. 9also shows that the average accuracy decreased with the number of discovered emotions. However, BM-SSEA outperformed the other two approaches used for comparisons. BM-SSEA demonstrated an average accuracy of 93.30% per emotion while ETM-LDA, the best of the other two approaches used for comparison, produced68.65% accuracy per emotion. The average relative improvement in accuracy of BM-SSEA compared to ETM-LDA was 24.65% per emotion. In conclusion, the 0.21 sec running time per document increase was, again, a small price to pay for the larger average accuracy of emotion discovery (24.65%).

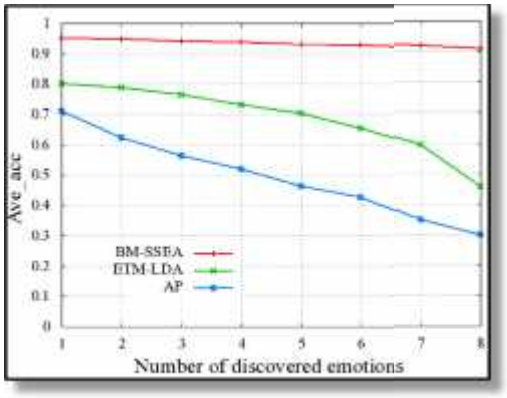

Fig. 9 Average detection accuracy for the number of discovered emotions

SUMMARY AND FUTURE WORK

In this paper, the goal was to increase the find ability (search, discover) of entities based on user interest using external and internal semantic metadata enrichment algorithms. As computers struggle to understand the meaning of natural language, enriching entities semantically with meaningful metadata can improve search engine capability. Words themselves have a wide variety of definitions and interpretations and are often utilized inconsistently. While topics and sentiments/emotions may have no relationship to individual words, thesauri express associative relationships between words, on tologies, entities and a multitude of relationships represented as triplets.

This paper has presented an enhanced V3 implementation of SMESE using metadata and data from the linked open data, structured data, metadata initiatives, concordance rules and authority metadata to create a master catalogue. It offers a foundation for an entire interest-based digital library of semantic mining activities, such as search, discovery and interest-based notifications. Finding bibliographic references or semantic relationships in texts makes it possible to localize specific text segments using on tologies to enrich a set of semantic metadata related to topic or sentiment/emotion.

To help users find interest-based contents, this paper has proposed an enhanced verions of the SMESE platform through text analysis approaches for sentiments/emotions detection. SMESE V3 can be used (or: makes it possible) to create and use a semantic master catalogue with enriched metadata that enables search and discovery interest-based engines. This paper has presented the design, implementation and evaluation of a SMESE V3 platform using metadata and data from the web, linked open data, harvesting and concordance rules, and bibliographic record authorities. The SMESE V3includes three distinct engines to:

1. Discover enriched sentiment/emotion metadata hidden within the text or linked to multimedia structure using the proposed BM-SSEA (BM-Semantic Sentiment and Emotion Analysis) algorithm.
2. Implement rule-based semantic metadata internal enrichment.
3. Propose a hybrid machine learning model for metadata enrichment.
4. Generate semantic topics by text, and multimedia content analysis using the proposed BM-SATD (BM-Scalable Annotation-based Topic Detection) algorithm.

The semantic aggregation of metadata content repository offers a foundation for an interest-based digital library of semantic mining activities, such as search, discover and smart notifications.

Table 1 shows the comparison with most known text mining algorithms (e.g., AlchemyAPI, DBpedia, Wikimeta, Open Calais, Bitext, AIDA, TextRazor) and a new algorithm SMESE with many attributes including keyword extraction,

