Query-Focused Opinion Summarization for User-Generated Content

Lu Wang[1] **Hema Raghavan**[2] **Claire Cardie**[1] **Vittorio Castelli**[3]
[1]Department of Computer Science, Cornell University, Ithaca, NY 14853, USA
{luwang, cardie}@cs.cornell.edu
[2]LinkedIn, CA, USA
hraghavan@linkedin.com
[3]IBM T. J. Watson Research Center, Yorktown Heights, NY 10598, USA
vittorio@us.ibm.com

Abstract

We present a submodular function-based framework for query-focused opinion summarization. Within our framework, relevance ordering produced by a statistical ranker, and information coverage with respect to topic distribution and diverse viewpoints are both encoded as submodular functions. Dispersion functions are utilized to minimize the redundancy. We are the first to evaluate different metrics of text similarity for submodularity-based summarization methods. By experimenting on community QA and blog summarization, we show that our system outperforms state-of-the-art approaches in both automatic evaluation and human evaluation. A human evaluation task is conducted on Amazon Mechanical Turk with scale, and shows that our systems are able to generate summaries of high overall quality and information diversity.

1 Introduction

Social media forums, such as social networks, blogs, newsgroups, and community question answering (QA), offer avenues for people to express their opinions as well collect other people's thoughts on topics as diverse as health, politics and software (Liu et al., 2008). However, digesting the large amount of information in long threads on newsgroups, or even knowing which threads to pay attention to, can be overwhelming. A text-based summary that highlights the diversity of opinions on a given topic can lighten this information overload. In this work, we design a submodular function-based framework for opinion summarization on community question answering and blog data.

Question: What is the long term effect of piracy on the music and film industry?

Best Answer: Rising costs for movies and music. ... If they sell less, they need to raise the price to make up for what they lost. The other thing will be music and movies with less quality. ...
Other Answers:
Ans1: Its bad... really bad. (Just watch this movie and you will find out ... Piracy causes rappers to appear on your computer).
Ans2: By removing the profitability of music & film companies, piracy takes away their motivation to produce new music & movies. If they can't protect their copyrights, they can't continue to do business. ...
Ans4: *It is forcing them to rework their business model, which is a good thing.* In short, I don't think the music industry in particular will ever enjoy the huge profits of the 90's. ...
Ans6: Please-People in those businesses make millions of dollars as it is!! I don't think piracy hurts them at all!!!

Figure 1: Example discussion on Yahoo! Answers. Besides the best answer, other answers also contain relevant information (in *italics*). For example, the sentence in blue has a contrasting viewpoint compared to the other answers.

Opinion summarization has previously been applied to restricted domains, such as product reviews (Hu and Liu, 2004; Lerman et al., 2009; Yu et al., 2012) and news (Stoyanov and Cardie, 2006), where the output summary is either presented in a structured way with respect to each feature (or aspect) of the product or organized along contrastive viewpoints. Unlike those works, we address user generated online data: community QA and blog data. These forums use a substantially less formal language than news articles, and at the same time address a much broader spectrum of topics than product reviews. As a result, they present new challenges for automatic summarization. For example, Figure 1 illustrates a sample question from Yahoo! Answers[1] along with the answers from different users. The question receives more than one answer, and one of them is selected as the "best answer" by the asker or other participants. In general, answers from other users also provide relevant information. While community QA successfully pools rich knowledge from the wisdom of the crowd, users might need to seine through numerous posts to extract the information they need. Hence, it would be beneficial to summarize answers automatically and present the summaries to users who ask similar questions in the future. In this work,

[1]http://answers.yahoo.com/

we aim to return a summary that encapsulates different perspectives for a given opinion question and a set of relevant answers or documents.

In our work we assume that there is a central topic (or query) on which a user is seeking diverse opinions. We predict query-relevance through automatically learned statistical rankers. Our ranking function not only aims to find sentences that are on the topic of the query but also ones that are "opinionated" through the use of several features that indicate subjectivity and sentiment. The relevance score is encoded in a submodular function. Diversity is accounted for by a dispersion function that maximizes the pairwise distance between the pairs of sentences selected.

Our chief contributions are:

(1) We develop a submodular function-based framework for query-focused opinion summarization. To the best of our knowledge, this is the first time that submodular functions have been used to support opinion summarization. We test our framework on two tasks: summarizing opinionated sentences in community QA (Yahoo! Answers) and blogs (TAC-2008 corpus). Human evaluation using Amazon Mechanical Turk shows that our system generates the best summary 57.1% of the time. On the other hand, the best answer picked by Yahoo! users is chosen only 31.9% of the time. We also obtain significant higher Pyramid F1 score on the blog task as compared to the system of Lin and Bilmes (2011).

(2) Within our summarization framework, the statistically learned sentence relevance is included as part of our objective function, whereas previous work on submodular summarization (Lin and Bilmes, 2011) only uses ngram overlap for query relevance. Additionally, we use Latent Dirichlet Allocation (Blei et al., 2003) to model the topic structure of the sentences, and induce clusterings according to the learned topics. Therefore, our system is capable of generating summaries with broader topic coverage.

(3) Furthermore, we are the first to study how different metrics for computing text similarity or dissimilarity affect the quality of submodularity-based summarization methods. We show empirically that lexical representation-based similarity, such as TFIDF scores, uniformly outperforms semantic similarity computed with WordNet. Moreover, when measuring the summary diversity, topical representation is marginally better than lexical representation, and both of them beats semantic representation.

