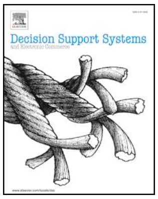
Contents lists available at ScienceDirect

Decision Support Systems

journal homepage: www.elsevier.com/locate/dss



Developing insights from social media using semantic lexical chains to mine short text structures

Cecil Eng Huang Chua[a,*], Veda C. Storey[b], Xiaolin Li[c], Mala Kaul[d]

[a] Department of Business and Information Technology, Missouri University of Science and Technology, USA
[b] Department of Computer Information Systems, J. Mack Robinson College of Business, Georgia State University, USA
[c] Xtracta, level 6/45 O'Rorke Rd, Penrose, Auckland 1061, New Zealand
[d] Department of Information Systems, College of Business, University of Nevada, Reno, 1664 N. Virginia Street, Reno, NV 89557, USA

ARTICLE INFO

Keywords:
Social media
Semantic clustering
Lexical chain
Short text
Text mining
Word sense disambiguation

ABSTRACT

Social media is increasingly being used for communication by individuals and organizations. Social media stores vast amounts of publicly available data that provides a rich source of information and insights. Often, social media users can easily infer meaning from short text such as microblogs and Facebook posts because they understand the context and terminology used. Although automated data-mining can be effective for gaining insights from text data, a significant challenge is to accurately infer meaning from social media text derived from a single social media account. This is difficult because social media communication uses very short, or sparse, text, which yields a relatively small sample of usable words for analysis. Furthermore, interpreting the contextual meaning from a relatively small set of words is challenging. This research proposes a methodology for extracting semantic lexical chains from frequently occurring words in a single social media account and using these chains to mine short text structures to infer the overall themes of the user. The methodology is based on a proposed clustering algorithm and illustrated with examples from Facebook posts. The algorithm is tested and illustrated by comparing it to existing work and further applying it to a variety of news posts. This methodology could be useful for gaining decision-making insights from social media, or other online forms with short or sparse text.

1. Introduction

The amount of user-generated text in social media, and other sources, continues to grow at a very rapid rate. This information is especially useful for applications such as sentiment analysis that tries to infer a user's (e.g., customer, politician) general attitude towards an event or product [1]. Not surprisingly, research on text mining has become increasingly important [2] because using social media data for enhanced decision-making has practical applications in many domains.

An example of social media use is to systematically categorize customer feedback. Businesses frequently use social media platforms to receive customer complaints and/or feedback [3]. The people managing these accounts are often too busy responding to, or remedying, feedback to be able to systematically identify patterns or clusters of feedback to address the root cause of customer complaints [4]. Manual review and analysis of social media data is time consuming, so a (semi-) automated approach to analyzing the data could reduce response times and labor costs [5,6]. Similarly, bloggers, worried about their own viewership, might want to know which of the myriad topics about which they post, draws an audience. Clustering their individual posts is the first step to identifying and analyzing the topics that draw a crowd.

Another application where interpreting social media information could be very useful, is the analysis of politician social comments [7]. Open-source intelligence, the collection of public information of value [8], is important to organizations and nations world-wide. Diplomats, for example, often seek a deep understanding of the people with whom they must negotiate. This includes knowing who else the other party has talked to, when, where, and on what topic. Journalists, similarly, often analyze politician social media posts for patterns and trends. Throughout this paper, we employ a politician's social media posts as a recurring example.

Existing approaches to text-mining require a large text corpus (i.e., set of sentences) to provide meaningful information. Unfortunately, social media data is both unstructured, and short, leading to the problem of "short text," which makes this data challenging to efficiently mine for acceptable actions. A particularly challenging variant of this

[*] Corresponding author.
E-mail addresses: cecq8z@mst.edu (C.E.H. Chua), vstorey@gsu.edu (V.C. Storey), xiaolin.li@xtracta.com (X. Li), mkaul@unr.edu (M. Kaul).

https://doi.org/10.1016/j.dss.2019.113142
Received 19 February 2019; Received in revised form 15 July 2019; Accepted 21 August 2019

problem is deriving such information from a single social media account (i.e., of one person or organization). Such information would be useful for building a profile of an individual, or for an organization trying to manage its social media presence. Therefore, one problem is how to derive meaning from inferred semantic relationships among words in short text (or sentence fragments) in social media from a single individual or organization. Although prior work has addressed some short text issues, there has not been much research on short text clustering. Furthermore, existing algorithms cannot be easily adapted to short text problems.

The objective of this research is to: address the problem of interpreting short text structures within social media content to extract useful information from a single user account. To do so, a methodology for short text mining is developed, which is based on lexical chaining algorithms in conjunction with Word Sense Disambiguation. The methodology employs synsets (sets of cognitive synonyms from WordNet [9]) to produce lexical chains of words. Each lexical chain represents one synset (word meaning) with all words in the chain having that word meaning. Posts that have words corresponding to that lexical chain are then clustered together and considered as posts with similar themes.

The contribution of this research is a methodology for deriving semantic meaning from short text when no meta-data is available, by creating lexical chains and developing a text clustering system to infer the overall meaning of the chains. The methodology is demonstrated by developing the profile of a politician using their publicly available Facebook posts. Such profiles can be useful for diplomats to gain an understanding of their counterparts, with whom they must communicate. The methodology is also applied to a subset of the 20 newsgroup data set [10]. To evaluate the methodology, it is compared to two current state-of-the-art clustering algorithms: non-negative matrix factorization (NMF) [11] and the Gibbs Sampling algorithm for the Dirichlet Multinomial Mixture (GSDMM) [12]. The results show that our methodology produces clusters of better quality for short texts.

This paper proceeds as follows. Section 2 reviews related research on social media short text and lexical chaining. Our proposed methodology is presented in Section 3. Section 4 presents an empirical evaluation. Section 5 summarizes and concludes the paper.

2. Related research

This section describes the short text problem and reviews related research to understand existing approaches and identify their limitations. Table 1 first defines the terms used in this research.

2.1. Short social media texts

All social media, in essence, employs short text. Twitter, for example, limits the number of characters in each tweet. After removing stopwords such as “the,” “and,” etc., a typically Facebook post contains approximately 11 distinct words [20].

A crucial aspect of online content, beyond the actual data itself, is metadata, which is specific information about the content. Metadata includes tags, location, and time-related informations, classifications of the data etc. These can be extremely useful for developing insights about related data (X. [21]). However, most types of metadata provide information about the links between types of data, rather than the meaning of particular social media short texts. Tags are specific types of metadata that can classify data that is similar, thus providing users the ability to search for related content. Hash-tags are a specific type of tag used by several social media websites that users can follow to help them find content pertaining to a keyword or topic. These could potentially link semantically similar data on a social media website.

Without such hash-tags, it becomes difficult to extract meaning from, or to group, social media text into meaningful clusters. Nevertheless, the majority of social media posts do not have hash-tags. For example, most customer complaints on corporate social media sites do not have a hash-tag; similarly, most posts by politicians describe daily activities or meetings without hash-tags. Without this type of meta-data, it is difficult to effectively aggregate the short texts to find meaning through systematic patterns. The inherent problem stems from the fact that short texts contain little information, making it difficult to find meaning simply by cross-comparing common information found in documents such as Facebook posts [22].

Consider, for example, the following two social media posts.

Great to recognise the global success of Kiwi businesses at last night's New Zealand International Business Awards in Auckland.
We're investing in innovative tourism projects to create new opportunities in the sector.

Both of these posts relate to creating business opportunities in New Zealand. Although semantically meaningful to a reader, these post are difficult to mine using traditional text mining approaches [23]. The challenge of semantically relating posts, which do not share the same key words, is referred to as “bridging the semantic gap in short social media texts” [24]. The remainder of this section reviews work that explores the semantic gap problem, demonstrating why existing work is difficult to apply to short text and what elements of existing research help to address it.

