Ontology- and Sentiment-aware Review Summarization

Nhat X.T. Le*, Vagelis Hristidis† and Neal Young‡
Computer Science and Engineering, University of California, Riverside
*nle020@ucr.edu, †vagelis@cs.ucr.edu, ‡neal.young@ucr.edu

***Abstract*—In this Web 2.0 era, there is an ever increasing number of product or service reviews, which must be summarized to help consumers effortlessly make informed decisions. Previous work on reviews summarization has simplified the problem by assuming that features (e.g., "display") are independent of each other and that the opinion for each feature in a review is Boolean: positive or negative. However, in reality features may be interrelated – e.g., "display" and "display color" – and the sentiment takes values in a continuous range – e.g., somewhat vs very positive.**

We present a novel review summarization framework that advances the state-of-the-art by leveraging a domain hierarchy of concepts to handle the semantic overlap among the features, and by accounting for different sentiment levels. We show that the problem is NP-hard and present bounded approximate algorithms to compute the most representative set of sentences, based on a principled opinion coverage framework. We experimentally evaluate the quality of the summaries using both intuitive coverage measure and a user study.

Keywords*—*Review summarization, sentiment analysis.

I. Introduction

Online users are increasingly relying on user reviews to make decisions on shopping (e.g., Amazon, Newegg), finding venues (e.g., Yelp, Foursquare), seeking doctors (e.g., Vitals.com, zocdoc.com) and many others. However, as the number of reviews per item grows, especially for popular products, it is infeasible for customers to read all of them, and discern the useful information from them. Therefore, many methods have been proposed to summarize customer opinions from the reviews [1]–[4]. They generally either adapt multi-document summarization techniques to choose important text segments [4], or they extract product concepts (also referred as features or attributes in other works), such as "display" of a phone, and customer's opinion (positive or negative) and aggregate them [1]–[3].

However, neither of these approaches takes into account the relationship among product's concepts. For example, assuming that we need the opinion summary of a smartphone, showing that the opinions for both *display* and *display color* are very positive is redundant, especially given that we would have to hide other concepts' opinion (e.g., "battery"), given the limited summary size. What makes the problem more challenging is that the opinion of a user for a concept is not Boolean (positive or negative) but can take values from a linear scale, e.g., "very positive", "positive", "somewhat positive", "neutral", and so on. Hence, if "display" has a positive opinion, but "display color" has neutral, the one does not subsume the other, and both should be part of the summary. Further, a more general concept may cover a more specific but not vice versa.

In Section IV we prove that the problem of selecting the best concepts and opinions to display such all opinions are covered as much as possible is NP-hard even when the relationships among the concepts are represented by a Directed-Acyclic-Graph (DAG). Therefore we proposed bounded approximation algorithms to solve it.

We experimentally evaluated our method on real collections of online patient reviews about doctors. We chose this dataset because rich concept hierarchies exist in the medical domain [5], and effective tools exist to extract medical concepts from free text [6], [7].

To summarize, the review summarization framework consists of the following tasks:

1) *Concept Extraction*: extract interesting medical concepts from reviews.
2) *Concept's Sentiment Estimation*: model the context around the extracted concepts and estimate its sentiment (opinion polarity on a linear scale).
3) *Select k representatives*: depending on the application, a representative can be a concept-sentiment pair (e.g., "display"=0.3) or a sentence from a review (e.g., "this phone has pretty sharp display"). Our proposed selection algorithms can be used to select representatives at any of these granularities.

II. Problem Definitions

Define an item (for example, a doctor) d as a set of reviews, where each review is a set of *concept-sentiment* pairs $\{(c_1, s_1), (c_2, s_2), \ldots, (c_n, s_n)\}$, and $s_j \in \mathbb{R}$ is the sentiment toward concept c_j in a review. Section III shows how the concept-sentiment pairs are extracted from the text of the reviews. The set of concepts (the *ontology*) depends on the application. We assume concepts are hierarchical, which is common in many domains (ConceptNet for example defines a general purpose concept hierarchy [8]). For the health-related content in our experiments, SNOMED CT [9] is a typical ontology with such a hierarchy (Figure 1).