2 Related Work

Our work falls in the realm of query-focused summarization, where a user asks a question and the system generates a summary of the answers containing pertinent and diverse information. A wide range of methods have been investigated, where relevance is often estimated through TF-IDF similarity (Carbonell and Goldstein, 1998), topic signature words (Lin and Hovy, 2000) or by learning a Bayesian model over queries and documents (Daumé and Marcu, 2006). Most work only implicitly penalizes summary redundancy, e.g. by downweighting the importance of words that are already selected.

Encouraging diversity of a summary has recently been addressed through submodular functions, which have been applied for multi-document summarization in newswire (Lin and Bilmes, 2011; Sipos et al., 2012), and comments summarization (Dasgupta et al., 2013). However, these works either ignore the query information (when available) or else use simple ngram matching between the query and sentences. In contrast, we propose to optimize an objective function that addresses both relevance and diversity.

Previous work on generating opinion summaries mainly considers product reviews (Hu and Liu, 2004; Lerman et al., 2009; Yu et al., 2012), and formal texts such as news articles (Stoyanov and Cardie, 2006) or editorials (Paul et al., 2010). Mostly, there is no query information, and summaries are formulated in a structured way based on product features or contrastive standpoints. Our work is more related to opinion summarization on user-generated content, such as community QA. Liu et al. (2008) manually construct taxonomies for questions in community QA. Summaries are generated by clustering sentences according to their polarity based on a small dictionary. Tomasoni and Huang (2010) introduce coverage and quality constraints on the sentences, and utilize an integer linear programming framework to select sentences.

3 Submodular Opinion Summarization

In this section, we describe how query-focused opinion summarization can be addressed by submodular functions combined with dispersion functions. We first define our problem. Then we introduce the components of our objective function (Sections 3.1–3.3). The full objective function is presented in

Basic Features	Sentiment Features
- answer position in all answers/sentence position in blog - length of the answer/sentence - length is less than 5 words	- number/portion of sentiment words from a lexicon (Section 3.2) - if contains sentiment words with the same polarity as sentiment words in query
Query-Sentence Overlap Features	**Query-Independent Features**
- unigram/bigram TF/TFIDF similarity with query - number of key phrases in the query that appear in the sentence. A model similar to that described in (Luo et al., 2013) was applied to detect key phrases.	- unigram/bigram TFIDF similarity with cluster centroid - sumBasic score (Nenkova and Vanderwende, 2005) - number of topic signature words (Lin and Hovy, 2000) - JS divergence with cluster

Table 1: Features used for candidate ranking. We use them for ranking answers in both community QA and blogs.

Section 3.4. Lastly, we describe a greedy algorithm with constant factor approximation to the optimal solution for generating summaries (Section 3.5).

A set of documents or answers to be summarized are first split into a set of individual sentences $V = \{s_1, \cdots, s_n\}$. Our problem is to select a subset $S \subseteq V$ that maximizes a given objective function $f : 2^V \rightarrow \mathbb{R}$ within a length constraint: $S^* = \arg\max_{S \subseteq V} f(S)$, subject to $\mid S \mid \leq c$. $\mid S \mid$ is the length of the summary S, and c is the length limit.

Definition 1 *A function $f : 2^V \rightarrow \mathbb{R}$ is submodular iff for all $s \in V$ and every $S \subseteq S' \subseteq V$, it satisfies $f(S \cup \{s\}) - f(S) \geq f(S' \cup \{s\}) - f(S')$.*

Previous submodularity-based summarization work assumes this diminishing return property makes submodular functions a natural fit for summarization and achieves state-of-the-art results on various datasets. In this paper, we follow the same assumption and work with non-decreasing submodular functions. Nevertheless, they have limitations, one of which is that functions well suited to modeling diversity are not submodular. Recently, Dasgupta et al. (2013) proved that diversity can nonetheless be encoded in well-designed *dispersion functions* which still maintain a constant factor approximation when solved by a greedy algorithm.

Based on these considerations, we propose an objective function $f(S)$ mainly considering three aspects: *relevance* (Section 3.1), *coverage* (Section 3.2), and *non-redundancy* (Section 3.3). Relevance and coverage are encoded in a non-decreasing submodular function, and non-redundancy is enforced by maximizing the dispersion function.

3.1 Relevance Function

We first utilize statistical rankers to produce a preference ordering of the candidate answers or sentences. We choose ListNet (Cao et al., 2007), which has been shown to be effective in many information retrieval tasks, as our ranker. We use the implementation from Ranklib (Dang, 2011).

Features used in the ranking algorithm are summarized in Table 1. All features are normalized by standardization. Due to the length limit, we cannot provide the full results on feature evaluation. Nevertheless, we find that ranking candidates by TFIDF similarity or key phrases overlapping with the query can produce comparable results with using the full feature set (see Section 5).

We take the ranks output by the ranker, and define the relevance of the current summary S as: $r(S) = \sum_i^{|S|} \sqrt{rank_i^{-1}}$, where $rank_i$ is the rank of sentence s_i in V. For QA answer ranking, sentences from the same answer have the same ranking. The function $r(S)$ is our first submodular function.

