Novelty based Ranking of Human Answers for Community Questions

Adi Omari
Technion IIT
Haifa,32000, Israel
omari@cs.technion.ac.il

David Carmel, Oleg Rokhlenko,
Idan Szpektor
Yahoo Research
Haifa 31905, Israel
{dcarmel,olegro,idan}@yahoo-inc.com

ABSTRACT

Questions and their corresponding answers within a community-based question answering (CQA) site are frequently presented as top search results for Web search queries and viewed by millions of searchers daily. The number of answers for CQA questions ranges from a handful to dozens, and a searcher would be typically interested in the different suggestions presented in various answers for a question. Yet, especially when many answers are provided, the viewer may not want to sift through all answers but to read only the top ones. Prior work on answer ranking in CQA considered the qualitative notion of each answer separately, mainly whether it should be marked as best answer. We propose to promote CQA answers not only by their relevance to the question but also by the diversification and novelty qualities they hold compared to other answers. Specifically, we aim at ranking answers by the amount of new aspects they introduce with respect to higher ranked answers (novelty), on top of their relevance estimation. This approach is common in Web search and information retrieval, yet it was not addressed within the CQA settings before, which is quite different from classic document retrieval. We propose a novel answer ranking algorithm that borrows ideas from aspect ranking and multi-document summarization, but adapts them to our scenario. Answers are ranked in a greedy manner, taking into account their relevance to the question as well as their novelty compared to higher ranked answers and their coverage of important aspects. An experiment over a collection of Health questions, using a manually annotated gold-standard dataset, shows that considering novelty for answer ranking improves the quality of the ranked answer list.

Keywords: Community-based question answering; Novelty; Diversification

1. INTRODUCTION

Community-based Question Answering (CQA), a platform in which people can ask questions and receive answers from other people, has become a useful tool for information needs that are not answered simply by viewing a Web page, including recommendations, suggestions and homework help [26]. In popular CQA sites, such as Yahoo Answers, Baidu Zhidao, Answers.com, and Stack Overflow, hundreds of millions of answered questions have been collected. The answers to the questions are not only viewed by the asker herself, but are frequently presented as top search results for Web search queries and viewed by millions of searchers daily, in the form of a question and its corresponding answers.

The number of answers for CQA questions ranges from a handful to dozens, and even sometimes hundreds in cases of popular questions. We found that in Yahoo Answers more than 38% of the answered questions have at least 5 answers. For some questions, such as factoids, readers would be content with a single high-quality answer. However, in other types of questions, such as asking for recommendations or opinions, the asker as well as other viewers would benefit from different views or suggestions. Still, especially when many answers are provided, the reader may not want to sift through all answers but to read only the top ones. While a few works did address the task of answer ranking in CQA [22, 42, 45], they considered mainly the qualitative notions of each answer separately, its relevance to the question or whether it should be marked as best answer. These works did not address the overall quality of the ranked list of answers. Especially, they did not consider the complementary information provided by different answers.

In this paper we follow diversification approaches in Web search and Information Retrieval (IR) [13, 14] and promote CQA answers not only by their relevance to the question but also by diversification and novelty qualities they hold compared to other answers. Specifically, assuming the information need behind a CQA question can be partitioned into relevant "subtopics" or "aspects", our goal is to rank the corresponding answers not only by their relevance but also by the amount of aspects they cover (diversity), and more specifically, by the amount of new aspects they introduce with respect to higher ranked answers (novelty). Though diversification of CQA answers for input questions was considered before, it was under an IR setup, where results are retrieved from a large collection of answers [1]. To the best of our knowledge, this is the first time this task is addressed under the CQA setup, where the dozen or so answers to be ranked are manually provided by answerers directly for the target question.

There is a large body of research on document novelty and diversification for Web Search and IR [13, 4, 2, 33, 12, 43]. Yet, under a CQA setup this task bears significantly different traits. First, since the answers are provided by humans in direct response to the given question, most of these answers are relevant to the question to some extent [3]. This is a very different scenario compared to document retrieval, in which only a handful of documents are relevant out of a large list of matching documents. As a result, IR methods that incorporate novelty detection on top of relevance estimation (*e.g.*, MMR [8]) are somewhat unfitted for the CQA scenario (see Section 4). Second, CQA answers are typically much shorter than Web

SIGIR '16, July 17–21, 2016, Pisa, Italy.
© 2016 ACM. ISBN 978-1-4503-4069-4/16/07. . . $15.00
DOI: http://dx.doi.org/10.1145/2911451.2911506

documents, and are therefore more condensed in terms of the content they provide. Third, IR methods aim at short ambiguous Web queries as input while CQA questions are longer and more detailed.