Define the (directed) *distance* $d(p_1, p_2)$ between two concept-sentiment pairs $p_1 = (c_1, s_1)$ and $p_2 = (c_2, s_2)$, based on the concepts' relationship in the hierarchy, as follows.

Definition 1. First, define the distance between two concepts $d(c_1, c_2)$ to be the shortest-path length from c_1 to c_2 in the hierarchy. Let r be the root of the hierarchy. Let $\epsilon > 0$ be a pre-defined (sentiment) threshold. The distance $d(p_1, p_2)$ is:

$$d(p_1, p_2) = \begin{cases} d(r, c_2) & \text{if } c_1 \text{ is the root } r, \text{ or} \\ d(c_1, c_2) & \text{if } c_1 \text{ is the ancestor of } c_2 \\ & \text{ and } |s_1 - s_2| \leq \epsilon, \text{ or} \\ \infty & \text{otherwise} \end{cases}$$

If pair p_1 has finite distance to p_2, say p_1 *covers* p_2. Pair p_1 covers p_2 iff p_1's concept c_1 is an ancestor of p_2's concept c_2, and either c_1 is the root concept or the sentiments of p_1 and

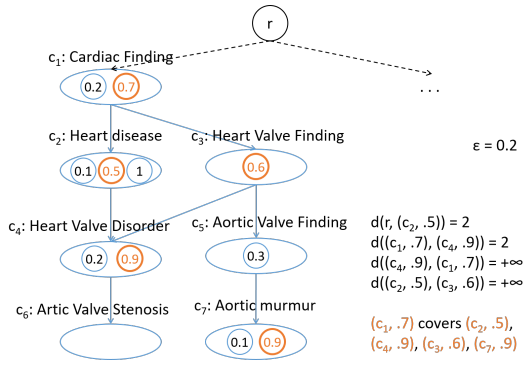


Fig. 1. Representation of concept-sentiment pairs on the concept hierarchy DAG that is a part of SNOMED CT

p_2 differ by at most ϵ. Figure 1 shows an example of how the concept-sentiment pairs of an item's reviews are mapped on the concept hierarchy, where the dashed line is the path from the root, and concept c_6 doesn't have any pairs. For instance, pair $(c_1, 0.7)$ represents an occurrence of concept c_1 in a review with sentiment 0.7. The same pair is also represented by the circled 0.7 value inside the c_1 tree node.

Given a set $P = \{p_1, p_2, \ldots, p_q\}$ of concept-sentiment pairs for the reviews of an item, and an integer k, our goal is to compute a set $F = \{f_1, f_2, \ldots, f_k\} \subseteq P$ of k pairs that best summarize P. To measure the quality of such a summary F, we define its cost $C(F, P)$ as the distance from F to P, defined as follows.

Definition 2. Define the distance from F to a pair p to be the distance of the closest pair in $F \bigcup \{r\}$ to p: $d(F, p) = \min_{f \in F \bigcup \{r\}} d(f, p)$. Define the cost of F to be the sum of its distances to pairs in P: $C(F, P) = \sum_{p \in P} d(F, p)$.

We introduce two summarization problems:
***k*-Pairs Coverage:** given a set P of concept-sentiment pairs (coming from a given set of reviews for an item) and integer $k \leq |P|$, find a subset $F \subseteq P$ with $|F| = k$ that summarizes P with minimum cost:

$$\min_{F \subseteq P, |F|=k} C(F, P)$$

***k*-Sentences Coverage:** given a set R of sentences and integer $k \leq |R|$, find a subset $X \subseteq R$ with $|X| = k$ that summarizes R with minimum cost:

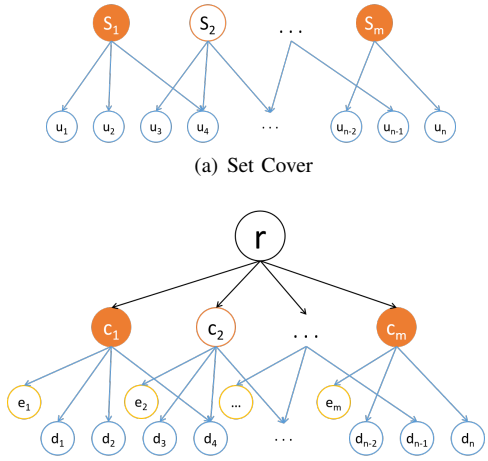
$$\min_{X \subseteq R, |X|=k} C(P(X), P(R)),$$

where $P(R)$ is the set of concept-sentiment pairs derived from the set R of sentences, and $P(X)$ is the set of concept-sentiment pairs derived from the subset X of R.

Intuitively, the first problem is appropriate when the summaries consist of concise concept-sentiment pairs, e.g. "good Heart Disease management", extracted from the reviews, and may be more suitable for mobile phone-sized screens. The second problem is appropriate if the summaries consist of whole sentences of reviews, which better preserve the meaning of the review, but may require more space to display.

III. Reviews Transformation

In this section we describe how we process reviews to transform them into the (concept, sentiment) pairs required by the problem definitions in Section II. We focus on doctor reviews to make the discussion and solutions concrete. First, we extract "interesting" medical concepts by adapting MetaMap [6] to only consider concepts in a set of semantic types that were manually selected. Then we compute the sentiment around that concept in the review. We implemented two methods for estimating the sentiment of a containing sentence: (a) using a sentiment dictionary [10] and (b) employing learning method based on vector representation of sentences [11]. Comparing these two methods with a user study, we decided to use the vector representation-based method with Ridge regression.



(a) Set Cover

(b) Corresponding instance of k-Pairs Coverage

Fig. 2. Reduction from Set Cover

IV. Both Problems are NP-Hard

Theorem 1. *The k-Pairs Coverage problem is NP-hard.*

Proof: The decision problem is, given a set P of concept-sentiment pairs, an integer $k \leq |P|$, and a target $t \geq 0$, to determine whether there exists a subset $F \subseteq P$ of size k with cost $C(F, P)$ at most t. We reduce Set Cover to it. Fix any Set-Cover instance (S, U, k) where U is the universe $\{u_1, u_2, \ldots, u_n\}$, and $S = \{S_1, S_2, \ldots, S_m\}$ is a collection of subsets of U, and $k \leq |S|$. Given (S, U, k), first construct a concept-hierarchy (DAG) with root r, concepts c_i and e_i for each subset S_i, and a concept d_j for each element u_j. For each set S_i, make c_i a child of r and e_i a child of c_i. For each element u_j, make d_j a child of c_i for each set S_i containing u_j. (See Fig. 2.) Next, construct $2m + n$ concept-sentiment pairs $P = \{p_1, \ldots, p_{2m+n}\}$, one containing each node in the DAG other than the root r, and all with the same sentiment, say 0. Take target $t = 3m + n - 2k$. This completes the reduction. It is clearly polynomial time. Next we verify that it is correct. For brevity, identify each pair with its node.

Suppose S has a set cover of size k. For the summary $F \subseteq P$ of size k, take the k concepts in P that correspond to the sets in the cover. Then each d_i has distance 1 to F, contributing n to the cost. For each set in the cover, the corresponding c_i and e_i have distance 0 and 1 to F, contributing k to the cost. For each set not in the cover, the corresponding c_i and e_i have distance 1 and 2 to F, contributing $3(m - k)$ to the cost, for a total cost of $n + 3m - 2k = t$.