3.2 Coverage Functions

Topic Coverage. This function is designed to capture the idea that a comprehensive opinion summary should provide thoughts on distinct aspects. Topic models such as Latent Dirichlet Allocation (LDA) (Blei et al., 2003) and its variants are able to discover hidden topics or aspects of document collections, and thus afford a natural way to cluster texts according to their topics. Recent work (Xie and Xing, 2013) shows the effectiveness of utilizing topic models for newsgroup document clustering. We first learn an LDA model from the data, and treat each topic as a cluster. We estimate a sentence-topic distribution $\vec{\theta}$ for each sentence, and assign the sentence to the cluster k corresponding to the mode of the distribution (i.e., $k = \arg\max_i \theta_i$). This naive approach produces comparable clustering performance to the state-of-the-art according to (Xie and Xing, 2013). $\mathcal{T}$ is defined as the clustering induced by our algorithm on the set V. The topic coverage of the current summary S is defined as $t(S) = \sum_{T \in \mathcal{T}} \sqrt{|S \cap T|}$.

From the concavity of the square root it follows that sets S with uniform coverages of topics are preferred to sets with skewed coverage.

Authorship Coverage. This term encourages the summarization algorithm to select sentences from different authors. Let $\mathcal{A}$ be the clustering induced by the sentence to author relation. In community QA, sentences from the answers given by the same user belong to the same cluster. Similarly, sentences from blogs with the same author are in the same cluster. The authorship score is defined as $a(S) = \sum_{A \in \mathcal{A}} \sqrt{|S \cap A|}$.

Polarity Coverage. The polarity score encourages the selection of summaries that cover both positive and negative opinions. We categorize each sentence simply by counting the number of polarized words given by our lexicon. A sentence belongs to a positive cluster if it has more positive words than negative ones, and vice versa. If any negator co-occurs with a sentiment word (e.g. within a window of size 5), the sentiment is reversed.[2] The polarity clustering $\mathcal{P}$ thus have two clusters corresponding to positive and negative opinions. The score is defined as $p(S) = \sum_{P \in \mathcal{P}} \sqrt{|\ S \cap P\ |}$. Our lexicon consists of MPQA lexicon (Wilson et al., 2005), General Inquirer (Stone et al., 1966), and SentiWordNet (Esuli and Sebastiani, 2006). Words with conflicting sentiments from different lexicons are removed.

Content Coverage. Similarly to Lin and Bilmes (2011) and Dasgupta et al. (2013), we use the following function to measure content coverage of the current summary S: $c(S) = \sum_{v \in V} \min(cov(v, S), \theta \cdot cov(v, V))$, where $cov(v, S) = \sum_{u \in S} sim(v, u)$. We experiment with two types of similarity functions. One is a Cosine TFIDF similarity score. The other is a WordNet-based semantic similarity score between pairwise dependency relations from two sentences (Dasgupta et al., 2013). Specifically, $sim_{Sem}(v, u) = \sum_{rel_i \in v, rel_j \in u} WN(a_i, a_j) \times WN(b_i, b_j)$, where $rel_i = (a_i, b_i), rel_j = (a_j, b_j)$, $WN(w_i, w_j)$ is the shortest path length. All scores are scaled onto $[0, 1]$.

3.3 Dispersion Function

Summaries should contain as little redundant information as possible. We achieve this by adding an additional term to the objective function, encoded by a dispersion function. Given a set of sentences S, a complete graph is constructed with each sentence in S as a node. The weight of each edge (u, v) is their dissimilarity $d'(u, v)$. Then the distance between any pair of u and v, $d(u, v)$, is defined as the total weight of the shortest path connecting u and v.[3] We experiment with two forms of dispersion function (Dasgupta et al., 2013): (1) $h_{sum} = \sum_{u,v \in V, u \neq v} d(u, v)$, and (2) $h_{min} = \min_{u,v \in V, u \neq v} d(u, v)$.

Then we need to define the dissimilarity function $d'(\cdot, \cdot)$. There are different ways to measure the dissimilarity between sentences (Mihalcea et al., 2006; Agirre et al., 2012). In this work, we experiment with three types of dissimilarity functions.

Lexical Dissimilarity. This function is based on the well-known Cosine similarity score using TFIDF weights. Let $sim_{tfidf}(u, v)$ be the Cosine similarity between u and v, then we have $d'_{Lex}(u, v) = 1 - sim_{tfidf}(u, v)$.

Semantic Dissimilarity. This function is based on the semantic meaning embedded in the dependency relations. $d'_{Sem}(u, v) = 1 - sim_{Sem}(v, u)$, where $sim_{Sem}(v, u)$ is the semantic similarity used in content coverage measurement in Section 3.2.

Topical Dissimilarity. We propose a novel dissimilarity measure based on topic models. Celikyilmaz et al. (2010) show that estimating the similarity between query and passages by using topic structures can help improve the retrieval performance. As discussed in the topic coverage in Section 3.2, each sentence is represented by its sentence-topic distributions estimated by LDA. For candidate sentence u and v, let their topic distributions be P_u and P_v. Then the dissimilarity between u and v can be defined as: $d'_{Topic}(u, v) = JSD(P_u || P_v) = \frac{1}{2}(\sum_i P_u(i) \log_2 \frac{P_u(i)}{P_a(i)} + \sum_i P_v(i) \log_2 \frac{P_v(i)}{P_a(i)})$ where $P_a(i) = \frac{1}{2}(P_u(i) + P_v(i))$.