Another task that our CQA scenario resembles is that of summarizing different news articles on the same event [30]. In this scenario all news articles (answers) are "relevant", describing the same event (question), but may provide different views and facts (aspects) on the event by different reporters (answerers). The news articles should be summarized to provide a comprehensive view about the event. Specifically, in query-focused summarization, a typical approach is to rank sentences based on their similarity to the query and then cluster them and pick representative sentences based on the clusters [25].

While drawing similarities between news summarization and our task, the final goal is quite different, since we do not need to provide a summarization of the answers but to rank them. Furthermore, news articles are longer and well structured. This is not the case in CQA answers, which are typically short with many empty connecting unstructured sentences. Most notably in our task, many aspects may be clamped together in a single sentence, which makes the typical approach of looking at a sentence as an atomic text unit inappropriate. As an example, consider the question "*Whats your best migraine cure?*" and the provided answers "*1) Excedrine migraine, phenergan, dark room, cold compress, 8 hours of sleep*", and "*2) Take medicine, go in a dark room and sleep for at least an hour, it helps to use earplugs*". These answers contain several complementary suggestions each and they share several discussed topics: sleeping, being in a dark room, and medicine taking.

The method we propose for novelty-based answer ranking looks at syntactic propositions instead of sentences as the atomic text units. Under this view, answer 2 in the example above is decomposed into "*2.1) Take medicine*", "*2.2) go in a dark room and sleep for at least an hour*" and "*2.3) it helps to use earplugs*". We then measure the similarity between propositions and generate a hierarchical clustering of them across all answers. Finally, answers are ranked in a greedy manner based on the amount of diverse propositions they contain, taking into account each proposition's relevance to the question as well as its dissimilarity to propositions in higher ranked answers.

We tested our algorithm on a collection of health-related questions and their answers from Yahoo Answers, which were manually annotated with gold-standard aspects in each answer. We used conventional diversity-based IR evaluation metrics and also propose two novel evaluation metrics that better emphasize novelty under our settings. We compare our approach with several state-of-the-art novelty-based ranking algorithms and show that our algorithm outperforms prior work under all metrics.

Our main contributions in this paper are:

- Introducing the task of novelty-based answer ranking under the CQA setup
- Novelty-base answer ranking algorithm for CQA, which considers novelty on top of relevance
- Manually annotated dataset[1] of CQA questions together with gold-standard aspects per answer
- New evaluation metrics that emphasize diversification and novelty in CQA answer ranking

2. RELATED WORK

Novelty-based answer ranking in CQA has not attracted much attention so far. However, it is related to a several research fields. In this section we review the most related ones.

2.1 Answer Ranking

Answer ranking is essential for CQA services due to the high variance in the quality of answers. Several previous studies dealt with answer ranking for CQA sites [21, 22, 6, 27, 42, 45]. Jeon et al. [21] predicted answer quality using non-textual features of the answers. Bian et al. [6] integrated answer similarity to the question with community feedback information. Jurczyk and Agichtein [22] measured user expertise using link analysis over the question-answers graph, assuming answers given by authoritative users tend to be of high quality. Tu et al. [42] proposed an analogical reasoning-based method by measuring how valuable an answer is given its similarity to the set of best answers of similar resolved questions. Zhou et al. [45] additionally exploited three categories of user profile information – engagement-related, authority-related and level-related, for answer ranking in CQA.

Other works estimated the likelihood of an answer to be selected as best answer by the asker; an estimate that might be further used for answer ranking. Liu et. al. [27] and Shah and Pomerantz [36] trained a classifier that predicts this likelihood based on features describing the question text, category, question-answer similarity, user reputation, and user feedback. Suryanto et al. [40] studied additional features derived from the answerer expertise. The authors argued that an answerer can have different expertise levels for different topics which should be taken into account during answer quality estimation. Dalip et al. [16] proposed a learning to rank approach using a comprehensive set of eight different groups of features derived from the question-answer pair.

