Online Social Networks and Media 5 (2018) 23–36

Contents lists available at ScienceDirect

Online Social Networks and Media

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

Hybrid semantic clustering of hashtags

Ali Javed, Byung Suk Lee*

Department of Computer Science, University of Vermont, Burlington, Vermont, USA

ARTICLE INFO

Article history:
Received 19 February 2017
Revised 28 October 2017
Accepted 28 October 2017

Keywords:
Hybrid clustering
Semantic clustering
Hashtag
Social media

ABSTRACT

Clustering hashtags based on their semantics is an important problem with many applications. The uncontrolled usage of hashtags in social media, however, makes the quality of semantics and the frequency of usage vary a lot, and this poses a challenge to the current approaches which capitalize on either the lexical semantics of a hashtag (by using metadata) or the contextual semantics of a hashtag (by using the texts associated with a hashtag). This paper presents a *hybrid* semantic clustering algorithm that uses the complementary strengths of lexical and contextual semantics of a hashtag to produce accurate clusters on a wider range of input data. The hybrid algorithm uses a consensus clustering approach, which finds the consensus between metadata-based sense-level semantic clusters and text-based semantic clusters. A gold standard test shows that the hybrid algorithm outperforms both the text-based algorithm and the metadata-based algorithm for a majority of ground truths tested and that it never underperforms both base algorithms. In addition, a larger-scale performance study, conducted with a focus on disagreements in cluster assignments between algorithms, show that the hybrid algorithm makes the correct cluster assignment in a majority of disagreement cases.

1. Introduction

A hashtag is “a word or phrase that starts with the symbol # and that briefly indicates what a message (such as a tweet) is about” [1]. Chris Messina first proposed to use ‘#’ on Twitter in August 2007, to tag topics of interest [2]. Hashtags are now used in social media for all sorts of reasons – to tell jokes, follow topics, launch campaigns, put advertisements, collect consumer feedback, and much more. #OccupyWallStreet, #ShareaCoke and #NationalFriedChickenDay are just a few examples of many successful hashtag campaigns. McDonald’s created hashtag #Mcdstories to collect consumer feedback.

Since Twitter is the first social media platform that introduced hashtags, it is used as the representative social media in this paper. It is estimated that, as of January 2016, Twitter has about 332 million active monthly users uploading 500 million tweets per day. A tweet is a string up to 140 characters, and most tweets contain one or more hashtags in them.

Clustering is a well-known data mining technique for dividing items into groups (or “clusters”) such that items within the same cluster tend to be more similar to each other than those in different clusters [3]. Clustering is commonly used as a text classification technique [4], and, as asserted by Vicient and Moreno [5], clustering of hashtags is the first step in the classification of tweets given that hashtags are used to index those tweets. Therefore, it can be argued that classification of tweets benefits from accurate clustering of hashtags.

Further, on average 6000 messages are posted per second [6] on Twitter alone, making social media arguably the best source of timely information. In this regard, social media analysts use clusters of hashtags as the basis for more complex tasks [7], such as retrieving relevant tweets [7,8], tweet ranking, sentiment analysis [9], data visualization [10], semantic information retrieval [11], and user characterization. Therefore, the accuracy of hashtag clustering is important to the quality of the information resulting from those tasks.

Hashtag clustering has real world impacts. For instance, it can be used to improve the user engagement in social media activities. Social media websites typically use posts (e.g., tweets) on “home timelines” to increase the level of user engagement. Posts may appear on a user’s home timeline for a number of reasons – because they are shared by the user’s direct contacts, because they are publicly disseminated as popular posts, and because they are advertisements sponsored by commercial entities. Given that a hashtag is a viable representation of the posts, accurate clustering of hashtags can improve the content rendering of those timelines for certain users by introducing posts that are beyond their social

* Corresponding author.
E-mail addresses: ajaved@uvm.edu, alijaved@live.com (A. Javed), bslee@uvm.edu (B.S. Lee).

https://doi.org/10.1016/j.osnem.2017.10.004
2468-6964/

network but relevant to their interests as gauged by the hashtags in their posts. In another instance, the categorization of users, resulting from clustering their posts by hashtag, can help advertisement agencies find new potential customers.

There are two major approaches to clustering hashtags. One approach identifies the *lexical* semantics of hashtags from external resources (i.e., "metadata") independent of the tweet messages themselves [5,12,13]. The other approach does that from the tweet texts (i.e., "data") accompanying hashtags [7,10,11,14–17] by identifying their *contextual* semantics [18].

Performance of the metadata-based approach depends on two factors – metadata quality and hashtag quality. It is out of question that the quality of the metadata has a direct impact on the performance. As importantly, with no syntactic or semantic control over the message content, it is common that hashtags contain errors and abbreviations, thus hampering metadata search quality because of poor quality of the search input.

The metadata-based approaches at the present time are a relatively new area of research that is benefiting from the increasing availability of metadata. This approach has the advantage of being immune to poor linguistic quality of tweet messages that contain hashtags, but has the disadvantage of being sensitive to the quality of metadata or the degree of match between them and hashtags.

There have been more works using the text-based approach [7,10,11,14–17]. In this approach, tweet messages are compared using the *bag-of-words* model [19], and thus the performance depends largely on the amount of text associated with the hashtag. This approach has the advantage of being largely unaffected by poor linguistic quality of hashtag and being able to span across all languages (including slang/informal languages).

It, however, has the disadvantage of working well only on *common* hashtags, as uncommon hashtags do not have enough tweet messages accompanying them. As cited by Tsur et al. [15], 1000 most popular hashtags, which comprise 0.003% of all distinct hashtags, cover about 43% of over 417 million tweets in their corpus – this puts the performance of the bag-of-words approach in question for the remaining 99.997% of hashtags.

Thus, the current approaches to semantic hashtag clustering do not possess the *versatility* needed to produce accurate clusters under varying circumstances, that is to say, all common or rare English language hashtags with varying semantic quality. The sources of hashtag semantics used in the current approaches are orthogonal to each other and their performances are complementary to each other. Hence, this paper aims to combine the two approaches into a *hybrid* approach. The aim is that the hybrid algorithm produces accurate results on a wider range of input data. Such a versatile algorithm unburdens the user from having to decide which algorithm to use for accurate results when there is no ground truth available or when the tweet dataset is so arbitrary that it is not clear which approach is better.

Thus, this paper addresses the problem of clustering hashtags based on two kinds semantics – lexical from metadata and contextual from texts. For this purpose, two base algorithms, each specializing in the respective semantic sources, are utilized and the hybrid semantics combining the two sources are realized by building a consensus from the results of the two base algorithms. To the best of our knowledge, this paper is the first one that addresses combing two distinct semantic sources, namely "lexical" and "contextual", to identify the semantics of hashtags for a certain task, e.g., clustering.

Specifically, we design a hybrid semantic clustering algorithm using two base algorithms, each representing one of the two approaches. The first base is the metadata-based semantic hashtag clustering algorithm introduced in our prior work [12,13] enhanced from the original algorithm by Vicient and Moreno [5]. The second base is the text-based semantic hashtag clustering algorithm adapted from the algorithm proposed by Tsur et al. [15,16] and Muntean et al. [7], which uses the bag-of-words model. Output clusters of these two base algorithms are input to the hybrid algorithm. This hybrid algorithm is based on the concept of *consensus clustering*, as a mere intersection of the two outputs would be too restrictive and not scalable (if more base algorithms were to be added later).

Our hybrid clustering is unique in that what it combines are the two distinct, yet complementary sources of semantics (i.e., lexical and contextual) on the same clustering method (e.g., hierarchical clustering), while other existing body of work on hybrid clustering (e.g., [20–23]) combine two distinct clustering methods. Additionally, no existing hashtag clustering algorithm utilizes multiple distinct sources of semantics to produce more accurate results on a wider range of data, thus validating the complementary nature of semantics used.

Our hybrid clustering algorithm was evaluated using two different experiments – a gold standard test and a "pairwise disagreement" test. The gold standard test showed that two, among the three (i.e., hybrid and the two base) algorithms, the hybrid algorithm achieved the highest accuracy for 57% of ground truth data sets and the second highest accuracy for the remainder (i.e., 43%) of them, and in this case the gap with the better one was marginal (i.e., 10% to 17% in "weighted average pairwise maximum f-score"). The pairwise disagreement test was done with a focus on the instances of disagreement occurring in clustering decision between the hybrid and the base algorithms, where a decision was made for each pair of hashtags whether to cluster them together or to separate them. The result showed that the hybrid clustering made the right clustering decision more than 90% of the time when there were disagreements. In addition, we present anecdotal examples from the clustering results to demonstrate the merit of the hybrid approach. Overall, the experiment results confirm that the performance of the hybrid approach is more versatile than either of the two underlying algorithms individually in various environments, thus demonstrating how these two different algorithms complement each other to hold up the performance together as a hybrid even when one algorithm performs poorly.

All source codes and datasets, including the gold standards, are available from Github at https://github.com/ali-javed/hybrid_semantic.

The remainder of the paper is organized as follows. Section 2 provides some background knowledge. Section 3 discusses related work. Section 4 discusses the base algorithms used in the design of the hybrid algorithm. Section 5 presents the details of the hybrid algorithm and its evaluation against the two base algorithms. Section 6 summarizes the paper and suggests future work.

2. Background

This section provides some background knowledge needed for the readers to understand this paper.

2.1. WordNet – synset hierarchy and similarity measure

WordNet is a free and publicly available lexical database of English language [24]. It groups words into sets of synonyms called synsets. Each word in WordNet must point to at least one synset, and each synset must point to at least one word. Hence, there is a many-to-many relationship between synsets and words. Synsets in WordNet are interlinked by their semantics and lexical relationships, which results in a network of meaningful related words and their senses.

Table 1 shows an example synset. The synset contains four different senses – e.g., "desert" meaning "arid land with little or

Table 1
Example senses in a synset for the word "desert" and their meanings.

Sense	Meaning
desert.n.01	Arid land with little or no vegetation
abandon.v.05	Leave someone who needs or counts on you; leave in the lurch
defect.v.01	desert (a cause, a country or an army), often in order to join the opposing cause, country, or army
desert.v.03	Leave behind

no vegetation", "desert" meaning "to leave someone who needs or counts on you". All of these senses are linked to each other using the semantic and lexical relationships. For example "oasis.n.01"(meaning "a fertile tract in a desert") is a meronym (i.e., part) of "desert.n.01".