3.4 Full Objective Function

The objective function takes the interpolation of the submodular functions and dispersion function:

$$\mathcal{F}(S) = r(S) + \alpha t(S) + \beta a(S) + \gamma p(S) + \eta c(S) + \delta h(S). \quad (1)$$

[2] There exists a large amount of work on determining the polarity of a sentence (Pang and Lee, 2008) which can be employed for polarity clustering in this work. We decide to focus on summarization, and estimate sentence polarity through sentiment word summation (Yu and Hatzivassiloglou, 2003), though we do not distinguish different sentiment words.

[3] This definition of distance is used to produce theoretical guarantees for the greedy algorithm described in Section 3.5.

The coefficients $\alpha, \beta, \gamma, \eta, \delta$ are non-negative real numbers and can be tuned on a development set.[4] Notice that each summand except $h(S)$ is a non-decreasing, non-negative, and submodular function, and summation preserves monotonicity, non-negativity, and submodularity. Dispersion function $h(s)$ is either h_{sum} or h_{min} as introduced previously.

3.5 Summary Generation via Greedy Algorithm

Generating the summary that maximizes our objective function in Equation 1 is NP-hard (Chandra and Halldórsson, 1996). We choose to use a greedy algorithm that guarantees to obtain a constant factor approximation to the optimal solution (Nemhauser et al., 1978; Dasgupta et al., 2013). Concretely, starting with an empty set, for each iteration, we add a new sentence so that the current summary achieves the maximum value of the objective function. In addition to the theoretical guarantee, existing work (McDonald, 2007) has empirically shown that classical greedy algorithms usually works near-optimally.

4 Experimental Setup

4.1 Opinion Question Identification

We first build a classifier to automatically detect opinion oriented questions in Community QA; questions in the blog dataset are all opinionated. Our opinion question classifier is trained on two opinion question datasets: (1) the first, from Li et al. (2008a), contains 646 opinionated and 332 objective questions; (2) the second dataset, from Amiri et al. (2013), consists of 317 implicit opinion questions, such as "*What can you do to help environment?*", and 317 objective questions. We train a RBF kernel based SVM classifier to identify opinion questions, which achieves F1 scores of 0.79 and 0.80 on the two datasets when evaluated using 10-fold cross-validation (the best F1 scores reported are 0.75 and 0.79).

4.2 Datasets

Community QA Summarization: Yahoo! Answers. We use the Yahoo! Answers dataset from Yahoo! *Webscope*^{TM} program,[5] which contains 3,895,407 questions. We first run the opinion question classifier to identify the opinion questions. For summarization purpose, we require each question having at least 5 answers, with the average length of answers larger than 20 words. This results in 130,609 questions.

To make a compelling task, we reserve questions with an average length of answers larger than 50 words as our test set for both ranking and summarization; all the other questions are used for training. As a result, we have 92,109 questions in the training set for learning the statistical ranker, and 38,500 in the test set. The category distribution of training and test questions (Yahoo! Answers organizes the questions into predefined categories) are similar. 10,000 questions from the training set are further reserved as the development set. Each question in the Yahoo! Answers dataset has a user-voted best answer. These best answers are used to train the statistical ranker that predicts relevance. Separate topic models are learned for each category, where the category tag is provided by Yahoo! Answer.

Blog Summarization: TAC 2008. We use the TAC 2008 corpus (Dang, 2008), which consists of 25 topics. 23 of them are provided with human labeled nuggets, which TAC used in human evaluation. TAC also provides snippets (i.e., sentences) that are frequently retrieved by participant systems or identified as relevant by human annotators. We do not assume those snippets are known to any of our systems.

4.3 Comparisons

For both opinion summarization tasks, we compare with (1) the approach by Dasgupta et al. (2013), and (2) the systems from Lin and Bilmes (2011) with and without query information. The sentence clustering process in Lin and Bilmes (2011) is done by using CLUTO (Karypis, 2003). For the implementation of systems in Lin and Bilmes (2011) and Dasgupta et al. (2013), we always use the parameters reported to have the best performance in their work.

For cQA summarization, we use the **best answer** voted by the user as a baseline. Note that this is a strong baseline since all the other systems are unaware of which answer is the best. For blog summarization, we have three additional baselines – the **best systems** in TAC 2008 (Kim et al., 2008; Li et al., 2008b), top sentences returned by our **ranker**, a baseline produced by TFIDF similarity and a lexicon

[4] The values for the coefficients are $5.0, 1.0, 10.0, 5.0, 10.0$ for $\alpha, \beta, \gamma, \eta, \delta$, respectively, as tuned on the development set.

[5] `http://sandbox.yahoo.com/`

(henceforth called **TFIDF+Lexicon**). In TFIDF+Lexicon, sentences are ranked by the TFIDF similarity with the query, and then sentences with sentiment words are selected in sequence. This baseline aims to show the performance when we only have access to lexicons without using a learning algorithm.