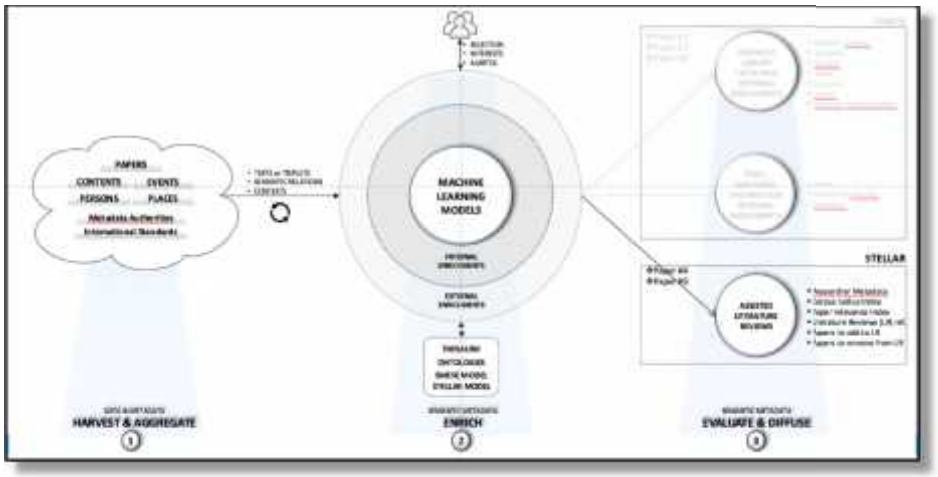

Fig. 10 Future work: Semantic Topics Ecosystem Learning-based Literature Assistant Review

concept extraction. It was noted that SMESE algorithms support more attributes than any other algorithms.

In future work, the focus will be to generate learning-based literature review enrichment and abstract of abstract.STELLAR (Semantic Topics Ecosystem Learning-based Literature Assistant Review) assess each citation to determine her ranking and her inclusion in the final literature assistant review. One goal of this enhanced ecosystem will be to reduce reading load by helping researcher to read only an intelligent selection of documents, using text data mining, machine learning, and a classification model that learn from users annotated data and detected metadata (see Fig. 10).