Conversely, suppose P has a summary of size k and cost $t = n+3m-2k$. Among size-k summaries of cost at most t, let

F be one with a maximum number of c_i nodes. We show that the sets corresponding to the (at most k) c_i nodes in F form a set cover. Assume some $c_{i'}$ is missing from F (otherwise $k \geq m$ so we are done). For every e_i in F, its parent c_i is also in F. (Otherwise adding c_i to F and removing e_i would give a better summary F', i.e., a size-k summary of cost at most t, but with more c_i nodes than F, contradicting the choice of F). No e_i is in F (otherwise removing e_i and adding the missing node $c_{i'}$ would give a better summary F'). No d_j is in F (otherwise, since neither $e_{i'}$ nor $c_{i'}$ are in F, removing d_j from F and adding $c_{i'}$ would give a better summary F'). Since no e_i or d_j is in F, only c_i nodes are in F. Since the cost is at most $t = n + 3m - 2k$, by calculation as in the preceding paragraph, the sets S_i corresponding to the nodes c_i in F must form a set cover. ■

V. Algorithms

In this manuscript we implement a greedy bounded approximation algorithm. We first describe the initialization phase before going into the algorithm's details.

A. Initialization

The transformation in Section III gives a set P of concept-sentiment pairs. The initialization phase computes the underlying edge-weighted bipartite graph $G = (U, W, E)$ where vertex sets U and W are the concept-sentiment pairs in the given set P, edge set E is $\{(p, p') \in U \times W : d(p, p') < \infty\}$, and edge (p, p') has weight equal to the pair distance $d(p, p')$. The initialization phase builds G in two passes over P. The first pass puts the pairs $p = (c, s)$ into buckets by category c. The second pass, for each pair $p = (c, s)$, iterates over the ancestors of c in the DAG (using depth-first-search from c). For each ancestor c', it checks the pairs $p' = (c', s')$ in the bucket for c'. For those with finite distance $d(p, p')$, it adds the corresponding edge to G.

For our problems, the time for the initialization phase and the size of the resulting graph G are roughly linear in $|P|$ because the average number of ancestors for each node in the DAG is small.

B. Greedy algorithm

The greedy algorithm is Algorithm 1. It starts with a set $F = \{r\}$ containing just the root. It then iterates k times, in each iteration adding a pair $p \in P$ to F chosen to minimize the resulting cost $C(F \cup \{p\}, P)$. Finally, it returns summary $F \setminus \{r\}$. This is essentially a standard greedy algorithm for k-medians. Since the cost is a submodular function of P, the algorithm is a special case of Wolsey's generalization of the greedy set-cover algorithm [12].

After the initialization phase, which computes the graph $G = (U, W, E)$, the algorithm further initializes a max-heap for selecting p in each iteration. The max-heap stores each pair p, keyed by $\delta(p, F) = C(F \cup \{p\}, P) - C(F, P)$. The max-heap is initialized naively, in time $O(m + n \log n)$ (where $m = |E|$, $n = |P|$). (This could be reduced to $O(m + n)$ with the linear-time build-heap operation.) Each iteration deletes the pair p with maximum key from the heap (in $O(\log n)$ time), adds p to F, and then updates the changed keys. The pairs q whose keys change are those that are neighbors of neighbors of p in G. The number of these updates is typically $O(d^2)$, where d is the typical degree of a node in G. The cost of each update is $O(\log n)$ time. After initialization, the algorithm typically takes $O(kd^2 \log n)$ time. In our experiments, our graphs are sparse (a typical node p has only hundreds of such pairs q), and k is a small constant, so the time after initialization is dominated by the time for initialization.

The following worst-case approximation guarantee is a direct corollary of Wolsey's analysis [12]. Let $H(i) = 1 + 1/2 + \cdots + 1/i \approx 1 + \log i$ be the ith harmonic number. Let Δ be the maximum depth of the concept DAG.

Theorem 2. *The greedy algorithm produces a size-k summary of cost at most $\text{OPT}_{k'}(P)$, where $k' = \lfloor k/H(\Delta n) \rfloor$.*

In our experiments, the algorithm returns near-optimal size-k summaries.