Given this network of relationships, WordNet provides different API functions that allow for the calculation of semantic similarity between synsets. The Wu–Palmer [25] similarity measure is used in this paper in order to stay consistent with the base algorithm by Vicient and Moreano [5]. In a lexical database like WordNet synset database, where senses are organized in a hierarchical structure, the Wu–Palmer similarity between two senses s_1 and s_2, denoted as $sim_{WP}(s_1, s_2)$, is defined as

$$sim_{WP}(s_1, s_2) = \frac{2 \cdot depth(LCS(s_1, s_2))}{depth(s_1) + depth(s_2)} \quad (1)$$

where $LCS(s_1, s_2)$ is the least common subsumer (i.e., lowest common ancestor) of s_1 and s_2 in the hierarchy of synsets. In Wu and Palmer [25], the formula for calculating the similarity is thus given as $\frac{2N_3}{N_1+N_2+2N_3}$ where N_1 is the number of nodes on the path from s_1 to s_3 ($\equiv LCS(s_1, s_2)$), N_2 is the number of nodes on the path from s_2 to s_3, and N_3 is the number of nodes on the path from s_3 to the root.

This WordNet functionality is used to calculate the semantic similarity between hashtags in this paper, that is, by grounding hashtags to specific senses (called "semantic grounding") and calculating the similarity between the senses.

Note that some hashtags cannot be semantically grounded (e.g., #fgddfdv – there is no English word this matches to, or a brand name like "#Honda" that can not be found in a dictionary). In addition, some hashtags may be incorrectly grounded – for example, grounding '4' in #date4fun to the number 4 is most likely incorrect because '4' actually represents the word "for" (which sounds the same as '4') and hence more appropriate grounding is with the word "fun" than with the number 4. The context determines which sense is truly the representative of the hashtag in a tweet.

2.2. Wikipedia – auxiliary categories

Wikipedia is by far the most popular crowdsourced encyclopedia. As mentioned above, not all hashtags can be grounded semantically using WordNet because many of them are simply not legitimate terms found in WordNet (e.g. #Honda). This situation is where Wikipedia can be used to look up those hashtags. Wikipedia provides auxiliary categories for each article. For example, when Wikipedia is queried for categories related to a page titled "Honda", it returns the following auxiliary categories [26]:

```
[Automotive companies of Japan',
 Companies based in Tokyo',
 Boat builders',
 Truck manufacturers',
 Vehicle manufacturing companies',
 ...
]
```

Auxiliary categories can be thought of as categories the page belongs to. In this example, if we are unable to look up the word "Honda" on WordNet, then, through the help of these auxiliary categories, we can relate the term to Japan, Automotive, Company, etc. There are several open source Wikipedia APIs available to achieve this purpose – for example, the Python library "wikipedia".

3. Related work

In Section 1 we introduced two sources for identifying the semantics of a hashtag as the lexical semantics extracted from metadata and the contextual semantics inferred from data (i.e., texts) containing the hashtag. In this section, we discuss other work related to hashtag clustering based on the two sources of hashtag semantics. Additionally, we discuss some existing hybrid clustering approaches for general clustering tasks.

3.1. Metadata-based lexical semantic clustering

As mentioned earlier, metadata-based semantic clustering of hashtags is a relatively new research problem in social media. The work by Vicient and Moreno [5] and our previous work [12,13] are, to the best of our knowledge, the only ones addressing exactly this problem. Vicient and Moreno [5] used WordNet and Wikipedia as a metadata source for identifying the lexical semantics of a hashtag. In their method, clustering decision is made at the *word* level, and it often leads to incorrect clusters. This was corrected in our work, where clustering was done at the *sense* level [12,13]. Note that the same word may have different senses and it is the sense that carries the semantics, as exemplified in Table 1.

In a related problem, Costa et al. [14] addressed hashtag-based *tweet classification* using metadata. They used a crowdsourcing platform to provide metadata in the form of hashtag clusters and classified tweets using the metadata.

In general, metadata is a useful resource to overcome the limitations coming from lack of quality in the data or complexity of the problem to solve, and has proven to be so in our work.

3.2. Text-based contextual semantic clustering

Text-based clustering of hashtags is a relatively well-addressed problem. It focuses on the tweets accompanying hashtags as the source of contextual semantics [18]. Representing tweets as a bag-of-words (a.k.a. vector space model) and processing them as such seems to be the most popular and commonly used approach [7,8,10,14–17,27]. More specifically, Tsur et al. [15,16] and Muntean et al. [7] used the text-based approach for hashtag clustering purposes. They appended tweets that belong to each unique hashtag into a unique document called "virtual document". These documents were then represented as vectors in the vector space model. Feng et al. [27] built on the virtual document concept by giving an additional weight to any hashtag that appears in a tweet. Rosa et al. [17] and Bhulai et al. [10] created *topical clusters* instead of hashtag clusters using the same bag-of-words model, where topics were predefined. In their work, an interesting issue spins around the inclusion of documents expanded from the links embedded in tweets. Rosa et al. [17] found that by expanding URL found in tweets, the performance of clustering actually degrades. They attribute the degradation to off-topic contents found in web pages linked through the URLs. Moreover, clustering large documents using the bag-of-words model is known to be a challenging task, as stated by Inderjit et al. [28]. For these reasons, we did not employ this expansion approach in our work, either. Costa et al. [14] used the solution to a hashtag clustering problem as the basis of their tweet classification problem. They classified a tweet as belonging to a specific hashtag based on the similarity

of the tweet with tweets belonging to the same hashtag. Park and Shin [8] used a similar approach for tweet search purposes.

For datasets that are gathered based on some co-occurrence patterns, a graph-based model [9,11] is used instead of the vector space model. Wang et al. [9] used the clustering based on co-occurrence of hashtags for sentiment analysis purposes. Teufl and Kraxberger [11], similarly through a graph-based model, used the co-occurrence based on words in a tweet for event detection purposes.

Stilo and Velardi [29,30] clustered hashtag senses based on *temporal* co-occurrence of a hashtag with other hashtags. Temporal co-occurrence applies to hashtags with similar usage patterns over time, based on the idea that hashtags with similar temporal usage are semantically related. These "temporal senses" are different from the lexical senses used in our paper, which are derived from metadata and are independent of the temporal usage.

3.3. Hybrid clustering

The notion of hybrid clustering has been introduced in other work as well. Currently popular hybrid approaches focus on combining the strengths of different clustering *methods* [20–23]. More specifically, Ghafarzadeh and Bouyer [20] used the "artificial bee colony" algorithm and k-means to balance the diversity and convergence ability of a new hybrid clustering algorithm. Wei-Chang et al. [21] used the power of swarm optimization to create a modified version of k-harmonic means that makes it less vulnerable to being trapped in local optima. Dai et al. [22] used the output of "canopy algorithm" as the seed to fuzzy c-means to improve initialization in fuzzy c-means. Yang and Jiang [23] used bagging and boosting to create a hybrid clustering algorithm that combines the strengths of both.

To the best of our knowledge, there has been no work for *hybrid semantic* clustering as ours, which focuses on using the complementary strengths of lexical and contextual semantics to create hybrid semantics for the *same* clustering method.

4. Base algorithms

In this section, we present two algorithms upon which our proposed hybrid clustering algorithm is designed. One algorithm is adapted from our prior work [12,13], and clusters hashtags by their *lexical* semantics identified from metadata, i.e., dictionaries and sources other than micro-message texts. We call it *metadata-based clustering*. The other algorithm is adopted from the algorithms proposed by Tsur et al. [15] and Muntean et al. [7]. It clusters hashtags by their *contextual* semantics identified from terms included in data, i.e., micro-message texts themselves. We call it *text-based clustering*. Additionally, in this section we compare the strengths and weaknesses of these two base algorithms.

4.1. Metadata-based clustering of hashtags

As already mentioned in Section 3, this approach uses WordNet and Wikipedia as metadata for identifying the lexical semantics of a hashtag. There are three major steps in this semantic clustering algorithm [5,12,13]: (a) semantic grounding, (b) similarity matrix construction, and (c) clustering. Algorithm 1 summarizes the steps.

Algorithm 1 creates *sense-level* overlapping clusters of hashtags, which is made possible by an enhancement in Stage 2 (similarity matrix calculation) of the word-level algorithm by Vicient and Moreneo [5]. Sense-level clusters are more accurate than word-level clusters, as demonstrated in our prior work [12,13], and are input to the hybrid clustering algorithm we will discuss in Section 5.

Algorithm 1 Metadata-based hashtag clustering [5,12].

Input: a set H of hashtags
Output: metadata-based hashtag clusters
Return: a set H' of semantically groundable hashtags

Stage 1 (Semantic grounding):

1: For each hashtag h in H, perform the step 1a below.

a: Look up h from WordNet. If h is found then *append* the synset of h to a list (LC_h). Otherwise, segment h into multiple words and drop the leftmost word and then try Step 1a again using the reduced h; repeat this until either a match is found from WordNet or no more word is left in h.

2: For each h in H that has an empty list LC_h (i.e., no match found from WordNet), look up h in Wikipedia. If an article matching h is found in Wikipedia, acquire the list of auxiliary categories for the article (see Section 2.2), extract main nouns from the auxiliary categories, and then, for each main noun extracted, perform the step 1a above using the main noun as h.

Stage 2 (Similarity matrix construction):

3: Discard from H any hashtag h that has an empty LC_h.

4: Given the remaining (i.e., semantically groundable) hashtags, H', construct a similarity matrix in the following two steps:

a: For each pair of hashtags h_i and h_j ($h_i \neq h_j$) in H', calculate the maximum pairwise similarity of senses between all s_p in LC_{h_i} and all s_q in LC_{h_j}, and then save the resulting maximum pairwise similarity, *maxSim*, and the corresponding pair of senses as a triplet ($h_i.s_p$, $h_j.s_q$, *maxSim*) in a list LH_s.

b: Count the number $|\hat{S}|$ of distinct hashtag senses in LH_s and initialize a similarity matrix $\mathbf{M}[|\hat{S}|, |\hat{S}|]$ as a $\mathbf{0}$ matrix. Then, for each triplet ($h_i.s_p$, $h_j.s_q$, *maxSim*) in LH_s, update the entry $\mathbf{M}[m, n]$ to *maxSim*, where (m, n) is the matrix index for ($h_i.s_p$, $h_j.s_q$).

Stage 3 (Clustering):

5: Perform hierarchical clustering of the hashtags in H' using the similarity matrix resulting from Stage 2, and extract flat clusters from the hierarchical clusters using a tunable distance threshold.

6: Return H'.