References

1. O. Appel, F. Chiclana, J. Carter, and H. Fujita, "A hybrid approach to the sentiment analysis problem at the sentence level," *Knowledge-Based Systems*, vol. 108, pp. 110-124, 2016. doi:http://dx.doi.org/10.1016/j.knosys.2016.05.040
2. G. Salton, and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information Processing & Management*, vol. 24, no. 5, pp. 513-523, 1988. doi:http://dx.doi.org/10.1016/0306-4573(88)90021-0
3. T. Niu, S. Zhu, L. Pang, and A. El Saddik, "Sentiment Analysis on Multi-View Social Data," in 22nd International Conference on MultiMedia Modeling (MMM), Miami, FL, USA, 2016, pp. 15-27. doi:http://dx.doi.org/10.1007/978-3-319-27674-8_2
4. K. Bougiatiotis, and T. Giannakopoulos, "Content Representation and Similarity of Movies based on Topic Extraction from Subtitles," in Proceedings of the 9th Hellenic Conference on Artificial Intelligence, Thessaloniki, Greece, 2016, pp. 1-7. doi:http://dx.doi.org/10.1145/2903220.2903235
5. H. Sayyadi, and L. Raschid, "A Graph Analytical Approach for Topic Detection," *ACM Transactions on Internet Technology*, vol. 13, no. 2, pp. 1-23, 2013. doi:http://dx.doi.org/10.1145/2542214.2542215
6. J. L. Hurtado, A. Agarwal, and X. Zhu, "Topic discovery and future trend forecasting for texts," *Journal of Big Data*, vol. 3, no. 1, pp. 1-21, 2016. doi:http://dx.doi.org/10.1186/s40537-016-0039-2
7. D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichl*et Al*location," *Journal of Machine Learning Research*, vol. 3, pp. 993-1022, 2003
8. V. Bijalwan, V. Kumar, P. Kumari, and J. Pascual, "KNN based Machine Learning Approach for Text and Document Mining," *International Journal of Database Theory and Application*, vol. 7, no. 1, pp. 61-70, 2014. doi:http://dx.doi.org/10.14257/ijdta.2014.7.1.06
9. G. A. Patel, and N. Madia, "A Survey: Ontology Based Information Retrieval For Sentiment Analysis," *International Journal of Scientific Research in Science, Engineering and Technology*, vol. 2, no. 2, pp. 460-465, 2016
10. J. A. Balazs, and J. D. Velásquez, "Opinion Mining and Information Fusion: A survey," *Information Fusion*, vol. 27, pp. 95-110, 2016. doi:http://dx.doi.org/10.1016/j.inffus.2015.06.002
11. K. Ravi, and V. Ravi, "A survey on opinion mining and sentiment analysis: Tasks, approaches and applications," *Knowledge-Based Systems*, vol. 89, pp. 14-46, 2015. doi:http://dx.doi.org/10.1016/j.knosys.2015.06.015
12. J. Serrano-Guerrero, J. A. Olivas, F. P. Romero, and E. Herrera-Viedma, "Sentiment analysis: A review and comparative analysis of web services," *Information Sciences*, vol. 311, pp. 18-38, 2015. doi:http://dx.doi.org/10.1016/j.ins.2015.03.040
13. M. Taboada, J. Brooke, M. Tofiloski, K. Voll, and M. Stede, "Lexicon-based methods for sentiment analysis," *Comput. Linguist.*, vol. 37, no. 2, pp. 267-307, 2011. doi:10.1162/COLI_a_00049
14. D. Vilares, M. A. Alonso, and C. GÓMez-RodrÍGuez, "A syntactic approach for opinion mining on Spanish reviews," *Natural Language Engineering*, vol. 21, no. 1, pp. 139-163, 2015. doi:http://dx.doi.org/10.1017/S1351324913000181
15. S. Kiritchenko, X. Zhu, and S. M. Mohammad, "Sentiment analysis of short informal texts," *Journal of Artificial Intelligence Research*, vol. 50, no. 1, pp. 723-762, 2014. doi:http://dx.doi.org/10.1613/jair.4272
16. S. T. Dumais, "Latent semantic analysis," *Annual Review of Information Science and Technology*, vol. 38, no. 1, pp. 188-230, 2004. doi:10.1002/aris.1440380105
17. J. Cigarrán, Á. Castellanos, and A. García-Serrano, "A step forward for Topic Detection in Twitter: An FCA-based approach," *Expert Systems with Applications*, vol. 57, pp. 21-36, 2016. doi:http://dx.doi.org/10.1016/j.eswa.2016.03.011
18. P. Chen, N. L. Zhang, T. Liu, L. K. M. Poon, and Z. Chen, "Latent Tree Models for Hierarchical Topic Detection," *arXiv preprint arXiv:1605.06650 [cs.CL]*, pp. 1-44, 2016
19. R. Moraes, J. F. Valiati, and W. P. Gavião Neto, "Document-level sentiment classification: An empirical comparison between SVM and ANN," *Expert Systems with Applications*, vol. 40, no. 2, pp. 621-633, 2013. doi:http://dx.doi.org/10.1016/j.eswa.2012.07.059
20. M. Ghiassi, J. Skinner, and D. Zimbra, "Twitter brand sentiment analysis: A hybrid system using n-gram analysis and dynamic artificial neural network," *Expert Systems with Applications*, vol. 40, no. 16, pp. 6266-6282, 2013. doi:http://dx.doi.org/10.1016/j.eswa.2013.05.057
21. R. Brisebois, A. Abran, and A. Nadembega, "A Semantic Metadata Enrichment Software Ecosystem (SMESE) based on a Multi-platform Metadata Model for Digital Libraries," *Accepted for publication in Journal of Software Engineering and Applications (JSEA)*, vol. 10, no. 04, 2017
22. Q. Dang, F. Gao, and Y. Zhou, "Early detection method for emerging topics based on dynamic bayesian networks in micro-blogging networks," *Expert Systems with Applications*, vol. 57, pp. 285-295, 2016. doi:http://dx.doi.org/10.1016/j.eswa.2016.03.050
23. J. M. Cotelo, F. L. Cruz, F. Enríquez, and J. A. Troyano, "Tweet categorization by combining content and structural knowledge," *Information Fusion*, vol. 31, pp. 54-64, 2016. doi:http://dx.doi.org/10.1016/j.inffus.2016.01.002