2.2. Approaches to extract meaning from social media text

In the text-mining literature, text less than 60 words is considered “short” [25,26]. Existing approaches to partially addressing challenges of extracting meaning from related text (which are difficult to apply to short text) are summarized in Table 2.

Table 1
Relevant terms.

Term	Description
Application domain	Problem space of a situation for which an information system needs to be developed (adapted from [13]).
Clustering algorithms	Computational algorithms that try to group a set of objects in such a way that objects in the same group (called a cluster) are more similar (in some sense or another) to each other than to those in other groups (clusters).
Lexicon	A dictionary.
Ontology	Explicit specification of a conceptualization; defines a set of formal constructs and the relationships among them [14]. Provides a means for the communication between different (software) agents on a semantic level [15]. Intended to facilitate certain kinds of automated reasoning.
Lexical chain	Sequences of semantically related words in a text [16].
Semantic relationship	Relationships with specialized, and recognized meaning. Examples are synonymy, antonymy, hypernymy andhyponymy.
Short text	Unstructured texts with short sentences.
Social media	Applications that enable the creation and exchange of user generated content [17].
Stopwords	Words that are filtered out before or after processing of natural language data (text) [18].
Word sense disambiguation (WSD)	Ability to identify the meaning of words in context in a computational manner [19].

Table 2
Approaches to extracting social media text.

Approach	Technique	References	Limitations
Probabilistic topic model (PTM)	Probabilistic algorithms to discover hidden thematic structures in large archives of documents	([27–31])	Assumes every word has only a single meaning in the corpus; model usually designed for long texts.
	Supervised methods	([32–34])	Makes strong assumption that words have only one meaning; not true in social media posts.
Topic detection in social media	Document-pivot methods which cluster documents using the vector space model	([11,26,35–38])	Documents with synonyms scattered into different clusters; most social media posts untagged.
	Feature-pivot methods which cluster frequently occurring pre-specified key words	([39,40])	Finding frequently occurring key words not applicable for social media with single account, because there are few frequently repeated words.
Dictionary based methods	Understanding the meaning of short texts using an ontology-based approach (e.g. WordNet or Wikipedia)	([37,41–47])	Adding lexicon terms (e.g. hyponyms) may increase the noise in document; mapping Wikipedia concepts requires great deal of processing effort.
Lexical chaining based method	Document clustering based on lexical chains	([48,49])	Current lexical chaining algorithms not designed for short texts.

2.2.1. Probabilistic topic models

Algorithms exist that calculate the probability that a sentence is in a topic based on the relative frequency of words in a document versus those words in the corpus. Examples include probabilistic latent semantic analysis (pLSA) [29] and latent Dirichlet allocation (LDA) [27,50,51]. Although these algorithms work for the general case of learning from text, they require large documents, and, therefore, are not useful for extracting meaningful insights from short texts. There are, of course, some algorithms that work for shorter documents containing approximately 60 words. For example, Kumar and Sridhar [52] propose an unsupervised probabilistic topic model for short text clustering. However, these algorithms do not work for documents with a length of 11 words or less. Kumar and Sridhar [52] combine the documents and generate a continuous-bag-of-words model [30] to process all windows of words of a particular length. Because their technique relies purely on syntactic information, their method cannot link sentences semantically as needed for the two posts in the motivating example on creating business opportunities in New Zealand where the posts share no common words.

Other research incorporates external knowledge into probabilistic topic models. Such approaches [32–34] impose constraints so that words identified by the probabilistic topic model should always appear together, whereas certain words identified as dissimilar by the probabilistic topic model are not allowed to appear at the same time in one topic. This makes it impractical to mine short text for semantic insight.

The word2vec ([28,31], and other neural-network or machine-learning based techniques, derive an understanding of words by first being trained on a training set before being applied to the problem domain. The accuracy of these learning methods often depends on the size and diversity of the training set [30], so the process is not practical for use on diverse data sets without training.

The main disadvantage of these kinds of probabilistic topic models is that they assume every word has only a single sense in the corpus, although this is not always the case. For example, apple and orange have similar representativeness when these words to refer to fruit. However, for computer workers, Apple (the company) is, obviously, a more relevant word than orange.

External knowledge probabilistic topic models exist that work with multiple word senses. For example, [53]) use a Markov Random Field to identify similar relationships among words from each document by attempting to differentiate the senses of words within the same document. Unfortunately, these methods of assessing context require many words, which is not conducive to working with social media posts.

2.2.2. Topic detection in social media

Topic detection is also used to analyze social media texts. It aims to extract topics from a stream of textual information sources (documents) and quantify their “trend” in time. Methodologically, general-purpose topic detection can produce two types of complementary outputs [39]. Either the documents in the collection are clustered (using document-pivot methods); or the most important terms or keywords are selected and then clustered (using feature-pivot methods).

Simple document-pivot methods cluster documents using similarity metrics. Phuvipadawat and Murata [54], for example, use vector space representation, and boost the weights of term frequency-inverse document frequency (tf-idf) for entities such as names of countries or public figures. However, because these techniques treat synonyms as distinct, documents concerning the same topic are still arranged into different clusters. Vector space representations rely on co-ocurrence patterns of words. Words that have similar meanings, but are not the same word, are not considered to be co-occurring words.

The majority of document-pivot methods cluster documents using the vector space model (otherwise called the “Bag of Word” model). The K-means clustering algorithm and its variants are the most popular algorithms [1]. A substantial literature on document-pivot methods leverages meta-data. Most such work relies on hash-tags, often denoted as keywords prefixed with the “#” symbol. For example, Rosa et al. [55] use hash-tags to train a Rocchio classifier. Petkos, Papadopoulos, and Kompatsiaris [56] cluster tweets based on two heuristics: (1) tweets that contain the same URL refer to the same topic; and (2) a tweet and its reply refer to the same topic. However, since most social media posts are untagged, an effective technique must work even without tagging. Thus, in this research, we strive to develop a technique that works, even without tagging.

Feature-pivot methods identify topics of interest a priori and attempt to identify clusters of conversations about that topic in a large pool of data. Sakaki et al. [57], for example, develop algorithms to detect earthquakes in real time and predict their locations based on tweets. Their main approach is a support vector machine (SVM) classification. Their approach targets specific events of a massive scale such as earthquakes or typhoons, focusing on targeted keywords (e.g., “earthquake” or “typhoon”) for processing. When tracked in real-time, the sudden formation of a cluster of conversations with the keyword “earthquake” can indicate that an earthquake is happening somewhere in the world [58]. However, a particular user account can write about a wide range of topics. A politician could write about their hobbies, visits, or a current political issue, so specific keywords cannot be determined in advance. In contrast, on an ordinary day, a politician, such as John Key, does not have more than 10 posts (with an average length less than 25 words). Such posts need not be on the same topic.

2.2.3. Dictionary based methods

A large literature reports on attempts to understand the meaning of short texts using an ontology-based approach. Most such research utilize WordNet or Wikipedia: Dave et al. [41] employ synsets from WordNet as features for clustering. They found that WordNet synsets decreased clustering performance. Hotho et al. [42] show that adding 5 hypernyms of each word from WordNet to enrich the vector space representation can improve clustering performance. However, adding ontology terms (e.g. hyponyms) may also increase the noise in the document [43,44,47].

Banerjee et al. [35] and Gabrilovich [59] use Wikipedia to improve the performance of clustering short texts, but their approaches requires a great deal of processing effort. Other efforts use Wikipedia titles as features [37,43,44], but the performance of the current all-word unsupervised word sense disambiguation only has a precision of 58.83% and 56.37% for recall [60]. The low retrieval rate severely limits the performance of such algorithms. Similarly, Oliva et al. [46] develop a method for assessing whether two sentences communicate the same meaning. Their method employs WordNet similarity coupled with other syntactic information such as word order. Their technique, though, is too specialized for the problem of extracting meaning from short text, because it is used to identify sentences that are paraphrases, rather than sentences covering the same area.