Let us focus our explanation on Stage 2 of the algorithm, as it is the stage intimately related to clustering. First, hashtags associated with an empty list of senses are discarded; in other words, hashtags that did not match any WordNet entry, either by themselves or by using word segmentation technique, and also had no entry found in Wikipedia are discarded. We call the remaining hashtags the "*semantically groundable*" hashtags and denote the set of them as H'. Note that hashtags in social media can be quite casual and, therefore, some of them may not be semantically groundable.

Next, given H', a similarity matrix between hashtag senses is constructed using a two-step algorithm (proposed in our previous work [12,13]). In the first step, for each pair of hashtags, the algorithm finds the pair of their senses ($h_i.s_p$, $h_j.s_q$) that gives the maximum similarity value and saves the found pair of hashtag senses and the similarity value between them in a list LH_s. Then, in the second step, for each triplet element ($h_i.s_p$, $h_j.s_q$, *maxSim*) in LH_s, the algorithm enters the maximum similarity value *maxSim* into a similarity matrix at the index assigned to the pair of senses ($h_i.s_p$, $h_j.s_q$). The total number of maximum similarity values in LH_s is $|\hat{S}|^2$, where $\hat{S}$ is the set of distinct hashtag senses in LH_s. The remaining entries are initialized to 0 and remain 0, as they are for pairs of senses that do not represent maximum similarity pair between any hashtags.

The last step is hierarchical clustering of the hashtags using the similarity matrix. We use bottom-up (or agglomerate) strategy because it is conceptually simpler than top-down [31]. For bottom-up strategy, several distance measurement methods are available to provide linkage criteria for building up a hierarchy of clusters. Among them, nearest point method and unweighted pair group method with arithmetic mean (UPGMA) are used most commonly, and are used here as well. To generate output clusters, "flat clusters" are extracted from the hierarchy. There are multiple possible criteria for doing that [32], and we use the distance criterion. That is, flat clusters are formed from the hierarchy when no two clusters are farther than the given distance threshold.

The complexity of this algorithm is $O(|H'|^2)$. Specifically, the step 1 takes $\Theta(|H|)$, as it is done for each hashtag in H; the step 2 takes $O(|H|)$ as it is done only for some of the hashtags (i.e., those that have an empty list LC_h); the step 3 takes $\Theta(|H|)$; the step 4 takes $\Theta(|H'|^2)$ as it is done for each pair of hashtags in H'; the step 5 takes $O(|H'|^2)$ in our implementation of bottom-up hierarchical clustering [33], which is the best one we have found. The number of semantically groundable hashtags, $|H'|$, is smaller than the number of the input hashtags, $|H|$, but not by a large margin, so $O|H'|^2$ is the dominant complexity of this algorithm overall.

4.2. Text-based clustering of hashtags

Text-based clustering of hashtags uses the terms in the text containing a hashtag to calculate distance between hashtags. The resulting distance is called the contextual distance, as it is based on contextual semantics of the hashtag [18]. We adopted the algorithm proposed by Tsur et al. [15], called the Scalable Multi-Stage Clustering (SMSC). This algorithm is meant to cluster tweets, but the first part of the algorithm can be used to cluster hashtags.

Algorithm 2 outlines the text-based hashtag clustering algorithm, adopted from the first part of the SMSC algorithm and augmented with word stemming and stop word removal capabilities. These two capabilities have been added to improve the semantic matching, reduce the feature space, avoid over-fitting, and help protect against spelling mistakes, as was done in other works [7,8,11,14].

Algorithm 2 Text-based hashtag clustering [15].

Input: set T of tweets, set H of hashtags
Output: text-based hashtag clusters

Stage 1 (Virtual document creation):

1: For each hashtag h in H, create a virtual document d_h by concatenating all tweet messages from T that contain the hashtag h. (If a tweet message contains more than one hashtag, then it is concatenated to more than one virtual document.The number of virtual documents thus createdis $|H|$ – the number of distinct hashtags in H.)
2: Remove common stop words from the virtual documents and apply word stemming techniques to reduce stemmed words to the root word.

Stage 2 (Similarity matrix construction):

3: Using the vector space model, represent each virtual document d_h as a feature vector of the words in it.
4: For each pair of hashtags h_i and h_j $(h_i \neq h_j)$ in H, calculate the cosine similarity measure between vectors of the two virtual documents d_{h_i} and d_{h_j}, and then enter the calculated measure, $sim(d_{h_i}, d_{h_j})$, into a similarity matrix $\mathbf{M}[|H|, |H|]$.

Stage 3 (Clustering):

5: Perform hierarchical clustering of the hashtags in H using the similarity matrix resulting from Stage 2, and extract flat clusters from the hierarchical clusters using a tunable distance threshold.

An interesting note is that if a tweet message contains more than one hashtag, it is added to multiple virtual documents. So, if two or more hashtags co-occur in a significant number of tweets, their virtual documents are similar to one another because they contain a lot of the same tweet messages. Algorithm 2 thus can also indirectly capture co-occurrence relationship among hashtags to some extent (more about this as the future work in Section 6.2).

The complexity of this algorithm with respect to the number of hashtags, $|H|$, is $O(|H|^2)$. Specifically, the step 1 takes $\Theta(|H|)$ as it is done for each hashtag; the step 2 takes $\Theta(|T|)$ as it is done for all tweet messages (divided into virtual documents); the step 3 takes $\Theta(|H|)$ as it is done for each virtual document and there is one virtual document per hashtag; the step 4 takes $\Theta(|H|^2)$ as it is done for each pair of hashtags; the step 5 takes $O(|H|^2)$ in our implementation of bottom-up hierarchical clustering [33], which is the same algorithm as used in the step 5 of Algorithm 1. Thus, the total complexity is $\max(O(|H|^2), \Theta(|T|))$.

4.3. Strengths and weaknesses of the two approaches

Each of the two base algorithms has its own strengths and weaknesses. We briefly outline them here at a conceptual level.

In the metadata-based semantic hashtag clustering, dictionaries (e.g., lexical database, encyclopedia – collectively referred to as metadata) are the source of lexical semantics and are crucial to generating quality clusters. Its strength stems from the fact that much of the quality of *semantic* clustering output depends on the literal meaning of hashtags that can be found from these metadata sources, aside from how hashtags are used in micro-messages. Besides, its reliance on metadata makes it an extensible approach, as the quality of clustering output will improve as metadata sources improve.

These positive aspects can also work against the approach. Its heavy reliance on metadata and word-breaking is not always successful in decoding the meaning of a hashtag (which is not restricted by any rule) correctly, and often times more validation is needed to handle hashtags that are complex or grammatically incorrect. For tweet messages in a controlled, specific user space and possibly limited to a specific purpose, the semantic clustering approach can perform considerably well. The tweet messages available from Symplur [34] is a good example that meets the criterion. Symplur specializes in tweets and hashtags that are related to oncology in healthcare domain.

The text-based semantic hashtag clustering is contrasted with the metadata-based approach in that the former relies on the contents of micro-messages (which can be referred to as data as opposed to metadata) whereas the latter solely relies on the semantics of the language as identified using metadata. Text-based clustering hence is a source of *contextual* semantics of a hashtag. Its independence from metadata makes the approach resilient to poor linguistic quality of micro-messages and hashtags.

Its sole dependence on data (i.e. micro-message text), however, can also work against the approach. It is a common situation in microblogging platforms to have a high number of unique hashtags, hence not enough data associated with each hashtag. This lack of data to work with adversely affects the performance of text-based clustering.

Based on the strengths and weaknesses of these two algorithms, they have potential to complement each other. These complementary abilities are exactly what we exploit to build a *hybrid* approach to semantic hashtag clustering in Section 5.

5. Hybrid semantic clustering algorithm

In this section, we present the hybrid clustering algorithm and then illustrate its working with a toy example. Evaluations of the hybrid algorithm are presented in three different ways: gold standard test, pairwise disagreement test, and anecdotal examples.

5.1. Algorithm

Algorithm 3 outlines the hybrid clustering algorithm based on the consensus graph approach. The algorithm first builds metadata-based clusters and text-based clusters by calling their respective clustering algorithms, and then builds a hybrid similarity matrix based on the consensus from the two sets of clusters and performs hierarchical clustering using the similarity matrix.

Algorithm 3 Hybrid clustering.

Input: tweets, a set H of hashtags
Output: hybrid clusters of semantically groundable hashtags (H').

Stage 1 (Base clustering):

1: Perform metadata-based clustering of hashtags in H (see Algorithm 1) to obtain metadata-based clusters and return semantically groundable hashtags (H').
2: Perform text-based clustering of hashtags in H' using tweets (see Algorithm 2) to obtain text-based clusters.

Stage 2 (Hybrid similarity matrix construction):

3: Initialize a similarity matrix $\mathbf{M}[|H'|, |H'|]$ as a $\mathbf{0}$ matrix. Then, for each pair of hashtags h_i and h_j in H', perform the following two steps:

a: If the metadata-based cluster assignments of h_i and h_j are the same, then increment $\mathbf{M}[i, j]$ by 0.5.
b: If the text-based cluster assignments of h_i and h_j are the same, then increment $\mathbf{M}[i, j]$ by 0.5.

Stage 3 (Clustering):

4: Perform hierarchical clustering of the hashtags in H' using the hybrid similarity matrix resulting from Stage 2, and extract flat clusters from the hierarchical clusters using a tunable distance threshold.

In order to construct the hybrid similarity matrix, we use a variant of consensus clustering that is based on the concept of *consensus graph* [35,36]. This method is adequate enough for our purpose of building a hybrid based on only two clustering outputs. It first creates a consensus graph, where each node represents a cluster item and each edge represents a pair of cluster items. Specifically, each edge (i, j), $i \neq j$, has a weight representing the similarity between the two items i and j, defined as t_{ij}/n where t_{ij} is the number of clustering outputs that contain the items i and j in the same cluster and n is the number of different clustering outputs considered. Once a consensus graph is created, then clustering is performed using the graph, that is, using the adjacency matrix representation of the graph as the similarity matrix.

The hybrid similarity matrix thus constructed represents an undirected weighted graph where each vertex represents a hashtag and each edge represents a pair of hashtags. The weight of an edge represents the hybrid distance between the hashtags represented by the two end vertexes. For each pair of hashtags, h_i and h_j, if they belong to one or more same clusters in metadata-based clustering, then the similarity is incremented by 0.5. The same is done for text-based clustering, though this time they can belong to at most one cluster in common, as hashtags are not replicated in text-based clustering. Hence, if the hashtags are in the same cluster for both metadata-based clustering and text-based clustering, then the similarity becomes 1.0.