5 Results

5.1 Evaluating the Ranker

We evaluate our ranker (described in Section 3.1) on the task of best answer prediction. Table 2 compares the average precision and mean reciprocal rank (MRR) of our method to those of three baselines, (1) where answers are ranked randomly (**Baseline (Random)**), (2) by length (**Baseline (Length)**), and (3) by Jensen Shannon Divergence (**JSD**) with all answers. We expect that the best answer is the one that covers the most information, which is likely to have a smaller JSD. Therefore, we use JSD to rank answers in the ascending order. Table 2 manifests that our ranker outperforms all the other methods.

	Baseline (Random)	**Baseline (Length)**	**JSD**	**Ranker (ListNet)**
Avg Precision	0.1305	0.2834	0.4000	**0.5336**
MRR	0.3403	0.4889	0.5909	**0.6496**

Table 2: Performance for best answer prediction. Our ranker outperforms the three baselines.

5.2 Community QA Summarization

Automatic Evaluation. Since human written abstracts are not available for the Yahoo! Answers dataset, we adopt the Jensen-Shannon divergence (JSD) to measure the summary quality. Intuitively, a smaller JSD implies that the summary covers more of the content in the answer set. Louis and Nenkova (2013) report that JSD has a strong negative correlation (Spearman correlation $= -0.737$) with the overall summary quality for multi-document summarization (MDS) on news articles and blogs. Our task is similar to MDS. Meanwhile, the average JSD of the best answers in our test set is smaller than that of the other answers (0.39 vs. 0.49), with an average length of 103 words compared with 67 words for the other answers. Also, on the blog task (Section 5.3), the top two systems by JSD also have the top two ROUGE scores (a common metric for summarization evaluation when human-constructed summaries are available). Thus, we conjecture that JSD is a good metric for community QA summaries.

Table 3 (left) shows that our system using a content coverage function based on Cosine using TFIDF weights, and a dispersion function (h_{sum}) based on lexicon dissimilarity and 100 topics, outperforms all of the compared approaches (paired-t test, $p < 0.05$). The topic number is tuned on the development set, and we find that varying the number of topics does not impact performance too much. Meanwhile, both our system and Dasgupta et al. (2013) produce better JSD scores than the two variants of the Lin and Bilmes (2011) system, which implies the effectiveness of the dispersion function. We further examine the effectiveness of each component that contributes to the objective function (Section 3.4), and the results are shown in Table 3 (right).

	Length	
	100	200
Best answer	0.3858	-
Lin and Bilmes (2011)	0.3398	0.2008
Lin and Bilmes (2011) + q	0.3379	0.1988
Dasgupta et al. (2013)	0.3316	0.1939
Our system	**0.3017**	**0.1758**

	JSD$_{100}$	**JSD$_{200}$**
Rel(evance)	0.3424	0.2053
Rel + Aut(hor)	0.3375	0.2040
Rel + Aut + TM (Topic Models)	0.3366	0.2033
Rel + Aut + TM + Pol(arity)	0.3309	0.1983
Rel + Aut + TM + Pol + Cont(ent Coverage)	0.3102	0.1851
Rel + Aut + TM + Pol + Cont + Disp(ersion)	**0.3017**	**0.1758**

Table 3: [**Left**] Summaries evaluated by Jensen-Shannon divergence (JSD) on Yahoo Answer for summaries of 100 words and 200 words. The average length of the best answer is 102.70. [**Right**] Value addition of each component in the objective function. The JSD on each line is statistically significantly lower than the JSD on the previous ($\alpha = 0.05$).

Human Evaluation. Human evaluation for Yahoo! Answers is carried out on Amazon Mechanical Turk[6] with carefully designed tasks (or "HITs"). Turkers are presented summaries from different systems in a random order, and asked to provide two rankings, one for overall quality and the other for information diversity. We indicate that informativeness and non-redundancy are desirable for quality; however, Turkers are allowed to consider other desiderata, such as coherence or responsiveness, and write down those when they submit the answers. Here we believe that ranking the summaries is easier than evaluating each summary in isolation (Lerman et al., 2009).

[6] https://www.mturk.com/mturk/

We randomly select 100 questions from our test set, each of which is evaluated by 4 distinct Turkers located in United States. 40 HITs are thus created, each containing 10 different questions. Four system summaries (best answer, Dasgupta et al. (2013), and our system with 100 and 200 words respectively) are displayed along with one noisy summary (i.e. irrelevant to the question) per question in random order.[7] We reject Turkers' HITs if they rank the noisy summary higher than any other. Two duplicate questions are added to test intra-annotator agreement. We reject HITs if Turkers produced inconsistent rankings for both duplicate questions. A total of 137 submissions of which 40 HITs pass the above quality filters.

Turkers of all accepted submissions report themselves as native English speakers. An inter-rater agreement of Fleiss' κ of 0.28 (fair agreement (Landis and Koch, 1977)) is computed for quality ranking and κ is 0.43 (moderate agreement) for diversity ranking. Table 4 shows the percentage of times a particular method is picked as the best summary, and the macro-/micro-average rank of a method, for both overall quality and information diversity. Macro-average is computed by first averaging the ranks per question and then averaging across all questions.