24. T. Hashimoto, T. Kuboyama, and B. Chakraborty, "Topic extraction from millions of tweets using singular value decomposition and feature selection," in 2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA), Hong Kong, China, 2015, pp. 1145-1150. doi:http://dx.doi.org/10.1109/APSIPA.2015.7415451
25. C. Zhang, H. Wang, L. Cao, W. Wang, and F. Xu, "A hybrid term-term relations analysis approach for topic detection," *Knowledge-Based Systems*, vol. 93, pp. 109-120, 2016. doi:http://dx.doi.org/10.1016/j.knosys. 2015.11.006
26. A. A. Salatino, and E. Motta, "Detection of Embryonic Research Topics by Analysing Semantic Topic Networks," in Semantics, Analytics, Visualisation: Enhancing Scholarly Data, Montreal, Quebec, Canada, 2016, pp. 1-15
27. S. N. Shivhare, and S. Khethawat, "Emotion Detection from Text," in Second International Conference on Computer Science, Engineering and Applications (ICCSEA), Delhi, India, 2012, pp. 1-7
28. A. Moreo, M. Romero, J. L. Castro, and J. M. Zurita, "Lexicon-based Comments-oriented News Sentiment Analyzer system," *Expert Systems with Applications*, vol. 39, no. 10, pp. 9166-9180, 2012. doi:http://dx.doi.org/10.1016/j.eswa.2012.02.057
29. C. Bosco, V. Patti, and A. Bolioli, "Developing corpora for sentiment analysis: The case of irony and senti-tut," *IEEE Intelligent Systems*, vol. 28, no. 2, pp. 55-63, 2013
30. H. Cho, S. Kim, J. Lee, and J.-S. Lee, "Data-driven integration of multiple sentiment dictionaries for lexicon-based sentiment classification of product reviews," *Knowledge-Based Systems*, vol. 71, pp. 61-71, 2014. doi:http://dx.doi.org/10.1016/j.knosys.2014.06.001
31. C. Lin, Y. He, R. Everson, and S. Ruger, "Weakly Supervised Joint Sentiment-Topic Detection from Text," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 6, pp. 1134-1145, 2012. doi:http://dx.doi.org/10.1109/TKDE.2011.48
32. E. Kontopoulos, C. Berberidis, T. Dergiades, and N. Bassiliades, "Ontology-based sentiment analysis of twitter posts," *Expert Systems with Applications*, vol. 40, no. 10, pp. 4065-4074, 2013. doi:http://dx.doi.org/10.1016/j.eswa.2013.01.001
33. B. Desmet, and V. Hoste, "Emotion detection in suicide notes," *Expert Systems with Applications*, vol. 40, no. 16, pp. 6351-6358, 2013. doi:http: //dx. doi. org/ 10. 1016/j.eswa.2013.05.050
34. M. Abdul-Mageed, M. Diab, and S. Kübler, "SAMAR: Subjectivity and sentiment analysis for Arabic social media," *Computer Speech & Language*, vol. 28, no. 1, pp. 20-37, 2014. doi:http://dx.doi.org/10.1016/j.csl.2013.03.001
35. L. K.-W. Tan, J.-C. Na, Y.-L. Theng, and K. Chang, "Phrase-Level Sentiment Polarity Classification Using Rule-Based Typed Dependencies and Additional Complex Phrases Consideration," *Journal of Computer Science and Technology*, vol. 27, no. 3, pp. 650-666, 2012. doi:http://dx.doi.org/10.1007/s11390-012-1251-y
36. L. Chen, L. Qi, and F. Wang, "Comparison of feature-level learning methods for mining online consumer reviews," *Expert Systems with Applications*, vol. 39, no. 10, pp. 9588-9601, 2012. doi:http://dx.doi.org/10.1016/ j.eswa.2012.02.158
37. C. Quan, and F. Ren, "Unsupervised product feature extraction for feature-oriented opinion determination," *Information Sciences*, vol. 272, pp. 16-28, 2014. doi:http://dx.doi.org/10.1016/j.ins.2014.02.063