Table 2
Cluster assignment from metadata-based clustering.

Hashtag	Hashtag sense	Cluster using UPGMA
#tree	tree.n.01	1
#date	date.n.02	1
#september	september.n.01	2
#date	date.n.06	2
#fruit	fruit.n.01	3
#date	date.n.08	3

Table 3
Cluster assignment from text-based clustering.

Hashtag	cluster assignment
#tree	1
#date	1
#fruit	2
#september	3

Hashtag	#tree	#date	#fruit	#september
#tree	1.0	1.0	0.0	0.0
#date	1.0	1.0	0.5	0.5
#fruit	0.0	0.5	1.0	0.0
#september	0.0	0.5	0.0	1.0

Fig. 1. Hybrid similarity matrix for the toy example.

Once a hybrid similarity matrix is built, then any clustering algorithm can be used to generate clusters. We used the same bottom-up hierarchical clustering used by the base algorithms. (In our implementation, the clustering algorithm actually used a distance matrix instead of a similarity matrix, where distance is simply 1's complement of similarity.)

The hybrid approach may incur the overhead of running the two base algorithms, but the overhead does not increase the complexity, which is $O(|H'|^2)$. Specifically, the step 1, which runs Algorithm 1, takes $O(|H'|^2)$; the step 2, which runs Algorithm 2, takes $O(|H'|^2)$ (note the input to Algorithm 2 here is H', not H); the step 3 takes $\Theta(|H'|^2)$, as it is done for each pair of hashtags; the step 4, using the same bottom-up hierarchical clustering, takes $O(|H'|^2)$.

5.2. A toy example

Let us illustrate the hybrid clustering algorithm using a toy example. This example employs the hashtags #september, #date, #fruit, and #tree. Table 2 show a cluster assignment made by metadata-based clustering using UPGMA with 0.5 as the distance threshold, where #tree and #date, #september and #date, and #fruit and #date are, respectively, in the same (overlapping) clusters. (Note that #date belongs to different clusters in different senses, i.e., n.02, n.06, and n.08.) Metadata-based clusters are formed at the sense level, but for the purpose of integration with text-based clusters, are expressed at the word level. Additionally, Table 3 shows a cluster assignment from text-based clustering, where #tree and #date are in the same cluster whereas #fruit and #september are not in any cluster.

Based on these two cluster assignments, the hybrid similarity matrix is as shown in Fig. 1. Fig. 2 is the consensus graph representing the similarity matrix. We see that, for example, the hybrid similarity value is 1.0 between #tree and #date since they are together in both a metadata-based cluster and a text-based cluster, 0.5 between #fruit and #date since they are together only in a metadata-based cluster, and 0.0 between #september and #fruit

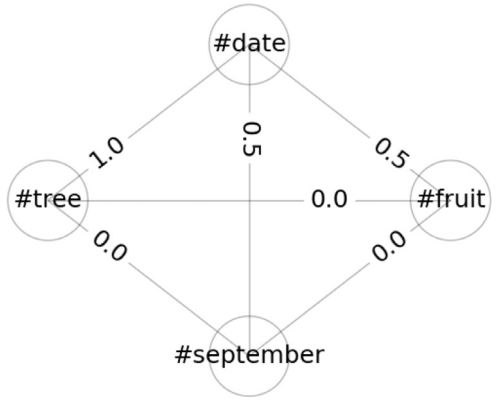


Fig. 2. Hybrid consensus graph for the toy example. (A node represents a hashtag, and an edge represents the similarity between two hashtags.)

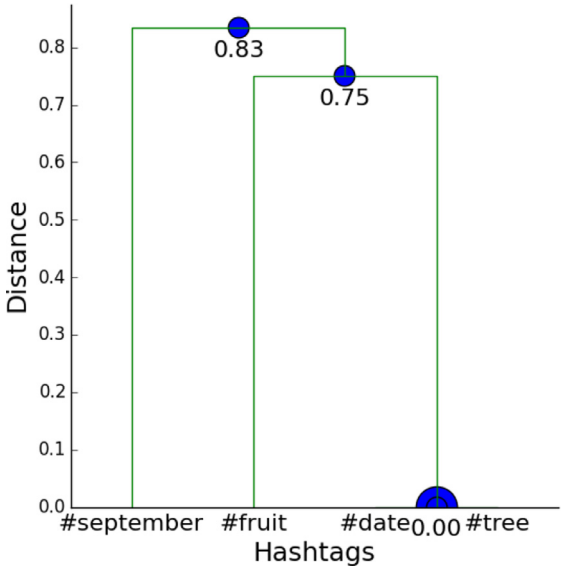


Fig. 3. Dendrogram of output clusters in the toy example.

since they are separate in both metadata-based and text-based clusters.

Fig. 3 shows a dendrogram of the hierarchical clusters formed by using the hybrid distance matrix (distance = 1 - similarity in Fig. 1) and UPGMA as the distance measure. It shows how flat clusters are extracted. That is, (1) #tree and #date are always in the same cluster with distance 0.0 (i.e., similarity 1.0) between them; (2) if the threshold is in [0.75, 0.83), then either #fruit or #september joins the cluster because both are at the UPGMA distance 0.75 (i.e., UPGMA similarity 0.25 = (0.0 + 0.5)/2) from the cluster that contains #tree and #date – let us assume the tie is broken in the alphabetical order and thus #fruit is included; (3) then, if the UPGMA distance is 0.83 (i.e., UPGMA similarity 0.17 = (0.0 + 0.5 + 0.0)/3)) or larger, then #september joins the cluster as well.

5.3. Evaluations

The objective of the evaluations is to compare the hybrid algorithm with the two base algorithms. The focus is on examining the versatility (mentioned in Section 1) of the hybrid algorithm under different circumstances that affect the performances of the two base algorithms differently. To this end, two sets of experiments have been conducted to compare the accuracy between the hybrid algorithm and each of the two base algorithms. The first set of experiments is a gold standard test using ground truth cluster sets, and the second set of experiments is a "pairwise disagreement test", which is a larger-scale experiment with a focus on the disagreements in clustering decisions. Additionally, qualitative comparison has been done through anecdotal examples.

In this section, we present the experiment setup in Section 5.3.1 and the two sets of experiments and their results in Section 5.3.2 and Section 5.3.3, respectively. Anecdotal examples are presented in Section 5.3.4. Additionally, orthogonal to these experiments, in Appendix A we compare between the sense-level and the word-level as the metadata-based clustering in terms of the resulting hybrid clusters.

5.3.1. Experiment setup

Data sets. Two kinds of tweet datasets are used for experiments – referred to as the *Symplur* dataset and the *Random* dataset, respectively. The Symplur dataset was acquired from the Symplur Healthcare Hashtag Project [34] by manually extracting tweets. It contains 2,910 tweets and 1,010 hashtags altogether. This dataset is specific to the healthcare domain and serves the mission of making "the use of Twitter more accessible for providers and the healthcare community as a whole." [37]. The Random dataset is made of approximately 72 million tweets randomly collected from all public user accounts with no selection bias through the Twitter API from January 2014 to January 2015. Thus, this dataset encompasses multiple arbitrary domains and is completely open-ended, grabbing all tweet message indiscriminately.

The two kinds of datasets are contrasted in two key aspects. First, hashtags in the Symplur dataset tend to show clearer lexical semantics, compared with the Random dataset which contains many hashtags showing ambiguous or even misleading lexical semantics. Second, the Random dataset contains a variety of more common hashtags (i.e., hashtags that have more tweets associated to them) and provides a more unbiased picture of hashtags in the Twittersphere. These contrasts are expected, given the distinction between the two in terms of the focuses of their domains and the missions of their operations (or lack of them).

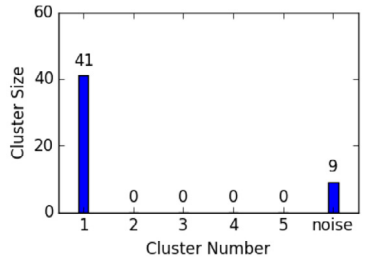
Parameters. When running the clustering algorithms, we need to set two parameters – the distance measure (i.e., UPGMA, nearest-neighbor) for hierarchical clustering and the distance threshold for extracting flat clusters from the resulting hierarchy of clusters. We took the best-result approach to determine the parameter values, that is, tried both distance measures and different distance threshold values and picked the ones that produced the best result. Specifically, for each distance measure, the distance threshold value was varied in gradient ascent at the increment of 0.05 starting with 0.5 to find the threshold value that gives the maximum clustering performance based on the f-score.

Platform. All algorithms were implemented in Python, and the experiments were performed on a computer with OS X operating system, 2.6 GHz Intel Core i5 processor, and 8 GB 1600 MHz DDR3 memory.

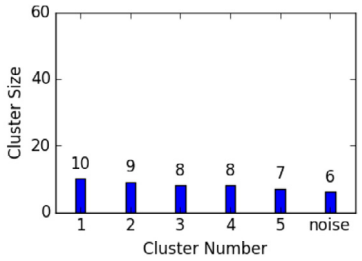
5.3.2. Gold standard test

Ground truths. We have built seven different cluster sets to be used as the ground truth (GT) for evaluating the hybrid algorithm against the base algorithms with respect to the accuracy of output clusters. Profiles of the GT cluster sets are shown in Table 4, and the distribution of the sizes of the clusters and their associated themes are shown in Fig. 4. We will refer to the three GT cluster sets GT-R1, GT-R2, and GT-R3 as the "Random GT" and the three, GT-S1, GT-S2, and GT-S3, as the "Symplur GT", and the combined one, GT-All, as the "Combined GT".

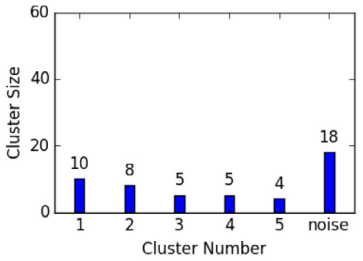
The steps of constructing these GT cluster sets are as follows. First, from the Random tweet dataset, approximately 2.5 million



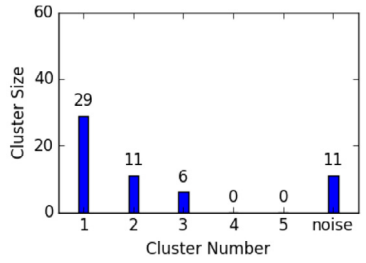
(a) GT-R1 (1. pornography).