Algorithm 1 Greedy Algorithm

Input: $G = (U, W, E)$ from initialization, computed from P.
Output: Size-k summary F.

```
1:  procedure GREEDY
2:      Define δ(p, F) = C(F ∪ {p}, P) − C(F, P).
3:      Let F = {r}.
4:      Initialize max-heap holding p ∈ U keyed by δ(p, F).
5:      while |F| < k + 1 do
6:          Delete p with highest key from max-heap.
7:          Add p to F.
8:          for w such that (p, w) ∈ E do
9:              for q such that (q, w) ∈ E do
10:                 Update max-heap key δ(q, F) for q.
11:     return F \ {r}
```

C. Adaptation for k-Sentences Coverage problem

When sentences (each contains a set of concept-sentiment pairs) must be selected, the above algorithms can still be applied with only a modification of the initialization stage of Section V-A. In particular, we modify the construction of bipartite graph $G = (U, W, E)$, instead of having both U and W as concept-sentiment pairs in P, U represents the set of candidate sentences R, and W represents concept-sentiment pairs as before.

VI. Experimental Evaluation

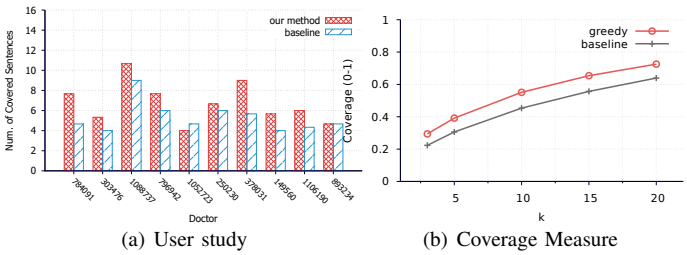
The goal of this section is to study the quality of the summarization achieved by the proposed algorithms and compare them to a state-of-the-art baseline.

Baseline summarization method The baseline method being used to select top k sentences in this evaluation is adapted from [1]. The algorithm in [1] was designed to summarize customer reviews of online shopping products. It first extracts product features (attributes like "picture quality" for product "digital camera"), then classifies review sentences that mention these features as positive or negative, and finally sums up the number of positive and negative sentences for each feature. To have a fair comparison, we adapt their method to select top k sentences. We first count the number of pair (concept, positive) or (concept, negative), for example: feature "picture quality" with sentiment "positive" occurs in 200 sentences. Then, we select k most popular pairs and return one containing sentence for each selected pair. Note that the task of extracting product features is equivalent to identifying interesting medical

TABLE I

A SAMPLE SURVEY OF A DOCTOR. WE ASK PARTICIPANTS TO FILL IN CELLS WITH "X" TO SAY THAT THE COLUMN SENTENCE CAN COVER THE ROW SENTENCE. THE MEDICAL CONCEPTS ARE UNDERLINED.

all sentences \ selected sentences	He has showed our family what kind of great physician he is, and I have no worries of his capabilities to take care of my neck.	My back pain has now come back, so I called Dr XX because I wanted to see him, and he will not even see me.	He has completely gave me back the mobility of my legs when he corrected the Lumbar.	I am now having my neck fixed by him on Tues.	He removed 9 disc in a single operation.
I got to say when the Lord gave me someone to fix my neck, he gave me a wonderful Dr with his staff.	X				
I had the pleasure of meeting, Dr XX, when I was referred about my neck.				X	



(a) User study (b) Coverage Measure

Fig. 3. Comparison with distance threshold 2, sentiment threshold 0.3

concepts in our case, so their first two tasks are executed by re-using our work described in Section III. From now on we refer to this adaptation as *baseline method*.

User Study During this user study, we ask users to indicate the semantic coverage between pairs of sentences found in doctor reviews. The user study involves three graduate students.