Liu et al. [45] provide a method for associating pairs of short texts. Their method boosts the similarity of two sentences if there are overlapping words. However, as noted above, this does not work on the kinds of short text being analyzed in this research because, in most cases, two sentences covering the same issue have no words in common.

2.2.4. *Lexical chaining based methods*

Lexical chains are sequences of semantically related words in a text [16]. They are useful for our short text problem, because they integrate external knowledge with different senses of words, and have been used extensively for keyword extraction [61], text summarization [62], malapropisms detection [63] and clustering [48,49].

One of the main difficulties in generating lexical chains is that words can have different senses. For example, the word "lion" in the post "Chinese New Year celebrations at Parliament last night came complete with lions" could refer to an animal or a celebrity, depending upon the context. Therefore, developing a lexical chain requires that an algorithm be integrated to perform Word Sense Disambiguation (WSD).

An example of a lexical chain for an education application is given below.

"school^{3}↔education^{3}↔ learning^{1}↔schooling^{2}↔study^{2}"

The superscript on each word indicates the definition number for the word in a dictionary such as WordNet. For example, "school^{3}" means the third meaning for the word "school." The word can refer to a physical structure for education, the act of educating people, a grouping of fish, etc. Each definition is given a number.

A crucial part of lexical chaining algorithms is Word Sense Disambiguation (WSD), which is the ability to identify the meaning of words within a context in a computational manner [19]. There are two main ways to apply Word Sense Disambiguation: (1) perform it during preprocessing; or (2) perform it during the lexical chain generation process. Nelken & Shieber [64] show that, to produce lexical chains accurately, WSD is crucial. Performing word sense disambiguation during the lexical chain generation process consistently underperforms preprocessing by a substantial amount [62,65,66]. Thus, performing word sense disambiguation as a preprocessing step should produce superior results.

Current research on lexical chaining algorithms does not provide a satisfactory solution for short social media texts. Prior work on lexical chaining for text clustering uses documents that range from 140 words [67] to 1000 words [49]. In contrast, social media posts only average 11 words per post. None of the current lexical generation algorithms can be directly adapted in a simple way to deal with such short text. Existing lexical chaining algorithms generate lexical chains within each document [48,49,61–63,65,68]. In our research, each document is a short post, so directly applying the existing algorithms would most likely produce very short chains with low quality.

2.3. *Similarity measure for lexical chaining*

One way of deriving similarity among words is by using WordNet [9], which organizes the senses of words hierarchically. For example, the words "fruit," "vegetables," "baked," "dairy" and "dish" are subordinate to the word "food" in WordNet. "Bread" is subordinate to "baked," and "cheese" to "dairy." In WordNet, a synset is a set of synonyms that share a common meaning. Each synset contains one or more lemmas, which represent a specific sense of a specific word. For example, the relevant synset of "vegetable" contains the words {vegetable, veggie, veg} and is defined as "edible seeds or roots or stems or leaves or bulbs or tubers or nonsweet fruits of any of numerous herbaceous plants." In addition to the actual words and their organization into hierarchies and synsets, WordNet provides facilities to calculate the similarity between synsets [69].

Substantial research has developed various algorithms to enhance WordNet similarity. For example, Liu et al. [45] develop an algorithm to identify similar word pairs using WordNet. They test their algorithm on a set of newsgroup posts, using the newsgroup the post originated from as a proxy for similarity of topic. Similarly, Oliva et al. [46] develop an algorithm to match word pairs using WordNet and the syntactical structure of sentences. However, none of these are appropriate for mining short text.

3. Methodology

This research develops a methodology to extract meaning from short text, as found in social media, using semantic clustering. The meaning is embedded in a set of words with similar meanings (lexical chains) that are used to group social media posts (clustering).

The general approach is to first disambiguate the words found in a set of short texts into senses (word sense disambiguation). The texts could be, for example, from multiple Facebook posts by one person. Then, the word senses, which are identified as potentially being similar, are grouped into lexical chains (lexical chain generation). Because some of these lexical chains can be excessively long, extraneous word senses are removed, based on a threshold (lexical chain pruning). Finally, posts having word senses that belong to the same lexical chain are grouped together (semantic clustering). The result is three main steps in the methdology: 1) word sense disambiguation; 2) lexical chain generation and pruning; and 3) semantic clustering. These steps are described below and further illustrated in the next section that evaluates the methodology. This combination of steps differs from prior research and the results from the algorithms developed as part of the methodology, provide better results than other algorithms, as shown below.

3.1. *Step 1: Word sense disambiguation*

Word sense disambiguation has been extensively studied with well-accepted approaches to carrying it out. We, therefore, adopted an existing, frequently-used disambiguation algorithm, "cosine Lesk" [70–73]. The algorithm measures the similarity of adjacent words to identify the word sense of a particular word, based on the way in which definitions are represented in WordNet [9]. For example, in the phrase "pine cone," the algorithm determines that pine is an adjective (strictly a noun adjunct) and cone is a noun, because one synset of pine is "a kind of evergreen tree with needle-shaped leaves," and one definition of cone is "fruit of certain evergreen trees." The overlap between the textual definitions indicates that the words are related.

Only noun senses identified by the algorithm are used to compute the lexical chains. Doing so increases the overall quality of the lexical chains generated, while dramatically reducing the number of features needed for clustering and, hence, reduces the computation time and processing complexity [69]. At the end of this step, every noun in the corpus is assigned to a word sense. For example, the word “cone” in our example, would be assigned the number 3, because it is mapped to the third definition of cone in WordNet (the one associated with pine cones), and not other definitions, such as “a shape whose base is a circle and whose sides taper up to a point,” or “a visual receptor cell in the retina that is sensitive to bright light and to color.” Note that our research is not intended to extend word sense disambiguation, but rather adapt it in our efforts to address short sentence challenges.

3.2. Step 2: lexical chain generation and pruning

After identifying the word senses for the nouns, the next step is to generate a lexical chain and then prune it to a manageable size. The purpose of the generation of a lexical chain is to identify semantic relationships among the short texts that do not share common keywords.

Our algorithm for lexical chain generation proceeds as follows. After the word sense disambiguation in Step 1, a list of word senses has been extracted from the corpus (e.g., a collection of Facebook posts). Two word senses are considered related if their Wu-Palmer similarity [74] score is larger than a threshold ϵ. In the implementation, $\epsilon = 0.7$ is used. This number was derived from repeated test trials (over 50) that applied in the implemented system to sets of Facebook posts.

A lexical chain is denoted by $C = \{{w_1}^{1_i}, \cdots, {w_k}^{k_i}\}$, which represents a set of word senses (i.e., words with one associated WordNet meaning). Throughout this paper, the variable for a lexical chain always uses a capital letter. Here, a lexical chain is formed as follows. We begin with the set W of all word senses found by the WSD algorithm. The set of lexical chains is Φ. Initialize $\Phi = \varnothing$; and make a comparison between any two word-senses (denoted as ${w_q}^{a_i}$and ${w_b}^{b_i}$) in W. If $\delta_{{Wu}_{Palmer}}({w_a}^{a_i}, {w_b}^{b_i}) \geq \epsilon$, create a lexical chain, C, comprising ${w_a}^{a_i}$and ${w_b}^{b_i}$. Then determine how to update Φ with the new lexical chain.

There are four resulting, possible situations.