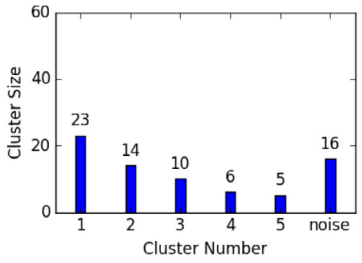
(b) GT-R2 (1. numbers; 2. cell phone; 3. days; 4. adult; 5.geographic).



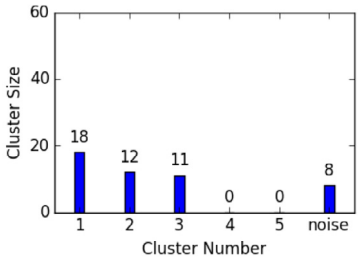
(c) GT-R3 (1. pornography; 2. photography; 3. media; 4. social media; 5.computers).



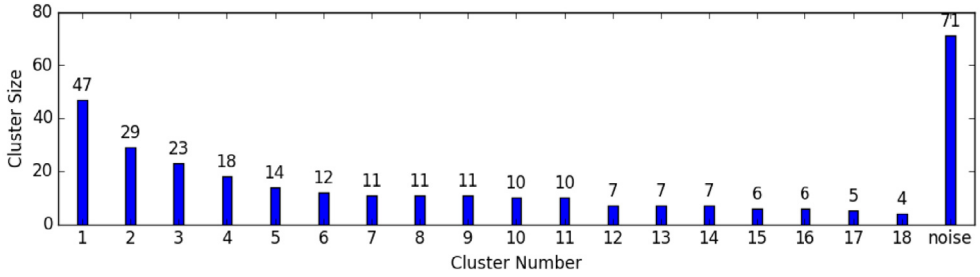
(d) GT-S1 (1. disease; 2. technology; 3. med imaging).



(e) GT-S2 (1. chat; 2. education; 3. care; 4. community; 5.diagnosis).



(f) GT-S3 (1. health; 2. medicine; 3. job roles).



(g) GT-All (1. pornography; 2. diseases; 3. chat; 4. health; 5. education; 6. medicine; 7. job roles; 8. technology; 9. cell phone; 10. care; 11. numbers; 12. photography; 13. adult; 14. geographic; 15. med imagining; 16. community; 17. diagnosis; 18. social media).

Fig. 4. Distribution of the size of clusters in each ground truth cluster set.

Table 4
Profiles of the ground truth cluster sets.

Name	# hashtags	Source	Min # tweets
GT-R1	50	Random	20
GT-R2	48	Random	20
GT-R3	50	Random	20
GT-S1	57	Symplur	1
GT-S2	74	Symplur	1
GT-S3	49	Symplur	1
GT-All	309	GT-R1 ∪ GT-R2 ∪ GT-R3 ∪ GT-S1 ∪ GT-S2 ∪ GT-S3	

tweets were extracted in chronological order. These 2.5 million tweets contained 708 hashtags that had 20 or more tweets associated with them. From these 708 hashtags, we selected approximately 50 hashtags based on their lexical semantics. We repeated this selection three times to generate three different GT cluster sets through random sampling with replacement, while independently assigning the themes of selected hashtags. Second, from the Symplur tweet dataset, we prepared the GT cluster sets based on observed hashtag semantics. Out of the 1,010 hashtags, we manually annotated the semantics to choose 230 hashtags and classified them into 15 clusters. The remaining hashtags were classified as noise. From the 15 clusters we randomly picked one cluster at a time, merging them, until the total number of hashtags in the selected clusters reached approximately 50. This step was repeated twice to generate two more GT cluster sets, each time selecting from the remaining clusters. Third, each resulting GT cluster set (based on lexical semantics) was again manually re-clustered based on the actual themes that were read from the *text contents* as well as hashtags – this emulates the hybrid clustering. Finally, we created another GT cluster set by merging the six GT cluster sets and making necessary adjustments in clustering manually. These steps were followed by three people who worked independently in the selection process.

Accuracy measures. The *f-score*, commonly used to evaluate clusters in conjunction with recall and precision, is used as the accu-

Table 5
Weighted average of pairwise maximum f-scores (f^a-score) comparison among the three algorithms.

	Text-based	Metadata-based	Hybrid
GT-R1	0.85	0.52	0.85
GT-R2	0.73	0.70	0.85
GT-R3	0.42	0.42	0.47
Average	0.69	0.56	0.74
GT-S1	0.22	0.75	0.67
GT-S2	0.20	0.71	0.63
GT-S3	0.25	0.69	0.57
Average	0.22	0.72	0.63
GT-All	0.45	0.52	0.55

racy measure. In this experiment, the f-score is calculated for each pair of a cluster in the GT cluster set and a cluster in the evaluated algorithm's output cluster set. Then, the final f-score resulting from the comparison of the two cluster sets is obtained in two different ways, depending on the purpose of the evaluation. For the purpose of evaluating individual output clusters, the pairwise maximum (i.e., "best match") f-score, denoted as f^m-score, is used as the final score. Given a GT cluster set G_i matched against an output cluster set $\mathbf{C}$, the f^m-score is obtained as

$$f^m\text{-score}(G_i, \mathbf{C}) = \max_{C_j \in \mathbf{C} \wedge \text{f-score}(G_i, C_j) > 0} \text{f-score}(G_i, C_j) \quad (2)$$

where the pairwise matching is one-to-one between $\mathbf{G}$ and $\mathbf{C}$. In addition, for comparing overall accuracy of the entire set of output clusters, the weighted average of pairwise maximum f-scores, denoted as f^a-score, is used instead. Given a GT cluster set $\mathbf{G}$ and an output cluster set $\mathbf{C}$, the f^a-score is calculated as

$$f^a\text{-score}(\mathbf{G}, \mathbf{C}) = \frac{\sum_{G_i \in \mathbf{G}} (f^m\text{-score}(G_i, \mathbf{C}) \times |G_i|)}{\sum_{G_i \in \mathbf{G}} |G_i|} \quad (3)$$

Test results. Fig. 5 shows the accuracy (pairwise maximum f-score) achieved by the hybrid algorithm compared with those of the base algorithms for the individual GT clusters in each of the seven GT cluster sets. Table 5 shows the weighted average of maximum pairwise f-scores over all GT clusters in each GT cluster set. Table 6 provides the details of these results.

The results show that the hybrid clustering is the "versatile" performer based on these measures. That is, the hybrid clustering is the best performer in a majority of cases of the GT cluster sets and, even when it is not, it is consistently the second best performer and the difference form the best performer is marginal.

More specifically, for the Random GT (GT-R1, R2, R3), the hybrid clustering achieves the highest f^m-score against 8 out of 11 GT clusters and the highest f^a-score for all three GT cluster sets. On the other hand, for the Symplur GT (GT-S1, S2, S3), the hybrid clustering achieves a f^m-score that is second to the metadata-based clustering against 10 out of 11 GT clusters, and the f^a-score is also second to the metadata-based clustering, but the difference is only 12.5%, which is relatively smaller than the difference of 69.4% that text-based clustering has from the metadata-based clustering. Besides, the total average considering all GT cluster sets (GT-All) shows that the hybrid clustering achieves the highest f^a-score overall.

Let us share some further insight into the relative performances of the two base algorithms with respect to the two GT cluster sets. The text-based clustering performs much better with the Random GT than with the Symplur GT. The reason is that for the Random GT we considered only hashtags with 20 or more tweets associated with them, and therefore the hashtags were amenable to the bag-of-words approach, whereas for the Symplur GT we dropped the limit and therefore many hashtags had only one or two tweets associated with them. The metadata-based clustering performs better

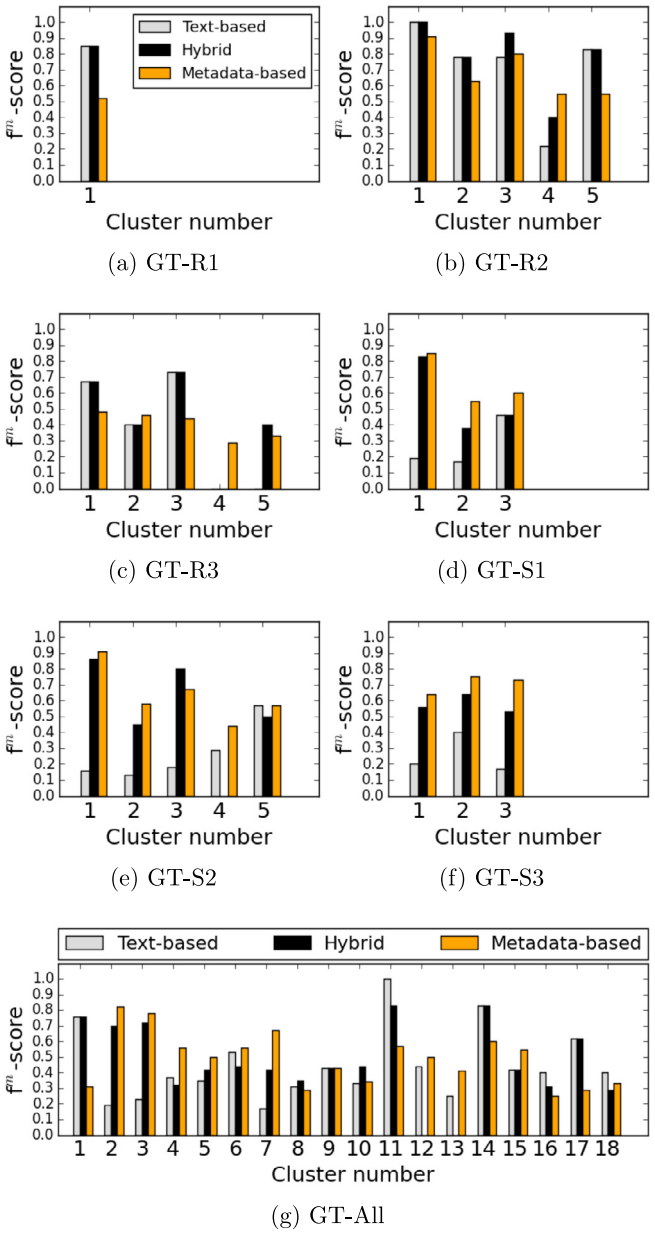


Fig. 5. Pairwise maximum f-score (f^m-score) comparison among the three algorithms.

with the Symplur GT than the Random GT, while the margin is not as conspicuous as the case of the text-based clustering. It stems from the fact that the domain-specific and mission-specific Symplur tweet dataset offers clear lexical semantics to all hashtags in the Symplur GT whereas, in the Random GT, the way we picked hashtags for this semantic clustering assured some sort of lexical semantics albeit not so clear as those in the Symplur GT.