38. S. Poria, E. Cambria, A. Hussain, and G.-B. Huang, "Towards an intelligent framework for multimodal affective data analysis," *Neural Networks*, vol. 63, pp. 104-116, 2015. doi:http://dx.doi.org/10.1016/ j.neunet.2014.10.005
39. M. D. Munezero, C. S. Montero, E. Sutinen, and J. Pajunen, "Are They Different? Affect, Feeling, Emotion, Sentiment, and Opinion Detection in Text," *IEEE Transactions on Affective Computing*, vol. 5, no. 2, pp. 101-111, 2014. doi:http://dx.doi.org/10.1109/TAFFC. 2014.2317187
40. W. Li, and H. Xu, "Text-based emotion classification using emotion cause extraction," *Expert Systems with Applications*, vol. 41, no. 4, Part 2, pp. 1742-1749, 2014. doi:http://dx.doi.org/10.1016/j.eswa.2013.08.073
41. V. Anusha, and B. Sandhya, "A Learning Based Emotion Classifier with Semantic Text Processing," *Advances in Intelligent Informatics*, M. E.-S. El-Alfy, M. S. Thampi, H. Takagi, S. Piramuthu and T. Hanne, eds., pp. 371-382, Cham, Switzerland: Springer International Publishing, 2015. doi:http://dx.doi.org/10.1007/978-3-319-11218-3_34
42. E. Cambria, P. Gastaldo, F. Bisio, and R. Zunino, "An ELM-based model for affective analogical reasoning," *Neurocomputing*, vol. 149, Part A, pp. 443-455, 2015. doi:http://dx.doi.org/10.1016/j.neucom.2014.01.064
43. M. F. Porter, "An algorithm for suffix stripping," *Program*, vol. 14, no. 3, pp. 130-137, 1980. doi:doi:10.1108/eb046814
44. D. M. Blei, and J. D. Lafferty, "Correlated Topic Models," in Proceedings of the 19th Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, 2005, pp. 1-8
45. D. Andrzejewski, X. Zhu, and M. Craven, "Incorporating domain knowledge into topic modeling via Dirichlet Forest priors," in Proceedings of the 26th Annual International Conference on Machine Learning, Montreal, Quebec, Canada, 2009, pp. 25-32. doi:http://dx.doi.org/10.1145/1553374.1553378
46. T. Pedersen, S. Patwardhan, and J. Michelizzi, "WordNet::Similarity: measuring the relatedness of concepts," in Demonstration Papers at Human Language Technology conference/North American chapter of the Association for Computational Linguistics (HLT-NAACL), Boston, Massachusetts, USA, 2004, pp. 38-41
47. Y. Tsuruoka, and J. i. Tsujii, "Bidirectional inference with the easiest-first strategy for tagging sequence data," in Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing, Vancouver, British Columbia,

Canada, 2005, pp. 467-474. doi:10.3115/ 1220575.1220634

48. de Marneffe M-C, MacCartney B, and Manning CD, "Generating typed dependency parsers from phrase structure parses " in fifth international conference on language resources and evaluation, GENOA , ITALY 2006, pp. 449-54
49. S. Bao, S. Xu, L. Zhang, R. Yan, Z. Su, D. Han, and Y. Yu, "Mining Social Emotions from Affective Text," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 9, pp. 1658-1670, 2012. doi:http://dx.doi.org/10.1109/TKDE.2011.188

How to cite this article:

Ronald Brisebois *et al*.2017, A Semantic Metadata Enrichment Software Ecosystembased on Machine Learning to Analyze Topic, Sentiment and Emotions. *Int J Recent Sci Res.* 8(4), pp. 16698-16714.
DOI: http://dx.doi.org/10.24327/ijrsr.2017.0804.0200

****