1. If neither word-sense ${w_a}^{a_i}$ or ${w_b}^{b_i}$exists in the set of lexical chains, then $\Phi' = \Phi \cup C$.That is, add the lexical chain C to the existing set of lexical chains with the two word-senses, and add the chain to the set of lexical chains.
2. If one of the senses is found in an existing chain, but the other sense is not found in any of the chains, then add the sense not found into that chain. That is:

If ${w_a}^{a_i} \in C_i \subseteq \Phi$, but ${w_b}^{b_i} \notin \Phi$, then update the chain C_i by $C_i' = C_i \cup \{{w_b}^{b_i}\}$.

3. If both word senses are found in the set of lexical chains, and the two senses are found in different existing chains, then merge these two chains. That is, remove these two chains and add one that is the union of the two:

If ${w_a}^{a_i}$, ${w_b}^{b_i} \in \Phi$, and $\exists {w_a}^{a_i} \in C_1$ and ${w_b}^{b_i} \in C_2$ while $C_1 \neq C_2$, then update Φ by $(\Phi \cup \{C_3\}) \backslash \{C_1, C_2\}$, where $C_3 = C_1 \cup C_2$.

4. If both senses are found in the set of lexical chains, and the two senses are found in the same existing chain, then no update is needed. That is:

If ${w_a}^{a_i}$, ${w_b}^{b_i} \in \Phi$ and $\exists C_i$ s.t. ${w_a}^{a_i} \in C_i$ and ${w_b}^{b_i} \in C_i$, then no update is needed.

These four situations are illustrated in Table 3 for an educational application.

At this point, the methodology has produced a set of lexical chains, consisting of word senses. The goal of lexical chain pruning is to remove the senses that are less related than most of the other ones in the same chain. The main focus is to select one central word sense, which best explains the semantic connection among most of the senses in the chain. We then eliminate all word senses from the chain whose similarity to that central word sense is lower than a threshold value. This is a similar practice to many other implemented word sense disambiguation algorithms [19,60]. There are two steps.

Table 3
Examples of lexical chain generation.

Situation	Φ At Beginning	C	Φ At End
1	{school, education}	{learning, study}	{school, education} {learning, study}
2	{school, education} {learning, study}	{education, schooling}	{school, education, schooling} {learning, study}
3	{school, education} {learning, study}	{education, learning}	{school, education, learning, study}
4	{school, education, learning, study}	{school, study}	{school, education, learning, study}

1. **Find the central word sense.** For every word sense in the lexical chain, sum its Wu-Palmer similarity to every other word sense in the lexical chain. The word sense with the highest score is the central word sense. If there is a tie, arbitrarily select the word sense that is first in the chain. Note that summing produces an identical result as averaging because, within a lexical chain, the denominator in the averaging function remains the same (the number of words in the chain).
2. **Prune word senses.** All word senses that have a Wu-Palmer similarity score to that central word sense less than a threshold are pruned. We used 0.7 as the threshold, based on our analysis of tests we conducted with other Facebook posts.

3.3. Step 3: semantic clustering

The output from the previous steps is a set of lexical chains, each with its own length, defined as the number of word senses in the chain. Chains of length 2 are removed because of the inherent problems associated with being able to infer semantics from a small chain. For example, a post of “left early,” mostly likely refers to leaving an event of some type before it was finished. It could have other interpretations, such as “took a left turn early,” “left an exam early,” etc. Of course, a human might be able to infer that the word “early” has some association with leaving before one would be expected to, or else would be considered early by some socially accepted norm.

Each remaining lexical chain produces one cluster. A post belongs to the cluster that corresponds to a lexical chain if it contains a word that has a word sense in the chain. Recall the businesses and tourism example. The words “tourism” and “business” would be merged in a lexical chain, because tourism is a form of business and thus the two are closely connected within the WordNet hypernym structure. The two posts would then be associated as part of that lexical chain.

4. Evaluation

To evaluate the methodology, we conducted two empirical assessments. The first used the Facebook posts of a well-known New Zealand political figure. The second used the CMU newsgroup dataset used by Liu et al. [45]. For both assessments, the methodology was compared to well-established clustering algorithms.

4.1. Assessment with Facebook posts

Traditional research in word sense disambiguation employs standard, well-accepted data sets with known properties. These data sets are not only standard, but are the basis for particular metrics used in word sense disambiguation research such as the Silhouette Coefficient [75] and the Calinski-Harabaz index [76]. Known measures, such as the F-measure [77], adjusted Rand index [78], Normalized Mutual Information (NMI) [79] and/or Adjusted Mutual Information (AMI) [80] assume clusters are partitions of the data set. In other words, in terms of this research, a short text that appears in one cluster will not appear in another.

However, the objective of the lexical chain generation algorithm developed for our methodology is to cluster short texts from a single account. There does not appear to be any well-known, accepted short text data sets. Furthermore, it is reasonable for a post in social media to appear in multiple clusters. Consider the following post from John Key: "I've spent the day in Houston, Texas talking about trade, tourism, investment, and business links between our two countries." This post could legitimately be simultaneously classified as being about the economy, business, or diplomacy. Our first assessment thus focused on evaluating the algorithm within the context for which it was designed.

To evaluate the performance of the lexical chain generation algorithms, we collected Facebook posts of John Key, the former prime minister of New Zealand. This data set was selected largely for its applicability to the diplomat problem domain. Therefore, the successful text mining of this data set has practical value. The John Key data set consists of all John Key's Facebook posts from Feb 3rd, 2016 to Dec 3rd, 2016. There are, in total, 499 posts, each post containing, on average, 25 words (including stopwords). Excluding stopwords, the average number of words in each post is 11.

For our lexical chain generation algorithm, we selected only the chains of length greater than 2, because it is difficult to determine what clusters of length 2 mean. In each cluster, there are, on average, 20.90 posts. Since there is no pre-existing standard to access the overall quality of the lexical chain generation algorithm proposed in this research, the posts in each cluster were categorized as belonging to one of three levels based on the quality of the cluster. Based on this evaluation, a "good" cluster is one where the posts have a common theme of ideas and the words in the posts are semantically similar. We label these as Level 1 clusters (the highest level of quality). For example, consider the posts in Table 4 below.

All of these posts demonstrate a pattern of similar thought; namely, that of government engaging with children and/or families.

Next, we categorize the clusters that have posts with two themes, as Level 2 (clusters of average quality). An example of a level 2 cluster is in Table 5. The posts associated with the lexical chain are considered a mixed cluster with several posts related to overseas visits and others related to visits to New Zealand communities.

Finally, level 3 clusters were those where there were 3 or more themes. The typical level 3 cluster would appear (to a human) as a collection of unrelated posts. Table 6 provides an example of a level 3 cluster.

In effect, a level 1 cluster is a "good" one, a level 3 is "poor," and a level 2 cluster has "mixed" results. To assess whether clusters produced were level 1, 2, or 3, two the authors independently reviewed the clusters that were generated by the algorithm and evaluated each cluster for quality and how well the posts mapped to the clusters. Inter-rater reliability was satisfactory, with a kappa of 0.78. Discrepant ratings among the raters were reconciled through discussion.

The John Key data set produced 23 level-1 clusters (51.1%), 2 level-2 clusters (4.4%), and 19 level-3 clusters (44.4%). The results of the application of the methodology show that it is feasible to use the methodology to mine short text, at least for Facebook applications. As an illustrative example, a diplomat who might be scheduled to meet with John Key would be able to mine the posts of John Key and obtain a good sense of the most salient topics on John Key's mind. Table 7 in Section 4.2 summarizes our results.

4.2. *Assessment comparing results of WSD-LC with NMF & GDSMM*

After executing our algorithm, we compared our clustering approach against two other clustering techniques to assess our algorithm's relative accuracy. These two techniques are non-negative matrix factorization (NMF) and a collapsed Gibbs Sampling algorithm for the Dirichlet Multinomial Mixture model (GDSMM), which are the most commonly used clustering algorithms.