5.3.3. Pairwise disagreement test

This test aims to conduct a larger-scale (i.e., with more hashtags) evaluation. In this case, building a set of ground truth clusters is not feasible due to the prohibitive overhead of manual efforts. Therefore, we conducted the "pairwise disagreement test", which focuses on the instances of disagreements occurring in clustering decisions between the hybrid algorithm and each of the two base algorithms and counting how often the hybrid algorithm's decision is right.

Table 6
Hybrid clustering gold standard test results.

Ground truth clusters			Text-based clusters				Metadata based clusters				Hybrid clusters			
GT	Id	Size	Recall	Precision	f^m-score	Size	Recall	Precision	f^m-score	Size	Recall	Precision	f^m-score	Size
GT-R1	1	41	0.78	0.94	0.85	34	0.37	0.89	0.52	18	0.73	1.00	0.85	30
GT-R2	1	10	1.00	1.00	1.00	10	1.00	0.83	0.91	12	1.00	1.00	1.00	10
	2	9	0.78	0.78	0.78	9	0.56	0.71	0.63	7	0.78	0.78	0.78	9
	3	8	0.88	0.70	0.78	10	1.00	0.67	0.80	12	0.88	1.00	0.93	7
	4	8	0.13	1.00	0.22	1	0.38	1.00	0.55	3	0.25	1.00	0.40	2
	5	7	0.71	1.00	0.83	5	0.43	0.75	0.55	4	0.71	1.00	0.83	5
GT-R3	1	10	0.70	0.64	0.67	11	0.50	0.45	0.48	11	0.70	0.64	0.67	11
	2	8	0.25	1.00	0.40	2	0.38	0.60	0.46	5	0.25	1.00	0.40	2
	3	5	0.80	0.67	0.73	6	0.40	0.50	0.44	4	0.80	0.67	0.73	6
	4	5	0.00	0.00	0.00	0	0.20	0.50	0.29	2	0.00	0.00	0.00	0
	5	4	0.00	0.00	0.00	0	0.25	0.50	0.33	2	0.25	1.00	0.40	1
GT-S1	1	29	0.10	1.00	0.19	3	0.79	0.92	0.85	25	0.76	0.92	0.83	24
	2	11	0.09	1.00	0.17	1	0.45	0.71	0.56	7	0.27	0.60	0.38	5
	3	6	0.50	0.43	0.46	7	0.50	0.75	0.60	4	0.50	0.43	0.46	7
GT-S2	1	23	0.09	1.00	0.16	2	0.87	0.95	0.91	21	0.83	0.90	0.86	21
	2	14	0.07	1.00	0.13	1	0.50	0.70	0.58	10	0.36	0.63	0.45	8
	3	10	0.10	1.00	0.18	1	0.90	0.53	0.67	17	0.80	0.80	0.80	10
	4	6	0.17	1.00	0.29	1	0.33	0.67	0.44	3	0.00	0.00	0.00	0
	5	5	0.40	1.00	0.57	2	0.40	1.00	0.57	2	0.40	0.67	0.50	3
GT-S3	1	18	0.11	1.00	0.20	2	0.50	0.90	0.64	10	0.39	1.00	0.56	7
	2	12	0.25	1.00	0.40	3	0.75	0.75	0.75	12	0.58	0.70	0.64	10
	3	11	0.09	1.00	0.17	1	0.73	0.73	0.73	11	0.45	0.63	0.53	8
GT-All	1	47	0.89	0.67	0.76	63	0.36	0.27	0.31	63	0.89	0.67	0.76	63
	2	29	0.10	1.00	0.19	3	0.79	0.85	0.82	27	0.55	0.94	0.70	17
	3	23	0.13	1.00	0.23	3	0.87	0.71	0.78	28	0.61	0.88	0.72	16
	4	18	0.28	0.56	0.37	9	0.50	0.64	0.56	14	0.22	0.57	0.32	7
	5	14	0.21	1.00	0.35	3	0.50	0.50	0.50	14	0.36	0.50	0.42	10
	6	12	0.42	0.71	0.53	7	0.75	0.45	0.56	20	0.33	0.67	0.44	6
	7	11	0.09	1.00	0.17	1	0.73	0.62	0.67	13	0.36	0.50	0.42	8
	8	11	0.18	1.00	0.31	2	0.27	0.30	0.29	10	0.27	0.50	0.35	6
	9	11	0.27	1.00	0.43	3	0.27	1.00	0.43	3	0.27	1.00	0.43	3
	10	10	0.20	1.00	0.33	2	0.80	0.22	0.34	37	0.60	0.35	0.44	17
	11	10	1.00	1.00	1.00	10	1.00	0.40	0.57	25	1.00	0.71	0.83	14
	12	7	0.29	1.00	0.44	2	0.43	0.60	0.50	5	0.00	0.00	0.00	0
	13	7	0.14	1.00	0.25	1	1.00	0.26	0.41	27	0.00	0.00	0.00	0
	14	7	0.71	1.00	0.83	5	0.43	1.00	0.60	3	0.71	1.00	0.83	5
	15	6	0.67	0.31	0.42	13	0.50	0.60	0.55	5	0.67	0.31	0.42	13
	16	6	0.33	0.50	0.40	4	0.17	0.50	0.25	2	0.33	0.29	0.31	7
	17	5	0.80	0.50	0.62	8	0.20	0.50	0.29	2	0.80	0.50	0.62	8
	18	4	0.25	1.00	0.40	1	0.25	0.50	0.33	2	0.25	0.33	0.29	3

For each GT cluster set, the clusters are sorted in a decreasing order of the size.

Table 7
Cluster profiles for the uncontrolled dataset.

Cluster type	Number of clusters per size range					
	≥ 100	50–99	20–49	10–19	5–9	1–5
Text-based	0	1	0	0	13	1082
Metadata-based	2	8	47	81	19	3182
Hybrid	0	0	1	0	2	1292

Table 8
Cluster profiles for the controlled dataset.

Cluster type	Number of clusters per size range					
	≥ 100	50–99	20–49	10–19	5–9	1–5
Text-based	1	0	5	43	55	327
Metadata-based	3	6	52	96	218	3842
Hybrid	0	0	4	6	38	987

The clustering outputs in our work, reveal a very small granularity (i.e., a large number of very small clusters, many of them including only one or two hashtags) – see Tables 7 and 8. In this case, hashtags in most pairs have a high chance of being separated into different clusters, so clustering outputs from different algorithms have a high chance of agreeing on keeping the pairs separated into different clusters. Therefore, what is meaningful in our work is to focus on hashtags in those pairs that are *disagreed* in the clustering outputs from different algorithms.

Disagreement cases. There are two cases of disagreements between hybrid and text-based and between hybrid and metadata-based, respectively, as summarized below. Note that these two disagreement cases correspond to two of the four cases used to calculate the Rand distance, which is a well-known measure of cluster quality.

- Case 1 (base:together–hybrid:separate): two hastags are together in the same base cluster but separate in different hybrid clusters.
- Case 2 (base:separate–hybrid:together): two hashtags are separate in different base clusters but together in the same hybrid cluster.

Note that metadata-based semantic clusters are overlapping clusters [12], and here we consider two hashtags together if they are together in at least one cluster and separate if they are separate in all clusters.

Controlled and uncontrolled hashtags. There are two hashtag sets used in this experiment – *controlled* and *uncontrolled*. The controlled set consists of all 1,010 hashtags from the Symplur tweet

Table 9
Pairwise disagreement test results from the controlled dataset.

Base	Case	# instances	# hybrid correct
Text-based	Case 1	13059	45/50 (=90%)
	Case 2	0	N/A
Metadata-based	Case 1	72938	49/50 (=98%)
	Case 2	1633	27/50 =(54%)

Weighted arithmetic average of # instances hybrid was correct: 95.99%
Non-weighted arithmetic average of # instances hybrid was correct: 80.67%

Table 10
Pairwise disagreement test results from the uncontrolled dataset.

Base	Case	# instances	# hybrid correct
Text-based	Case 1	914	18/50 (= 36%)
	Case 2	0	N/A
Metadata-based	Case 1	63269	46/50 (=92%)
	Case 2	266	50/50 (=100%)

Weighted arithmetic average of # instances hybrid was correct: 91.24%
Non-weighted arithmetic average of # instances hybrid was correct: 76.00%

dataset and 1000 hashtags randomly extracted from the Random tweet dataset, where all random hashtags selected have 20 or more tweets associated with them. The intent of using this controlled set is to increase the number of the test hashtags while maintaining the same configuration used in the gold standard test. The uncontrolled hashtag set is extracted from the Random tweet dataset collected in the chronological order of their timestamps until 2000 unique hashtags are extracted. The intent of using this uncontrolled hashtag set is to see the effect of using a completely random set of hashtags without any requirement on the number of associated tweets.

With these two hashtag sets, the metadata-based clustering algorithm (Algorithm 1) found semantic grounding of hashtags for 73.28% of hashtags in the controlled set and for 79.21% hastags in the uncontrolled set. This was not too far off from what was reported in the original work published by Vicient and Moreno [5], where the percentage of semantically grounded hashtags stood at 86%.

Parameter setting. In this experiment, we set the values the two clustering parameters (e.g., distance measure, distance threshold) based on what we learned through the gold standard testing, that is, set the distance measure to UPGMA, which showed better result than the nearest neighbor method, and set the distance threshold to 0.5, which was the median of the optimal threshold values obtained for the different seven GT cluster sets. We believe these two values may well be used as default values by any user adapting our algorithm into practical use.

Test results. Tables 9 and 10 show the results of evaluating the hybrid algorithm by counting the number of times it was correct in the clustering decisions that were disagreed by the compared base algorithm. The judgment on the correctness of hybrid algorithm's decision was made based on the actual content of the tweet messages and the manual interpretation of the hashtag semantics. For this purpose, a sample of 50 hashtag-pairs were selected randomly in each case. From these tables, we see that the hybrid clustering made the correct decision in a majority of disagreement instances in all cases, with the exception of the case 1 against the text-based clustering when the uncontrolled hashtag set was used.

In order to aggregate the performance numbers of the hybrid clustering, Tables 9 and 10 also show the weighted average of the number of times the hybrid clustering was correct, where the weighting was by the number of disagreement instances in each case. Overall, the hybrid clustering was right more than 90% of the times on weighted average in the sampled disagreement cases for both controlled and uncontrolled hashtag sets. Even when the weighting was dropped, the hybrid clustering was right more than 75% of the time on average.