For overall quality, our system with a 200 word limit is selected as the best in 44.6% of the evaluations. It outperforms the best answer (31.9%) significantly, which suggests that our system summary covers relevant information that is not contained in the best answer. Our system with a length constraint of 100 words is chosen as the best for quality 12.5% times while that of Dasgupta et al. (2013) is chosen 11.0% of the time. Our system is also voted as the best summary for diversity in 78.7% of the evaluations. More interestingly, both of our systems, with 100 words and 200 words, outperform the best answer and Dasgupta et al. (2013) for average ranking (both overall quality and information diversity) significantly by using Wilcoxon signed-rank test ($p < 0.05$). When we check the reasons given by Turkers, we found that people usually prefer our summaries due to "helpful suggestions that covered many options" or being "balanced with different opinions". When Turks prefer the best answers, they mostly stress on coherence and responsiveness. Sample summaries from all the systems are displayed in Figure 2.

	Length of Summary	**Overall Quality**			**Information Diversity**		
		% Best	Average Rank Macro	Micro	% Best	Average Rank Macro	Micro
Best answer	102.70	31.9%	2.68	2.69	9.6%	3.27	3.29
Dasgupta et al. (2013)	100	11.0%	2.84	2.83	5.0%	2.95	2.94
Our system	100	12.5%	**2.50***	**2.50***	6.7%	**2.43***	**2.43***
Our system	200	**44.6%**	**1.98***	**1.98***	**78.7%**	**1.35***	**1.34***

Table 4: Human evaluation on Yahoo! Answer Data. **Boldface** implies statistically significance compared to other results in the same columns using paired-t test. Both of our systems are ranked higher (i.e. numbers in **bold** with *) than the best answers voted by Yahoo! users and system summaries from Dasgupta et al. (2013).

Question: What is the long term effect of piracy on the music and film industry?

Dasgupta et al. (2013) (Qty Rank=2.75 Div. Rank=2.5):
- In short, I don't think the music industry in particular will ever enjoy the huge profits of the 90's.
- Please-People in those businesses make millions of dollars as it is !! I don't think piracy hurts them at all !!!
- The other thing will be music and movies with less quality.
- Its a big gray area, I dont see anything wrong with burning a mix cd or a cd for a friend so long as youre not selling them for profit.
- By removing the profitability of music & film companies, piracy takes away their motivation to produce new music & movies.

Our system (100 words) (Qty Rank=2.25 Div. Rank=2.25):
- Rising costs for movies and music. The other thing will be music and movies with less quality.
- Now, with piracy, there isn't the willingness to take chances.
- But it's also like the person put the effort into it and they aren't getting paid. It's a big gray area, I don't see anything wrong with burning a mix cd or a cd for a friend so long as you're not selling them for profit.
- It is forcing them to rework their business model, which is a good thing.

Our system (200 words) (Qty. Rank=2.25, Div Rank=1.25):
- Rising costs for movies and music. The other thing will be music and movies with less quality.
- Now, with piracy, there isn't the willingness to take chances. American Idol is the result of this. .... The real problem here is that the mainstream music will become even tighter. Record labels will not won't to go far from what is currently like by the majority.
- I hate when people who have billions of dollars whine about not having more money. But it's also like the person put the effort into it and they aren't getting paid ... I don't see anything wrong with burning a mix cd or a cd for a friend ....
- It is forcing them to rework their business model, which is a good thing.
- By removing the profitability of music & film companies, piracy takes away their motivation to produce new music & movies.

Figure 2: Sample summaries from Dasgupta et al. (2013), and our systems (100 words and 200 words). Sentences from separate bullets (•) are partial answers from different users.

[7]Note that we aim to compare results with the gold-standard best answers of about 100 words. The evaluation of the 200-word summaries is provided only as an additional data-point.

5.3 Blog Summarization

Automatic Evaluation. We use the ROUGE (Lin and Hovy, 2003) software with standard options to automatically evaluate summaries with reference to the human labeled nuggets as those are available for this task. ROUGE-2 measures bigram overlap and ROUGE-SU4 measures the overlap of unigram and skip-bigram separated by up to four words. We use the ranker trained on Yahoo! data to produce relevance ordering, and adopt the system parameters from Section 5.2. Table 5 (left) shows that our system outperforms the best system in TAC'08 with highest ROUGE-2 score (Kim et al., 2008), the two baselines (TFIDF+Lexicon, and our ranker), Lin and Bilmes (2011), and Dasgupta et al. (2013).

	ROUGE-2	**ROUGE-SU4**	**JSD**
Best system in TAC'08	0.2923	0.3766	0.3286
TFIDF + Lexicon	0.3069	0.3876	0.2429
Ranker (ListNet)	0.3200	0.3960	0.2293
Lin and Bilmes (2011)	0.2732	0.3582	0.2330
Lin and Bilmes (2011) + q	0.2852	0.3700	0.2349
Dasgupta et al. (2013)	0.2618	0.3500	0.2370
Our system	**0.3234**	**0.3978**	*0.2258*

	Pyramid F-score
Best system in TAC'08	0.2225
Lin and Bilmes (2011)	0.2790
Our system	**0.3620**

Table 5: Results on TAC'08 dataset. [**Left**] Our system has statistically significant better ROUGE scores than all the other systems except our ranker (paired-t test, $p < 0.05$). Our system also achieves the best (lowest) JS divergence. [**Right**] Human evaluation with Pyramid F-score. Our system significantly outperforms the other two systems (paired-t test, $p < 0.05$).