The non-negative matrix factorization (NMF) was selected because it is a popular benchmark [81]. The procedure for applying NMF in this research is as follows:

1. Remove all stopwords in each post and stem the words with the Porter Stemmer [82].
2. Generate the term-document matrix for the corpus.
3. Run the NMF algorithm and obtain the clustering results.

An undefined parameter in the NMF algorithm is the number of clusters, which affects its performance. If each cluster has only one post, then all the clusters are cohesive by themselves. On the other hand, the more posts each cluster contains, the less accurate each cluster is likely to be. To achieve a fair comparison with our lexical chaining algorithm, we assess both algorithms with the same number of posts for each cluster on average.

Our lexical chaining algorithm produced 43 clusters, with each cluster having 20.90 posts. NMF partitions all posts. Each post appears in one and only one cluster. Thus, if we allow 20.90 posts per cluster, this will result in 499/20.90 = 23.87clusters. Therefore, we chose 24 clusters for the NMF algorithm. We coded each cluster generated by the NMF algorithm in the same way as our own algorithm; that is, by dividing the 24 clusters generated into three groups based on the cohesiveness of the theme. NMF generated 11 level-1 clusters (45.8%), 4 level-2 clusters (16.7%), and 9 level-3 clusters (37.5%). In total, there are 185 posts clustered in level-1 clusters.

Table 4
Level 1 cluster example.

Date	Facebook posts of child ↔ grandson ↔ son (only 10 posts excerpted)
2016-11-30	An absolute pleasure to meet Sharon van der Gulik today, along with her husband Harry and her grandson Matt, who donated one of his kidneys to Sharon. Live organ donors really are inspirational.
2016-11-23	Visited Waiau today to catch-up with locals after the earthquake and popped into the school to deliver some toys made by Hanmer children.
2016-09-01	Visited Mind Lab, an after school programme doing a great job teaching Gisborne kids about technology.
2016-08-24	Getting asked the tough questions by the kids at Alexandra Kindergarten - like how many bees live in the Beehive.
2016-08-17	The new Ministry for Vulnerable Children, Oranga Tamariki announced today demonstrates the Government's commitment to properly caring for and protecting our children and young people.
2016-06-29	Today Health Minister Jonathan Coleman and I launched the Raising Healthy Kids target, an important step in helping us tackle childhood obesity.
2016-06-16	Visited Rainbow's End today where they are developing a new attraction called Driver's Town - which allows kids to drive round in miniature cars.
2016-05-10	We've just announced that we're providing more teacher aide support for 1250 kids who have additional learning needs.
2016-05-05	Spoke and met with senior students from both Marlborough Girls' and Boys' Colleges.
2016-03-30	On 1 April Kiwi families will be better off with 18 weeks paid parental leave, an increase in the minimum wage to $15.25, $25 more a week and increased work obligations for beneficiaries with children, and around $12.50 more a week through Working for Families.

Table 5
Level 2 cluster example.

Date	Facebook post of Jakarta ↔ city ↔ town ↔ Hamilton ↔ Sydney ↔ Beijing ↔ Nice ↔ Santos ↔ Colombo ↔ Lincoln ↔ Lyon ↔ Lima (only first 10 posts replicated)
2016-12-02	Caught up with Dr. Matt Glenn, Dr. Roger Hill and the team from Hill Laboratories in Hamilton today - a New Zealand company providing lab tests to domestic and international markets
2016-11-18	Arrived in Lima, Peru for this year's APEC Leaders' meeting which is focused on improving growth and living standards
2016-11-17	Later today, I'm heading off for a short visit to Peru to attend the APEC Leaders' Meeting. It'll be a great opportunity to promote New Zealand as a good place to do business and invest. I return on Tuesday morning and will be keeping a close eye on the earthquake response while I'm away.
2016-11-02	Nice way to start the day in Hamilton popped into Tamahere Model Country School to meet students and teachers, and got to try some of the recipes from their new cook book.
2016-10-27	Caught up with brand ambassador and Bollywood star Sidharth Malhotra at a Tourism NZ event in New Delhi. Looking forward to having him back in NZ.
2016-10-25	Pleasure to be officially welcomed in New Delhi by India Prime Minister Narendra Modi.
2016-10-25	Arrived in New Delhi and am looking forward to meeting with India's Prime Minister Narendra Modi later today.
2016-10-04	Great to visit the Toyota Signature Class plant in Thames where they refurbish 3400 cars a year and employ 70 locals.
2016-10-04	Talked transport, trade, immigration, and the economy in a town hall meeting in Kaiaua with locals and MP Scott Simpson this morning.
2016-09-21	Preparing to chair the United Nations Security Council with UK Foreign Secretary Boris Johnson, Colombian President Juan Manuel Santos, Chilean President Michelle Bachelet, and UN Secretary-General Ban Ki-moon.

We also compared our algorithm against GSDMM [12], which is an unsupervised clustering algorithm especialy designed for short texts. To do so, we used the GSDMMin GitHub implementation [83]. The GSDMM algorithm partitions all posts into clusters. It then iteratively shifts posts between clusters based on:

- Factor 1: The size of the cluster: Documents are shifted from smaller clusters to bigger ones.
- Factor 2: The amount of overlap between the words in a post, and the words in a cluster. The greater the degree of overlap, the more likely the post is to be shifted to that cluster.
- Factor 3: The number of iterations the algorithm will use. Purportedly, the more iterations, the more accurate the clustering will be.

GSDMM is an algorithm that puts posts together based on shared common words and the size of clusters. It does not use any semantic information from the post at all. Threshold values are specified to influence the extent that factors 1 and 2 drive the GSDMM algorithm. The same parameters as those used by Yin & Wang [12] (the algorithm's developers) were employed; that is, 0.1 for factor 1 and 2 and 30 for factor 3. We also specified that the number of clusters would be 24 following the same logic as above. However, the GSDMM algorithm itself determined that the optimal number of clusters was 16.

The results of empirically evaluating the clusters produced by NMF and GSDMM are summarized in Table 7. GSDMM produces 8 level-1 clusters (50.0%), 5 level-2 clusters (31.3%) and 3 level-3 clusters (18.8%). The total number of posts in level-1 clusters is 182.

Among the three clustering algorithms, lexical chaining produced the highest percentage of level-1 clusters (51.1 vs 45.8 vs 50%). Furthermore, lexical chaining produced more relevant clusters (23 vs 11 vs 8) and the clusters produced by lexical chaining were larger than those of the other algorithms.

Table 7
Comparison between lexical chaining, NMF and GSDMM clustering results on 20 Newsgroups Dataset.

	Level-1 clusters		Level-2 clusters		Level-3 clusters	
Lexical chaining	23 clusters	51.1%	2 clusters	4.4%	19 clusters	44.4%
NMF	11 clusters	45.8%	4 clusters	16.7%	9 clusters	37.5%
GSDMM	8 clusters	50.0%	5 clusters	31.3%	3 clusters	18.8%

Thus, our lexical chaining appears to have outperformed the other algorithms. This result can be attributed to the algorithm using semantic information from WordNet, whereas the other clustering algorithms rely principally on co-occurring words. As a result, the application of the lexical chains should be useful for short text structures, such as those found on Facebook. Then, better and more automated results can be obtained, but it also means that the users of social media need to be aware of the inferences that can be made from their posts. Finally, even though the lexical chains appear to be effective in making some attempt to capture semantics, the strengths of the lexical chains must be considered.

4.3. Assessment with newsgroup dataset

For the second assessment, a more traditional data set was used; the baseball and politics subdatasets of the 20 Newsgroups dataset [10]. We chose these subdatasets, because the language employed was associated with common English. The other subdata sets such as the comp.OS data set employ terms one would not expect to be found in WordNet.