5.3.4. Anecdotal examples

We pick three interesting examples of hashtag clustering results and illustrate how the base algorithms and the hybrid algorithm clustered them. The first example showcases the hybrid clustering making the correct decision by using the power of text-based clustering to overcome the error of the metadata-based clustering. The second example showcases the hybrid clustering making a correct decision using the power of metadata-based clustering to overcome the error of the text-based clustering. The third example shows the hybrid clustering making an incorrect decision.

Anecdote 1. In this example, we consider a hashtag #foodporn. The hashtag #foodporn is a compound word of "food" and "porn". It may have the literal meaning of pornography related to food or the connotative meaning of hearty food. The clusters containing this hashtag, generated by the three clustering algorithms, are listed below.

- *Metadata-based cluster*: {#foodporn, #trainart, #thecoolestart, #summerporn, #rangrasiya, #programming, #premierdesigns, #porno, #porn, #palestineunderattack, #make, #linkbuilding, #lesbianporn, #ibmdesign, #gucci, #graffitiporn, #graffitiart, #gazaunderattack, #fendi, #digitalart, #design, #construction, #bestanalporn, #aulani, #art, #analporn, #amwriting, #abstractart}
- *Text-based cluster*: {#foodporn, # dinner, #food, #vsco, #vscocam}
- *Hybrid cluster*: {#foodporn, # dinner, #food, #vsco, #vscocam}

Our observation is that the metadata-based clustering picked on the second word "porn" and assigned it to a cluster that predominantly has to do with pornography whereas the text-based clustering assigned it to a cluster that has to do with food, and the hybrid clustering was in agreement with the text-based clustering and assigned it to the same cluster as the text-based clustering. Among the hashtags in the text-based cluster, the relevance of the lexical semantics of #vsco and #vscocam to food is not clear, but their associated tweet message texts, listed below, clearly show the relevance.

- `Pixin en salsa de oricios ?? #dinner #friday #food #foodporn #vscocam #vsco #tagsforlikes #tbl? http://t.co/WfPLisEoyZ`
- `#yummy ?? | ? (__) #letommys #milkshake #banana #choco #diner #american #angers #vscocam #vintage ? http://t.co/uahzzde0nr`

Evidently, the hybrid clustering borrowed the power of the text-based clustering to figure out their relevance to food. Our reasoning is that the hybrid clustering tapped on the co-occurrences of #vsco and #foodporn and of #vscocam and #foodporn to correctly disregard the hashtags about pornography that were dominant in the metadata-based cluster.

Anecdote 2. In this example, we consider hashtags related to "disease". Shown below are the clusters (from the three clustering algorithms) that were the best match to the ground truth cluster (i.e., cluster 1 in GT-S1) that had the theme of disease (see Fig. 4).

- *Metadata-based cluster*: {#braincancer, #braininjury, #braintumor, #braintumors, #breastcancer, #cancer, #depression, #diabetes, #dylexia, #dysthymia, #herediatarycancer, #itmakessenseifyouhavediabetes, #lungcancer, #majordepression, #menscancer, #overiancancer, #pancreaticcancer, #pancreaticcyst,

#pancreaticcysts, #penilecancer, #prostatecancer,#raredisease, #saydyslexia, #testicularcancer, #type1diabetes}

- *Text-based cluster*: {#penilecancer,#testicularcancer, #prostatecancer}
- *Hybrid cluster*: {#braincancer, #braintumor,#braintumors, #breastcancer, #cancer, #depression, #diabetes, #dylexia, #dysthymia, #herediatarycancer, #itmakessenseifyouhavediabetes, #lungcancer, #majordepression, #menscancer, #overiancancer, #pancreaticcancer, #pancreaticcyst, #pancreaticcysts, #penilecancer, #prostatecancer, #raredisease, #saydyslexia, #testicularcancer, #type1diabetes}

We see that the metadata-based cluster has a richer set of hashtags that are related to disease and the text-based cluster has only three and that the hybrid cluster is almost the same as the metadata-based cluster (except one hashtag #braininjury). This comparison clearly shows that the hybrid clustering used the power of the metadata-based clustering to create a cluster of higher quality.

Anecdote 3. This example is with regard to a pair of hashtags #sushi and #dinner. These two hashtags are obviously semantically close to each other, but their associated tweets do not appear to have much in common in terms of the bag-of-words model. From the Random tweet dataset, we found that #sushi has only one tweet (below) associated with it.

```
#breakfast #awake How to #make #sushi | how
to make sushi rice | sushi rice recipe: Brand
new high quality sus... http://t.co/WfPLisEoyZ
```

In addition, #dinner also has only one tweet.

```
Pixin en salsa de oricios ?? #dinner #friday
#food #foodporn #vscocam #vsco #tagsforlikes
#tbl?
```

These two hashtags were clustered together in the metadata-based clustering but separately in the text-based clustering. Then, the hybrid clustering put them in separate clusters. Evidently, the reason is that it did not see any common words between the two tweet texts and, therefore, the text-based clustering influenced the hybrid clustering to keep the two hashtags separate. Our manual vetting confirms that both text messages are relevant to food and, therefore, we conclude that the hybrid clustering should have not separated the two hashtags.

6. Conclusion

6.1. Summary

This paper addressed the problem of semantic hashtag clustering using Twitter hashtags as an example. We identified two major approaches – metadata-based and text-based – and categorized the semantics of hashtags into *lexical*, acquired from dictionaries, and *contextual*, acquired from tweet texts accompanying hashtags. Then, we presented a hybrid approach to semantic hashtag clustering, which uses the two approaches together.

The hybrid clustering leverages the complementary strengths to overcome the weaknesses of these two approaches. A consensus clustering scheme was used to build a hybrid approach combining the two approaches as the bases. The metadata-based clustering algorithm was our own sense-level semantic clustering algorithm, and the text-based clustering algorithm was the scalable multi-stage clustering algorithm by Tsur et al. The consensus scheme was meta-clustering, which builds a consensus graph and performs clustering using the graph.

We evaluated the hybrid clustering algorithm using a gold standard test and a pairwise disagreement test, and presented anecdotal examples showcasing the hybrid clustering. For the gold standard test, seven different ground truth (GT) cluster sets were constructed. The test results confirmed that the hybrid algorithm outperformed both of the two base algorithms against a majority of GT cluster sets. Moreover, the hybrid never underperformed *both* base algorithms against any GT cluster set, thus demonstrating its versatility of drawing strength from the two base algorithms. In the pairwise disagreement test, we focused on the instances of disagreement in clustering decision between the hybrid and the base algorithms. In aggregate (i.e., weighted average), the hybrid's clustering decision was right overall more than 90% of the time.

6.2. Future work

The future work can be pursued on two fronts – (a) improving the current hybrid clustering algorithm and (b) validating its impact on existing applications that historically used either of the base algorithms.

Regarding the improvement of the hybrid algorithm, we suggest three different aspects – the metadata sources, the semantic treatment of texts accompanying a hashtag, and the consensus scheme.

First, new metadata sources can enhance the metadata-based semantic hashtag clustering algorithm. For example, online translation services like Google Translate (https://translate.google.com) can be a good source since empirical evidences suggest that it can be very effective in identifying spelling errors, abbreviations, etc. (as well as translating hashtags of a different language). Additionally, crowdsourced websites like Urban Dictionary (www.urbandictionary.com) that specializes in informal human communication can be a helpful metadata source for identifying lexical semantics of a hashtag. Internet search engines also provide rich information on the semantics of hashtags.

Second, the text-based semantic hashtag clustering algorithm, which currently relies on classic document comparison methods, can also benefit from the same metadata sources used for identifying the *lexical* semantics of a hashtag. This is specially critical to uncommon hashtags, i.e, hashtags that have only one or two tweets associated with them. By extracting the main nouns present in a tweet and using metadata sources, we can semantically ground uncommon hashtags using the context to determine the topic of the hashtag (if the hashtag itself fails to be semantically grounded).

Third, since the hybrid clustering algorithm uses a consensus clustering approach, it may benefit from additional clustering algorithms. One candidate is a clustering algorithm based on the *co-occurrence relationship* between hashtags by using an association rule mining algorithm. Another candidate is a clustering algorithm based on the *temporal semantics* of hashtags. With either or both of these algorithms added, a better consensus may results from a multi-party decision. This addition may make the matters complicated, however. In the consensus clustering approach we used, individual base clustering algorithms can process different subsets of hashtags, and the consensus scheme is applied to an intersection of the subsets. So, as more base algorithms are added, taking an intersection of all these subsets may result in losing a significant portion of the original set of hashtags.

Regarding the validation with the existing applications, many of the related work used hashtag/tweet clustering only as a step toward more complex tasks, but they used either the metadata-based or the text-based approach. It will be interesting to see how the performances of those related work improve when our hybrid approach is used instead of what they used.

Table A.11
Hybrid clustering using word level semantic clustering vs sense level semantic clustering.

Ground truth clusters		Word-level hybrid clusters				Sense-level hybrid clusters			
Id	Size	Recall	Prec	f^m-score	Size	Recall	Prec	f^m-score	Size
1	47	0.89	0.67	0.76	63	0.89	0.67	0.76	63
2	29	0.07	1.00	0.13	2	0.55	0.94	0.70	17
3	23	0.26	0.19	0.22	32	0.61	0.88	0.72	16
4	18	0.22	0.50	0.31	8	0.22	0.57	0.32	7
5	14	0.21	1.00	0.35	3	0.36	0.50	0.42	10
6	12	0.42	0.83	0.56	6	0.33	0.67	0.44	6
7	11	0.09	1.00	0.17	1	0.36	0.50	0.42	8
8	11	0.18	0.20	0.19	10	0.27	0.50	0.35	6
9	11	0.27	1.00	0.43	3	0.27	1.00	0.43	3
10	10	0.20	0.22	0.21	9	0.60	0.35	0.44	17
11	10	1.00	1.00	1.00	10	1.00	0.71	0.83	14
12	7	0.29	1.00	0.44	2	0.00	0.00	0.00	0
13	7	0.14	1.00	0.25	1	0.00	0.00	0.00	0
14	7	0.71	0.28	0.40	18	0.71	1.00	0.83	5
15	6	0.67	0.31	0.42	13	0.67	0.31	0.42	13
16	6	0.33	0.50	0.40	4	0.33	0.29	0.31	7
17	5	0.80	0.50	0.62	8	0.80	0.50	0.62	8
18	4	0.25	1.00	0.40	1	0.25	0.33	0.29	3

Acknowledgments

The authors thank Ahmed Hamed, previously at the University of Vermont, for providing the random tweet datasets. This project was supported by a Fulbright Program grant sponsored by the Bureau of Educational and Cultural Affairs of the United States Department of State and administered by the Institute of International Education. The authors thank the anonymous reviewers for their comments, which were invaluable to improve the quality of the original manuscript.