Human Evaluation. For human evaluation, we use the standard Pyramid F-score used in the TAC'08 opinion summarization track with $\beta = 3$, as recommended by (Dang, 2008). In the TAC task, systems are allowed to return up to 7,000 non-white characters for each question. Since the TAC metric favors recall we do not produce summaries shorter than 7,000 characters.

We ask two human judges to evaluate our system as well as the one that got the highest Pyramid F-score in the TAC competition. We also compare with the summaries generated by (Lin and Bilmes, 2011). Cohen's κ for inter-annotator agreement is 0.68 (substantial). While we did not explicitly evaluate the non-redundancy of the systems, both of our judges report that summaries generated by our system contains less redundant information than Lin and Bilmes (2011).

5.4 Further Discussion

Yahoo! Answer				
	DISPERSION$_{sum}$		DISPERSION$_{min}$	
DISSIMI	Cont$_{tfidf}$	Cont$_{sem}$	Cont$_{tfidf}$	Cont$_{sem}$
Semantic	0.3143	0.324 3	0.3129	0.3232
Topical	0.3101	0.3202	0.3106	0.3209
Lexical	**0.3017**	0.3147	0.3071	0.3172

TAC 2008				
	DISPERSION$_{sum}$		DISPERSION$_{min}$	
DISSIMI	Cont$_{tfidf}$	Cont$_{sem}$	Cont$_{tfidf}$	Cont$_{sem}$
Semantic	0.2216	0.2169	0.2772	0.2579
Topical	0.2128	0.2090	**0.3234**	0.3056
Lexical	0.2167	0.2129	0.3117	0.3160

Table 6: Effect of different dispersion functions, content coverage, and dissimilarity metrics on our system. [**Left**] JSD values for different combinations on Yahoo! data, using LDA with 100 topics. All systems are significantly different from each other at significance level $\alpha = 0.05$. Systems using summation of distances for dispersion function (h_{sum}) uniformly outperform the ones using minimum distance (h_{min}). [**Right**] ROUGE scores of different choices for TAC 2008 data. All systems use LDA with 40 topics. The parameters of our systems are adopted from the ones tuned on Yahoo! Answers.

Given that the text similarity metrics and dispersion functions play important roles in the framework, we further study the effectiveness of different content coverage functions (Cosine using TFIDF vs. Semantic), dispersion functions (h_{sum} vs. h_{min}), and dissimilarity metrics used in dispersion functions (Semantic vs. Topical vs. Lexical). Results on Yahoo! Answer (Table 6 (left)) show that systems using summation of distances for dispersion functions (h_{sum}) uniformly outperform the ones using minimum distance (h_{min}). Meanwhile, Cosine using TFIDF is better at measuring content coverage than WordNet-based semantic measurement, and this may due to the limited coverage of WordNet on verbs. This is also true for dissimilarity metrics. Results on blog data (Table 6 (right)), however, show that using minimum distance for dispersion produces better results. This indicates that optimal dispersion function varies by genre. Topical-based dissimilarity also marginally outperforms the other two metrics in blog data.

6 Conclusion

We propose a submodular function-based opinion summarization framework. Tested on community QA and blog summarization, our approach outperforms state-of-the-art methods that are also based on submodularity in both automatic evaluation and human evaluation. Our framework is capable of including statistically learned sentence relevance and encouraging the summary to cover diverse topics. We also study different metrics on text similarity estimation and their effect on summarization.

References

Eneko Agirre, Daniel Cer, Mona Diab, and Aitor Gonzalez-Agirre. 2012. Semeval-2012 task 6: A pilot on semantic textual similarity. In *Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 385–393, Montréal, Canada, 7-8 June. Association for Computational Linguistics.

Hadi Amiri, Zheng-Jun Zha, and Tat-Seng Chua. 2013. A pattern matching based model for implicit opinion question identification. In *AAAI*. AAAI Press.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, March.

Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. 2007. Learning to rank: From pairwise approach to listwise approach. In *Proceedings of the 24th International Conference on Machine Learning*, ICML '07, pages 129–136, New York, NY, USA. ACM.

Jaime Carbonell and Jade Goldstein. 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, pages 335–336, New York, NY, USA. ACM.

Asli Celikyilmaz, Dilek Hakkani-Tur, and Gokhan Tur. 2010. Lda based similarity modeling for question answering. In *Proceedings of the NAACL HLT 2010 Workshop on Semantic Search*, SS '10, pages 1–9, Stroudsburg, PA, USA. Association for Computational Linguistics.

Barun Chandra and Magnús M. Halldórsson. 1996. Facility dispersion and remote subgraphs. In *Proceedings of the 5th Scandinavian Workshop on Algorithm Theory*, SWAT '96, pages 53–65, London, UK, UK. Springer-Verlag.

Hoa Tran Dang. 2008. Overview of the tac 2008 opinion question answering and summarization tasks. In *Proc. TAC 2008*.

Van Dang. 2011. RankLib. http://www.cs.umass.edu/˜vdang/ranklib.html.

Anirban Dasgupta, Ravi Kumar, and Sujith Ravi. 2013. Summarization through submodularity and dispersion. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1014–1022, Sofia, Bulgaria, August. Association for Computational Linguistics.

Hal Daumé, III and Daniel Marcu. 2006. Bayesian query-focused summarization. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, ACL-44, pages 305–312, Stroudsburg, PA, USA. Association for Computational Linguistics.