We ask each user to evaluate the sentences' coverage for the same 10 randomly selected doctors from our dataset, where for each doctor we select 3 random positive and 3 random negative reviews. These 6 reviews are broken down into sentences that are input to our Greedy algorithm (with sentiment threshold of 0.3) and the baseline method to select the top 3 sentences per method. We then take the union of the top-3 results of the two methods and ask the participants to judge if a selected sentence semantically covers another one from the full set of sentences. A typical task for a doctor is presented in Table I, in which the first row shows 5 sentences selected by our method (Greedy) and the baseline (1 overlapping sentence). The "*X*"s are filled by the participant to express that the selected sentence of that column covers that row's sentence. As mentioned above, each row corresponds to one sentence of one of the doctor's reviews, and the columns are the sentences selected by one of the two summarization methods. Note that the sentence of the first column does not cover the sentence of second row because the former has very high sentiment and the latter is closer to neutral. The same explains why the fourth column review does not cover the first row review.

The number of covered sentences of each method is averaged for all participants and is shown in Figure 3(a). The result shows that our method outperforms the baseline in 8 cases, and is equivalent or worse in only 1 case. The number of covered sentences of our is 64% higher for doctor 784091. We further observe that for doctor 1088737 the number of covered sentences is much higher. The reason is that this doctor has longer reviews and hence more sentences (20 in total), compared to doctor 1052723, which only has 10 sentences.

Coverage Measures Besides the user survey, we also compare our method with the baseline based on an intuitive coverage measure that is different than the one proposed in Section II, to avoid giving an unfair advantage to our method. Specifically, the measure is defined as the percentage of concept-sentiment pairs covered by that selected k sentences divided by the total number of pairs of a doctor. A sentence *covers* a pair p if the sentence contains at least one pair that covers p. A pair is said to cover another if their sentiment difference is less than or equal to a (sentiment) threshold and their dissimilarity is at most a (distant) threshold. In our experiment we use the path length between concepts of pairs in the ontology as dissimilarity measure. We evaluated this coverage measure for several distance and sentiment thresholds, but only show the results for distance threshold of 2, and sentiment threshold of 0.3 due to the space limitation, in Figure 3(b). Consistently in all cases, our method outperforms the baseline about 10 – 30% per case.

ACKNOWLEDGMENT

Partially supported by NSF grants IIS-1216007, IIS-1447826 and IIS-1619463.

REFERENCES

[1] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *10th ACM SIGKDD*, 2004, pp. 168–177.

[2] X. Ding, B. Liu, and P. S. Yu, "A holistic lexicon-based approach to opinion mining," in *WSDM 2008*. ACM.

[3] Y. Lu, C. Zhai, and N. Sundaresan, "Rated aspect summarization of short comments," in *WWW 2009*.

[4] G. Carenini, J. C. K. Cheung, and A. Pauls, "Multi-document summarization of evaluative text," *Computational Intelligence*, vol. 29, no. 4, pp. 545–576, 2013.

[5] O. Bodenreider, "The unified medical language system (umls): integrating biomedical terminology," *Nucleic acids research*, vol. 32, no. suppl 1, pp. D267–D270, 2004.

[6] A. R. Aronson, "Effective mapping of biomedical text to the umls metathesaurus: the metamap program." in *AMIA*, 2001.

[7] G. K. Savova, J. J. Masanz, P. V. Ogren, J. Zheng, S. Sohn, K. C. Kipper-Schuler, and C. G. Chute, "Mayo clinical text analysis and knowledge extraction system (ctakes): architecture, component evaluation and applications," *JAMIA*, 2010.

[8] H. Liu and P. Singh, "Conceptnet - a practical commonsense reasoning tool-kit," *BT technology journal*, 2004.

[9] "SNOMED CT," https://www.nlm.nih.gov/research/umls/Snomed/snomed_main.html, 2016.

[10] S. Baccianella, A. Esuli, and F. Sebastiani, "Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining." in *LREC*, 2010.

[11] Q. V. Le and T. Mikolov, "Distributed representations of sentences and documents." in *ICML*, 2014.

[12] L. A. Wolsey, "An analysis of the greedy algorithm for the submodular set covering problem," *Combinatorica*, vol. 2, no. 4, pp. 385–393, 1982.