Appendix A. Sense-level versus word-level hybrid clustering

The hybrid clustering algorithm uses the *sense*-level metadata-based clustering as a base algorithm. The merit of using the sense-level (as opposed to the word-level) has been demonstrated in our previous publications [12,13], but its effect on the hybrid clustering algorithm has not. Thus, in this section, we examine how the performance advantage gained through the *sense-level* semantic clustering translates into the hybrid algorithm performance.

We use only the combined ground truth dataset (GT-All) (see Section 5.3.2) for this experiment since it includes all the smaller ground truth datasets and, therefore, possesses a mixture of diverse hashtags encompassing both the Random and Symplur tweet datasets.

As explained earlier (see Section 5.3.2), we calculate the maximum f-score by finding a one-to-one best match based on f-scores between the output clusters and the ground truth clusters (18 of them in GT-All). Fig. A.6 shows the maximum f-scores of individual clusters generated by the hybrid algorithm when using the sense-level versus word-level metadata-based semantic clustering algorithm. (We call them "hybrid-sense" algorithm and "hybrid-word" algorithm to make the distinction clear.) There was no best match to the clusters 12 and 13 for hybrid-sense algorithm. Table A.11 shows the detailed results, including precision, recall, and cluster size. Weighted average f-score is 0.55 for hybrid-sense, which is 30% higher than 0.42 for hybrid-word. This is about the same improvement (i.e., 26%) the sense-level achieved for the metadata-based clustering alone [12,13], and so it indicates that the benefit of sense-level clustering transpires into the hybrid clustering with no decrease (with a slight increase in fact).

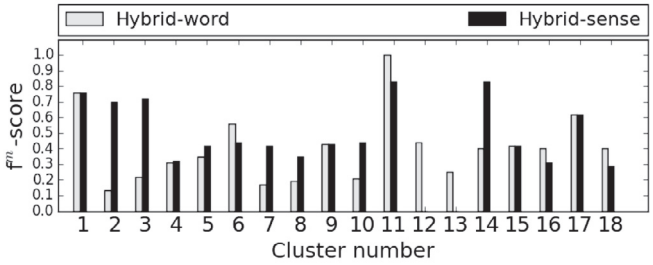


Fig. A.6. Hybrid clustering with sense-level versus word-level metadata-based semantic clustering. (Weighted average of pairwise maximum f-scores, i.e., f^a-score, is 0.42 for hybrid-word and 0.55 for hybrid-sense.)

References

[1] Merriam Webster, Simple defintion of hashtag. [cited 2016-05-01]. URL http://www.merriam-webster.com/dictionary/hashtag.
[2] #originstory. [cited 2016-05-13]. URL http://www.cmu.edu/homepage/computing/2014/summer/originstory.shtml.
[3] X. Wu, V. Kumar, The Top Ten Algorithms in Data Mining, 1st edition, Chapman & Hall/CRC, 2009.
[4] A. Kyriakopoulou, T. Kalamboukis, Text classification using clustering, Proceedings of the ECML-PKDD Discovery Challenge Workshop, 2006, pp. 28–38.
[5] C. Vicient, A. Moreno, Unsupervised semantic clustering of Twitter hashtags, Proceedings of the 21st European Conference on Artificial Intelligence, 2014, pp. 1119–1120.
[6] Usage statistics. [cited 2016-05-20]. URL http://www.internetlivestats.com/twitter-statistics/.
[7] C.I. Muntean, G.A. Morar, D. Moldovan, Exploring the meaning behind Twitter hashtags through clustering, Lect. Notes Bus. Inf. Process. 127 (2012) 231–242.
[8] S. Park, H. Shin, Identification of implicit topics in Twitter data not containing explicit search queries, Proceedings of the 25th International Conference on Computational Linguistics, 2014, pp. 58–68.
[9] X. Wang, F. Wei, X. Liu, M. Zhou, M. Zghan, Topic sentiment analysis in Twitter: a graph-based hashtag sentiment classification approach, Proceedings of the 20th ACM Conference on Information and Knowledge Management, 2011, pp. 1031–1040.
[10] S. Bhulai, P. Kampstra, L. Kooiman, G. Koole, M. Deurloo, B.K. CCing, Trend visualization on Twitter: what's hot and what's not?, Proceedings of the 1st International Conference on Data Analytics, Springer-Verlag, 2012, pp. 43–48.
[11] P. Teufl, S. Kraxberger, Extracting semantic knowledge from Twitter, Proceedings of the 3rd IFIP WG 8.5 International Conference on Electronic Participation, Springer-Verlag, 2011, pp. 48–59.
[12] A. Javed, B.S. Lee, Sense-level semantic clustering of hashtags in social media, in: Proceedings of the 3rd Annual International Symposium on Information Management and Big Data, 2016, pp. 140–149.
[13] A. Javed, B.S. Lee, Sense-level semantic clustering of hashtags, Communications in Computer and Information Science 656 (2017) 1–16.

[14] J. Costa, C. Silva, M. Antunes, B. Ribeiro, Defining semantic meta-hashtags for Twitter classification, Lect. Notes Comput. Sci. 7824 (2013) 226–235.
[15] O. Tsur, A. Littman, A. Rappoport, Scalable multi stage clustering of tagged micro-messages, Proceedings of the 21st International Conference on World Wide Web, 2012, pp. 621–622.
[16] O. Tsur, A. Littman, A. Rappoport, Efficient clustering of short messages into general domains, in: Proceedings of the 7th International AAAI Conference on Weblogs and Social Media, 2013, pp. 621–630.
[17] K.D. Rosa, R. Shah, B. Lin, Topical clustering of tweets, Proceedings of the 3rd Workshop on Social Web Search and Mining, 2011, pp. 133–138.
[18] H. Saif, Y. He, H. Alani, Semantic sentiment analysis of Twitter, Proceedings of the 11th International Conference on The Semantic Web - Volume Part I, Springer-Verlag, Berlin, Heidelberg, 2012, pp. 508–524.
[19] C.D. Manning, P. Raghavan, H. Schütze, Introduction to Information Retrieval, Cambridge University Press, New York, NY, USA, 2008.
[20] H. Ghafarzadeh, A. Bouyer, An efficient hybrid clustering method using an artificial bee colony algorithm and mantegna lévy distribution, Int. J. Artif. Intell. Tools 25 (2) (2016) 1–17.
[21] W.-C. Yeh, C.-M. Lai, K.-H. Chang, A novel hybrid clustering approach based on k-harmonic means using robust design, Neurocomputing 173 (2016).
[22] W. Dai, C. Yu, Z. Jiang, An improved hybrid canopy-fuzzy c-means clustering algorithm based on MapReduce model, J. Comput. Sci. Eng. 10 (1) (2016).
[23] Y. Yang, J. Jiang, Hybrid sampling-based clustering ensemble with global and local constitutions, IEEE Trans. Neural Netw. Learn. Syst. 27 (5) (2016) 952–965.
[24] Princeton University, WordNet: a lexical database for English. [cited 2016-05-08]. URL http://wordnet.princeton.edu.
[25] Z. Wu, M. Palmer, Verbs semantics and lexical selection, Proceedings of the 32nd Annual Meeting on Association for Computational Linguistics, 1994, pp. 133–138.
[26] Wikipedia, Help:Category. [cited 2016-05-01]. URL https://en.wikipedia.org/wiki/Help:Category.
[27] W. Feng, C. Zhang, W. Zhang, J. Han, J. Wang, C. Aggarwal, J. Huang, STREAMCUBE: hierarchical spatio-temporal hashtag clustering for event exploration over the twitter stream, Proceedings of the IEEE 31st International Conference on Data Engineering, 2015, pp. 1561–1572.
[28] I.S. Dhillon, Y. Guan, J. Fan, Efficient Clustering of Very Large Document Collections, Kluwer Academic Publishers, 2001.
[29] G. Stilo, P. Velardi, Temporal semantics: time-varying hashtag sense clustering, Proceedings of the 19th International Conference on Knowledge Engineering and Knowledge Management, Springer, 2014, pp. 563–578.
[30] G. Stilo, P. Velardi, Hashtag sense clustering based on temporal similarity, Comput. Linguist. 43 (1) (2017) 181–200.
[31] C.D. Manning, P. Raghavan, H. Schütze, Introduction to Information Retrieval, Cambridge University Press, New York, NY, USA, 2008.
[32] Flat cluster. [cited 2016-05-20]. URL http://docs.scipy.org/doc/scipy-0.14.0/reference/generated/scipy.cluster.hierarchy.fcluster.html.
[33] Scipy. [cited 8-02-17]. URL https://docs.scipy.org/doc/scipy-0.18.1/reference/generated/scipy.cluster.hierarchy.linkage.html#scipy.cluster.hierarchy.linkage.
[34] Symplur. [cited 2016-05-12]. URL http://www.symplur.com.
[35] P. Xanthopoulos, A review on consensus clustering methods, in: T.M. Rassias, C.A. Floudas, S. Butenko (Eds.), Optimization in Science and Engineering, Springer New York, 2014, pp. 553–566.
[36] S. Haghtalab, P. Xanthopoulos, K. Madani, A robust unsupervised consensus control chart pattern recognition framework, Expert Syst. Appl. 42 (19) (2015) 6767–6776.
[37] The healthcare hashtag project. [cited 2016-05-12]. URL http://www.symplur.com/healthcare-hashtags/.

Ali Javed is a Fulbright alumnus and a Ph.D. student in Computer Science at the University of Vermont, USA. He received MS in Computer Science from the University of Vermont, USA and BS in Computer Science from the National University of Computer and Emerging Sciences, Pakistan. His research interests include data mining, machine learning, and information retrieval.

Byung Suk Lee is a Professor of Computer Science at the University of Vermont, USA. He received BS in Electronics Engineering from Seoul National University, South Korea, MS in Electrical Engineering from KAIST, South Korea, and Ph.D. in Electrical Engineering (Computer Science) from Stanford University, USA. His research interests include database, data mining, data stream with a focus on modeling, querying, and performance.