Although the 20 Newsgroups dataset is perhaps the closest public available one to the problem we are trying to address, and has been used by other researchers, it is by no means close. Text in this data set tend to be long paragraphs, rather than short social media messages.

Table 6
Level 3 cluster example.

Date	Facebook Posts of day↔ night ↔ weekend ↔ evening ↔ calendar ↔ season ↔ decade ↔ century ↔ flower (only first 10 posts replicated)
2016-12-01	Had a lovely evening out with my beautiful wife Bronagh last night to celebrate our 32nd wedding anniversary.
2016-11-28	Great to attend the Indian Newslink Business Awards in Auckland last night.
2016-11-26	Was great to get down to the Canterbury A&P Show earlier this month and catch up with those involved in our vital primary sector industries.
2016-11-24	Great to recognise the global success of Kiwi businesses at last night's New Zealand International Business Awards in Auckland.
2016-11-20	If you missed my open letter on the weekend about how proud I am of the way the local communities and support services have responded to last week's earthquake - you can read it here: http://www.nzherald.co.nz/nz/news/article.cfm?c_id = 1&objectid = 11750075
2016-11-16	New Zealand will stand alongside earthquake-affected communities in the coming days, weeks and months.
2016-11-13	Prime Minister John Key provides an update following last night's 7.5 earthquake near Culverden
2016-11-02	Nice way to start the day in Hamilton 鈥? Popped into Tamahere Model Country School to meet students and teachers, and got to try some of the recipes from their new cook book.
2016-11-01	In the past three months more than 35,000 people have found jobs.
2016-10-20	Tough assignment to start the day in New Plymouth with local MP Jonathan Young story time at Plunket.

Table 8
Comparison between lexical chaining, NMF and GSDMM clustering results on John Key Data Set.

	Level-1 clusters		Level-2 clusters		Level-3 clusters	
Lexical Chaining	12 clusters	52.1%	2 clusters	8.7%	9 clusters	39.1%
NMF	16 clusters	33.3%	5 clusters	10.4%	27 clusters	56.3%
GSDMM	0 clusters	0.0%	0 clusters	0.0%	5 clusters	100%

Interrater reliability for this data set was low- Kappa = 0.54. This was because the longer nature of the posts meant each post often covered multiple topics making the posts difficult to cluster. Our results after reconciliation are presented in Table 8.

The results showed an abysmal performance by GSDMM. GSDMM essentially concluded that most posts in the data set should form one large (unintelligible) cluster. For the baseball data set, GSDMM formed 3 clusters: one cluster having 3 posts, one having 9 (both unintelligible) and the last cluster being every other post. For the politics data set, it formed 2 clusters, one with one post, and the other containing every other post in the data set. NMF formed many small clusters, most of which did not make a lot of sense.

In contrast, our algorithm generated 16 clusters for baseball, and 7 clusters on politics. Some of these clusters had clear associations. For example, one baseball cluster had conversations about player salaries. Another was about teams associated with birds (e.g., Cardinals, Blue Jays). Similarly, one politics cluster was about global warming (but included posts associated with burning during the Waco siege). Another cluster concerned “the police state.” Of course, our algorithm also produced nonsense clusters. One such cluster found in both data sets associated words such as “duke,” and “knight.” The algorithm associated email addresses (e.g., duke.edu) and other unrelated words to create that cluster.

5. Conclusion

This research has proposed a methodology, based on lexical chains, to cluster various types of short text structures found in social media texts. The methodology includes a lexical chain generation algorithm, which was empirically evaluated by applying it to a Facebook corpus from a diplomat who served as the leader of a country, and to a subsegment of the 20 newsgroups corpus. The results suggest that our lexical chain generation algorithm outperforms the most commonly applied clustering algorithm, for short text sentences. This research, then, attempts to capture and represent semantics when there is a lack of commonality of terms, through the generation of the lexical chains and the development of a text clustering system to infer the overall meaning of the generated lexical chains.

Future research will include additional empirical evaluation on large volumes of social media data, expanding the development and evaluation of the lexical chains. This may include graph-based or word-embedded word sense disambiguation (e.g., [84]). Applications to large-scale business problems are needed to further demonstrate the applicability and usefulness of the research.

It should also be possible to combine the methodology presented in this research with others to better enhance short-text clustering. For example, machine-learning methods for text-clustering, such as word2vec [28,31], require an extensive training set to learn how to perform effective clustering. But in the politician/social media and marketing/social media domains, the effort required to create such a training set is often not justified. It is better to manually cluster the posts involved in the analysis than to build the training set. Our method reduces the cost of clustering, thus providing quick insights that should be useful to analysts, as well as a way to more cheaply create training sets for relevant machine-learning algorithms.

Cecil Eng Huang Chua is an associate professor at the University of Auckland, Missouri University of Science and Technology. He received a PhD in Information Systems from Georgia State University, a Masters of Business by Research from Nanyang Technological University and both a Bachelor of Business Administration in Computer Information Systems and Economics and a Masters Certificate in Telecommunications Management from the University of Miami. Cecil has several publications in such journals as *Information Systems Research, Journal of the AIS, MIS Quarterly* and the *VLDB Journal.* Cecil has consulted for a range of organizations including Daimler SEAsia, General Motors Singapore, the Singapore Ministry of Defense, and Fonterra.

Veda C. Storey is the Tull Professor of Computer Information Systems and professor of computer science at the J. Mack Robinson College of Business, Georgia State University. Her research interests are in intelligent information systems, data management, conceptual modeling, and design science research. Dr. Storey is a member of the AIS College of Senior Scholars, an AIS Fellow, and an advisor to the Workshop on Information Technologies and Systems. She is also a member of the steering committee of the International Conference of Conceptual Modeling.

Xiaolin Li received his Ph.D. from Hong Kong University of Science and Technology. He is now working as an artificial intelligence engineer for Xtracta. His research interests include text mining, machine learning, statistical modeling, information theory and combinatorics.

Mala Kaul is an assistant professor of information systems in the College of Business at the University of Nevada, Reno. She received her Ph.D. from the Robinson College of Business at Georgia State University. Her research focuses on information systems design, cyber security and privacy, and health information technology. She has extensive industry experience as an Information Systems professional. Her work has been published in MIS Quarterly, Journal of Management Information Systems, European Journal of Information Systems, Harvard Business Review, and other journals.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.dss.2019.113142.

References

[1] N. Li, D.D. Wu, Using text mining and sentiment analysis for online forums hotspot detection and forecast, Decision Support Systems 48 (2) (2010) 354–368.

[2] V.B. Kobayashi, S.T. Mol, H.A. Berkers, G. Kismihók, D.N.D. Hartog, Text Mining in Organizational Research, Organizational Research Methods 21 (3) (2018) 733–765.

[3] J. DeMers, 5 skills your social media manager must have, Retrieved from, 2015. https://www.forbes.com/sites/jaysondemers/2015/06/08/5-skills-your-social-media-manager-must-have/#5ed34c17ac56.

[4] D. Cohen, STUDY: customer service via Facebook hampered by companies's choice of platforms, Retrieved from, 2014. http://www.adweek.com/digital/study-customer-service-forrester-research-conversocial/.

[5] J. Girard, D. Klein, K. Berg, Strategic Data-Based Wisdom in the Big Data Era: Information Science Reference, IGI Publishing Hershey, PA, USA, 2017.

[6] H.-S. Lee, H.-R. Lee, J.-U. Park, Y.-S. Han, An abusive text detection system based on enhanced abusive and nonabusive word lists, Decision Support Systems 113 (1) (2018) 22–31.

[7] S. Stieglitz, L. Dang-Xuan, Emotions and information diffusion in social media—-sentiment of microblogs and sharing behavior, Journal of Management Information Systems 29 (4) (2013) 217–248.

[8] R. Steele, Open source intelligence: What is it? Why is it important to the military? Paper presented at the Open Source Intelligence: READER Proceedings, 6th International Conference & Exhibit Global Security & Global Comp, 1997.