Andrea Esuli and Fabrizio Sebastiani. 2006. Sentiwordnet: A publicly available lexical resource for opinion mining. In *In Proceedings of the 5th Conference on Language Resources and Evaluation (LREC06*, pages 417–422.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, pages 168–177, New York, NY, USA. ACM.

George Karypis. 2003. CLUTO - a clustering toolkit. Technical Report #02-017, November.

Hyun Duk Kim, Dae Hoon Park, V.G.Vinod Vydiswaran, and ChengXiang Zhai. 2008. Opinion summarization using entity features and probabilistic sentence coherence optimization: Uiuc at tac 2008 opinion summarization pilot. In *Proc. TAC 2008*.

J R Landis and G G Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.

Kevin Lerman, Sasha Blair-Goldensohn, and Ryan McDonald. 2009. Sentiment summarization: Evaluating and learning user preferences. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '09, pages 514–522, Stroudsburg, PA, USA. Association for Computational Linguistics.

Baoli Li, Yandong Liu, and Eugene Agichtein. 2008a. Cocqa: Co-training over questions and answers with an application to predicting question subjectivity orientation. In *EMNLP*, pages 937–946.

Wenjie Li, You Ouyang, Yi Hu, and Furu Wei. 2008b. Polyu at tac 2008. In *Proc. TAC 2008*.

Hui Lin and Jeff Bilmes. 2011. A class of submodular functions for document summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 510–520, Stroudsburg, PA, USA. Association for Computational Linguistics.

Chin-Yew Lin and Eduard Hovy. 2000. The automated acquisition of topic signatures for text summarization. In *Proceedings of the 18th Conference on Computational Linguistics - Volume 1*, COLING '00, pages 495–501, Stroudsburg, PA, USA. Association for Computational Linguistics.

Chin-Yew Lin and Eduard Hovy. 2003. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, pages 71–78.

Yuanjie Liu, Shasha Li, Yunbo Cao, Chin-Yew Lin, Dingyi Han, and Yong Yu. 2008. Understanding and summarizing answers in community-based question answering services. In *Proceedings of the 22Nd International Conference on Computational Linguistics - Volume 1*, COLING '08, pages 497–504, Stroudsburg, PA, USA. Association for Computational Linguistics.

Annie Louis and Ani Nenkova. 2013. Automatically assessing machine summary content without a gold standard. *Comput. Linguist.*, 39(2):267–300, June.

Xiaoqiang Luo, Hema Raghavan, Vittorio Castelli, Sameer Maskey, and Radu Florian. 2013. Finding what matters in questions. In *HLT-NAACL*, pages 878–887.

Ryan McDonald. 2007. A study of global inference algorithms in multi-document summarization. In *Proceedings of the 29th European Conference on IR Research*, ECIR'07, pages 557–564, Berlin, Heidelberg. Springer-Verlag.

Rada Mihalcea, Courtney Corley, and Carlo Strapparava. 2006. Corpus-based and knowledge-based measures of text semantic similarity. In *Proceedings of the 21st National Conference on Artificial Intelligence - Volume 1*, AAAI'06, pages 775–780. AAAI Press.

G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. 1978. An analysis of approximations for maximizing submodular set functionsI. *Mathematical Programming*, 14(1):265–294, December.

Ani Nenkova and Lucy Vanderwende. 2005. The impact of frequency on summarization. *Microsoft Research, Redmond, Washington, Tech. Rep. MSR-TR-2005-101*.

Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Found. Trends Inf. Retr.*, 2(1-2):1–135, January.

Michael J. Paul, ChengXiang Zhai, and Roxana Girju. 2010. Summarizing contrastive viewpoints in opinionated text. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, pages 66–76, Stroudsburg, PA, USA. Association for Computational Linguistics.

Ruben Sipos, Pannaga Shivaswamy, and Thorsten Joachims. 2012. Large-margin learning of submodular summarization models. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '12, pages 224–233, Stroudsburg, PA, USA. Association for Computational Linguistics.

Philip J. Stone, Dexter C. Dunphy, Marshall S. Smith, and Daniel M. Ogilvie. 1966. *The General Inquirer: A Computer Approach to Content Analysis*. MIT Press, Cambridge, MA.

Veselin Stoyanov and Claire Cardie. 2006. Partially supervised coreference resolution for opinion summarization through structured rule learning. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, EMNLP '06, pages 336–344, Stroudsburg, PA, USA. Association for Computational Linguistics.

Mattia Tomasoni and Minlie Huang. 2010. Metadata-aware measures for answer summarization in community question answering. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 760–769, Stroudsburg, PA, USA. Association for Computational Linguistics.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 347–354, Stroudsburg, PA, USA. Association for Computational Linguistics.

Pengtao Xie and Eric Xing. 2013. Integrating document clustering and topic modeling. In *Proceedings of the Twenty-Ninth Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-13)*, pages 694–703, Corvallis, Oregon. AUAI Press.

Hong Yu and Vasileios Hatzivassiloglou. 2003. Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Jianxing Yu, Zheng-Jun Zha, and Tat-Seng Chua. 2012. Answering opinion questions on products by exploiting hierarchical organization of consumer reviews. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, pages 391–401, Stroudsburg, PA, USA. Association for Computational Linguistics.