[9] C. Fellbaum, WordNet - an electronic lexical database, MIT Press, Cambridge, MA, 1998.

[10] K. Lang, 20 newsgroups data set, http://qwone.com/~jason/20Newsgroups/, (2007).

[11] W. Xu, X. Liu, Y. Gong, *Document clustering based on non-negative matrix factorization.* Paper presented at the Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Toronto, Canada, (2003).

[12] J. Yin, J. Wang, A Dirichlet multinomial mixture model-based approach for short text clustering, Paper presented at the Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York,

New York, USA, 2014.

[13] T.M. Shaft, I. Vessey, Research report—the relevance of application domain knowledge: the case of computer program comprehension, Information Systems Research 6 (3) (1995) 286–299.

[14] T.R. Gruber, A translation approach to portable ontology specifications, Knowledge Acquisition 5 (2) (1993) 199–220.

[15] R. Alt, M. Wittwer, Towards an ontology-based approach for social media analysis, Paper presented at the The 22nd European Conference on Information Systems, Tel Aviv, Israel, 2014.

[16] M.A.K. Halliday, R. Hasan, Cohesion in english, Longman, London, 1976.

[17] A.M. Kaplan, M. Haenlein, Users of the world, unite! The challenges and opportunities of social media, Business Horizons 53 (1) (2010) 59–68.

[18] J. Leskovec, A. Rajaraman, J. Ullman, Mining of massive datasets, (2011).

[19] R. Navigli, M. Lapata, An experimental study of graph connectivity for unsupervised word sense disambiguation, IEEE Trans. Pattern Anal. Mach. Intell. 32 (4) (2010) 678–692, https://doi.org/10.1109/tpami.2009.36.

[20] Newswhip, What's the perfect length of a Facebook post? Retrieved from, 2016. https://www.newswhip.com/2016/07/perfect-length-of-a-facebook-post/.

[21] Wang, X., Zhao, K., Cha, S., Amato, M. S., Cohn, A. M., Pearson, J. L., … Graham, A. L. (2019). Mining user-generated content in an online smoking cessation community to identify smoking status: a machine learning approach. Decision Support Systems, 116(1), 26–34.

[22] X.-H. Phan, L.-M. Nguyen, S. Horiguchi, *Learning to classify short and sparse text & web with hidden topics from large-scale data collections*. Paper presented at the Proceedings of the 17th International Conference on World Wide Web, Beijing, China, (2008).

[23] T. Narock, L. Zhou, V. Yoon, Semantic similarity of ontology instances using polarity mining, Journal of the American Society for Information Science and Technology 64 (2) (2013) 416–427.

[24] C.C. Aggarwal, C.X. Zhai, Mining text data, Springer Publishing Company, Incorporated, 2012.

[25] D.E.P. Avendano, *Analysis of narrow-domain short texts clustering.* (diploma of advanced studies), Retrieved from Polytechnic University of Valencia, http://users.dsic.upv.es/~prosso/resources/PintoDEA.pdf, (2007).

[26] L. Jing, M.K. Ng, X. Yang, J.Z. Huang, A text clustering system based on k-means type subspace clustering and ontology, International Journal of Computer, Electrical, Automation, Control and Information Engineering 2 (4) (2008) (1296-1130).

[27] D.M. Blei, A.Y. Ng, M.I. Jordan, Latent Dirichlet allocation, Journal of Machine Learning Research 3 (2003) 993–1022.

[28] Y. Goldberg, O. Levy, word2vec explained: Deriving Mikolov et al.'s negative-sampling word-embedding method, (2014) (Retrieved from arXiv).

[29] T. Hofmann, Probabilistic latent semantic indexing, Paper presented at the Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Berkeley, California, USA, 1999.

[30] T. Mikolov, K. Chen, G. Corrado, J. Dean, *Efficient estimation of word representations in Vector space.* Paper presented at the in Proceedings of Workshop at ICLR, (2013) 2013/01/16.

[31] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, Paper presented at the Proceedings of the 26th International Conference on Neural Information Processing Systems, Lake Tahoe, Nevada, 2013.

[32] D. Andrzejewski, X. Zhu, M. Craven, Incorporating domain knowledge into topic modeling via Dirichlet Forest priors, Paper presented at the Proceedings of the 26th Annual International Conference on Machine Learning, Montreal, Quebec, Canada, 2009.

[33] D. Newman, E.V. Bonilla, W. Buntine, Improving topic coherence with regularized topic models, Paper presented at the Proceedings of the 24th International Conference on Neural Information Processing Systems, Granada, Spain, 2011.

[34] J. Petterson, W. Buntine, S.M. Narayanamurthy, T.S. Caetano, A.J. Smola, Word features for latent Dirichlet allocation, Paper presented at the Advances in Neural Information Processing Systems, 2010.

[35] S. Banerjee, K. Ramanathan, A. Gupta, Clustering short texts using Wikipedia, Paper presented at the Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Amsterdam, The Netherlands, 2007.

[36] I.S. Dhillon, D.S. Modha, Concept decompositions for large sparse text data using clustering, Machine Learning 42 (1) (2001) 143–175, https://doi.org/10.1023/A:1007612920971.

[37] J. Hu, L. Fang, Y. Cao, H.-J. Zeng, H. Li, Q. Yang, Z. Chen, Enhancing text clustering by leveraging Wikipedia semantics, Paper presented at the Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Singapore, Singapore, 2008.

[38] L. Jing, M.K. Ng, J. Xu, J.Z. Huang, Subspace clustering of text documents with feature weighting K-means algorithm, Paper presented at the Proceedings of the 9th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining, Hanoi, Vietnam, 2005.

[39] L.M. Aiello, G. Petkos, C. Martin, D. Corney, S. Papadopoulos, R. Skraba, … A. Jaimes, Sensing trending topics in twitter, IEEE Transactions on Multimedia 15 (6) (2013) 1268–1282, https://doi.org/10.1109/tmm.2013.2265080.

[40] M. Mathioudakis, N. Koudas, TwitterMonitor: Trend detection over the twitter stream, Paper presented at the Proceedings of the 2010 ACM SIGMOD International Conference on Management of data, Indianapolis, Indiana, USA, 2010.

[41] K. Dave, S. Lawrence, D.M. Pennock, Mining the Peanut gallery: opinion extraction and semantic classification of product reviews, Paper presented at the Proceedings of the 12th International Conference on World Wide Web, Budapest, Hungary, 2003.

[42] A. Hotho, S. Staab, G. Stumme, Wordnet improves text document clustering, Paper presented at the 26th Annual International ACM SIGIR Conference, Toronto, Canada, 2003.

[43] X. Hu, N. Sun, C. Zhang, T.-S. Chua, Exploiting internal and external semantics for the clustering of short texts using world knowledge, Paper Presented at the Proceedings of the 18th ACM Conference on Information and Knowledge Management, Hong Kong, China, 2009.

[44] X. Hu, X. Zhang, C. Lu, E.K. Park, X. Zhou, Exploiting Wikipedia as external knowledge for document clustering, Paper Presented at the Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France, 2009.

[45] W. Liu, X. Quan, M. Feng, B. Qiu, A short text modeling method combining semantic and statistical information, Information Sciences 180 (20) (2010) 4031–4041.

[46] J. Oliva, J.I. Serrano, M.D.D. Castillo, A. Iglesias, SyMSS: a syntax-based measure for short-text semantic similarity, Data & Knowledge Engineering 70 (4) (2011) 390–405.

[47] J. Sedding, D. Kazakov, *WordNet-based text document clustering.* Paper presented at the Proceedings of the 3rd Workshop on Robust Methods in Analysis of Natural Language Data, Geneva, (2004).

[48] D. Jayarajan, D. Deodhare, B. Ravindran, S. Sarkar, Document clustering using lexical chains, Paper presented at the in Proceedings of the Workshop on Text Mining and Link Analysis (TextLink 2007), Hyderbad, India, 2007.

[49] T. Wei, Y. Lu, H. Chang, Q. Zhou, X. Bao, A semantic approach for text clustering using WordNet and lexical chains, Expert Systems and Applications 42 (4) (2015) 2264–2275, https://doi.org/10.1016/j.eswa.2014.10.023.

[50] Y. Wang, W. Xu, Leveraging deep learning with LDA-based text analytics to detect automobile insurance fraud, Decision Support Systems 105 (1) (2018) 87–95.

[51] H. Yuan, R.Y.K. Lau, W. Xu, The determinants of crowdfunding success: a semantic text analytics approach, Decision Support Systems 91 (1) (2016) 67–76.

[52] V. Kumar, R. Sridhar, *Unsupervised topic modeling for short texts using distributed representations of words.* Paper presented at the in Proceedings of NAACL-HLT 2015, Denver, Colorado, (2015) May 31–June 5, 2015.

[53] P. Xie, D. Yang, E.P. Xing, Incorporating word correlation knowledge into topic modeling, Paper presented at the Conference of the North American Chapter of the Association for Computational Linguistics, 2015.

[54] S. Phuvipadawat, T. Murata, Breaking news detection and tracking in twitter, Paper presented at the Proceedings of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, Vol. 03 2010.

[55] K.D. Rosa, R. Shah, B. Lin, A. Gershman, R. Frederking, Topical clustering of tweets, Paper presented at the in Proceedings of SWSM'10, Beijing, China, 2011 July 28, 2011.

[56] G. Petkos, S. Papadopoulos, Y. Kompatsiaris, Two-level message clustering for topic detection in twitter, Paper presented at the in Proceedings of the SNOW 2014 Data Challenge, Seoul, Korea, 2014 April, 2014.

[57] T. Sakaki, M. Okazaki, Y. Matsuo, Earthquake shakes twitter users: Real-time event detection by social sensors, Paper Presented at the Proceedings of the 19th International Conference on World Wide Web, Raleigh, North Carolina, USA, 2010.

[58] Oskin, B. (2014, May 6, 2014). #Earthquake! Tweets beat official quake alerts. Retrieved from http://www.livescience.com/45385-earthquake-alerts-from-twitter.html

[59] E. Gabrilovich, Feature generation for textual information retrieval using world knowledge, SIGIR Forum 41 (2) (2007) 123, https://doi.org/10.1145/1328964.1328988.

[60] R. Sinha, R. Mihalcea, *Unsupervised graph-based word sense disambiguation using measures of word semantic similarity*. Paper presented at the Proceedings of the International Conference on Semantic Computing, (2007).

[61] G. Ercan, I. Cicekli, Using lexical chains for keyword extraction, Information Processing & Management 43 (6) (2007) 1705–1714, https://doi.org/10.1016/j.ipm.2007.01.015.

[62] H.G. Silber, K.F. McCoy, Efficient text summarization using lexical chains, Paper Presented at the Proceedings of the 5th International Conference on Intelligent User Interfaces, New Orleans, Louisiana, USA, 2000.

[63] G. Hirst, D. St Onge, Lexical chains as representation of context for the detection and correction of malapropisms *WordNet: An Electronic Lexical Database (Language, Speech, and Communication)*, The MIT Press, 1998.

[64] R. Nelken, S.M. Shieber, Lexical chaining and word-sense-disambiguation, Retrieved from, 2007. https://dash.harvard.edu/handle/1/9136730.

[65] R. Barzilay, M. Elhadad, Using lexical chains for text summarization, Paper presented at the Proceedings of Intelligent Scalable Text Summarization Workshop Madrid, Spain, 1997.

[66] M. Galley, K. McKeown, Improving word sense disambiguation in lexical chaining, Paper presented at the Proceedings of the 18th International Joint Conference on Artificial Intelligence, Acapulco, Mexico, 2003.

[67] Pinto, D., Rosso, P., Jim, C., & Nez-Salazar. (2010). On the assessment of text corpora. Paper presented at the Proceedings of the 14th international conference on Applications of Natural Language to Information Systems, Saarbr&#252; Germany.

[68] D. Jayarajan, D. Deodhare, B. Ravindran, Lexical chains as document features, Paper Presented at the in Proceedings of the Third International Joint Confernce on Natural Language Processing (IJCNLP 2008), Hyderabad, India, 2008 January 2008.

[69] S. Fodeh, B. Punch, P.-N. Tan, On ontology-driven document clustering using core semantic features, Knowledge and Information Systems 28 (2) (2011) 395–421, https://doi.org/10.1007/s10115-010-0370-4.

[70] R. Laatar, C. Aloulou, L.H. Belguith, Word sense disambiguation using skip gram model to create a historical dictionary for Arabic, Paper presented at the 15th

International Conference on Computer Systems and Applications, Aqaba Al 'Aqaba, Jordan, 2018.

[71] S. Patwardhan, S. Banerjee, T. Pedersen, *Using measures of semantic relatedness for word sense disambiguation.* Paper presented at the Proceedings of the 4th International Conference on Computational Linguistics and Intelligent Text Processing, Mexico City, Mexico, (2003).

[72] M. Purvali, S. Orlando, Enriching documents by linking salient entities and lexical-semantic expansion, Journal of Intelligent Systems (2019) (Forthcoming).

[73] L. Tan, Pywsd: python implementations of Word Sense Disambiguation (WSD) technologies [software], Retrieved from, 2014. https://github.com/alvations/pywsd.

[74] Z. Wu, M. Palmer, *Verbs semantics and lexical selection.* Paper presented at the Proceedings of the 32nd annual meeting on Association for Computational Linguistics, Las Cruces, New Mexico, (1994).

[75] P. Rousseeuw, Silhouettes: a graphical aid to the interpretation and validation of cluster analysis, Journal of Computational and Applied Mathematics 20 (1) (1987) 53–65, https://doi.org/10.1016/0377-0427(87)90125-7.

[76] T. Caliński, J. Harabasz, A dendrite method for cluster analysis, Communications in Statistics 3 (1) (1974) 1–27, https://doi.org/10.1080/03610927408827101.

[77] C.J.V. Rijsbergen, Information retrieval, Butterworth-Heinemann, 1979.

[78] L. Hubert, P. Arabie, Comparing partitions, Journal of Classification 2 (1) (1985) 193–218, https://doi.org/10.1007/BF01908075.

[79] A. Strehl, J. Ghosh, Cluster ensembles — a knowledge reuse framework for combining multiple partitions, Journal of Machine Learning Research 3 (2003) 583–617, https://doi.org/10.1162/153244303321897735.

[80] N.X. Vinh, J. Epps, J. Bailey, Information theoretic measures for clusterings comparison: variants, properties, normalization and correction for chance, Journal of Machine Learning Research 11 (2010) 2837–2854.

[81] E. Hosseini-Asl, J.M. Zurada, Nonnegative matrix factorization for document clustering: a survey, Cham, (2014).

[82] M.F. Porter, An algorithm for suffix stripping, in: J. Karen Sparck, W. Peter (Eds.), Readings in information retrieval, Morgan Kaufmann Publishers Inc, 1997, pp. 313–316.

[83] R. Walker, GSDMM: short text clustering, Retrieved from, Sep 11, 2017. https://github.com/rwalk/gsdmm.

[84] I. Iacobacci, M.T. Pilehvar, R. Navigli, Embeddings for word sense disambiguation: an evaluation study, Paper Presented at the Annual Meeting of the Association for Computational Linguistics, Berlin, Germany, 2016 http://aclweb.org/anthology/P/P16/P16-1085.pdf.