In this work we follow up on relevance estimation in prior work and combine it with novelty detection. We leave the integration with non-textual features such as answerer reputation and user feedback (*e.g.* votes) for future work.

2.2 Document Diversification

Novelty detection and search result diversification is an important research track in IR. Carbonel and Goldstein [8] proposed the Maximal Marginal Relevance (MMR) approach, in which documents in the result list are evaluated based on their relevance to the query as well as their difference from previously selected documents. Zhai et al. [44] studied how an interactive retrieval system can best support a user that gathers information about the different aspects of a topic. Agrawal et al. [4] selected the next document that best matches DMOZ categories that are related to a Web query but are not well covered by higher ranked documents. The xQuAD algorithm [35] measures document diversity by its relevance to sub-queries of an ambiguous Web query, which are not related to previously ranked documents. In the heart of xQuAD, a set of sub-queries that describe the different aspects of the query are assessed against each retrieved document. The sub-queries are generated using query expansion techniques or using query reformulations from search engines. Croft and Dang [15] suggested to use surface terms instead of topics as indications of aspects. They identify candidate terms as words in the vicinity of query terms that appear in retrieved documents and weigh their topicality and productiveness. However, in the CQA setup, question terms, which express information need, often do not appear in the answers, which express solutions. Therefore, a different aspect identification approach is required. We emphasize that diversification approaches for short ambiguous Web queries, whose aspects can be modeled

[1] Available at http://webscope.sandbox.yahoo.com

by high level categories or additional keywords [4, 35, 17, 15], deal with scenarios that are very different from ours, in which the "query" is a detailed CQA question and its aspects are much more fine grained and very specific to the question's context.

Clarke et al. [13] proposed a framework for evaluation that systematically rewards novelty and diversity in the search results, based on cumulative gain. The task of novelty detection at the sentence level was explicitly defined at the TREC novelty track [39]: find relevant sentences for a given query; then extract only novel sentences from that list [5, 24].

Two works that are strongly related to our paper deal with diversifying user generated content. Krestel et al. [23] re-ranked user product reviews using star ratings and a latent topic model. The algorithm uses Latent Dirichlet Allocation (LDA) [7] to map different reviews of the same product to a latent topic space. It attempts to approximate the topic distribution for all reviews by selecting the top k reviews in a greedy way. Specifically, it chooses the next review to the ranked list as the one that minimizes the KL divergence between the all-reviews topic distribution and the topic distribution of the top reviews (including the new candidate). The algorithm's final goal is to construct a good summary of all reviews, assuming the reviews complement each other. This framework covers different ranking strategies based on the user need: summarizing all reviews; focusing on a particular latent topic; or focusing on sentiment aspects of the reviews.

Achananuparp et al. [1] introduced a graph ranking model for retrieving a diverse set of answers from a large answer archive given a complex input question. This algorithm constructs an ergodic directed transition matrix that combines similarities between all pairs of retrieved answers and the relevance of each answer to the question. Redundancy relations among answers are modeled in this algorithm by assigning a negative sign to the weight of edges between the answer nodes in the graph of candidate answers. The unique stationary distribution of the nodes, which is extracted via a PageRank-like approach, induces the ranking of the answers, with high probability answers on top. This algorithm showed significant improvement over prior work, including MMR. We replicated the algorithms in [1] and [23] as baselines for our study.

2.3 Multi Document Summarization

An abundance of prior work explored the task of multi-document summarization (MDS), in which a coherent summarization is generated from different documents covering similar information [28, 32, 38, 30]. We focus here on the application of these techniques in the CQA domain. Liu et al. [27] demonstrated that many open or opinion questions have multiple good answers which sometimes are better than the selected best answer. By summarizing these answers using query focused summarization techniques they were able to provide an alternative best answer. Chan et al. [9] addressed the problem of "incomplete answer", *i.e.* a best answer for a complex question that misses valuable information contained in other answers. They proposed a general Conditional Random Field (CRF) framework to create a complementary answer summary. Pande et al. [31] tackled the same incomplete answer problem by identifying a diverse set of informative threads in the user answers and summarizing each one of them.

Motivated by MDS techniques, we are interested in recognizing the important topics in an answer. Yet, in our task we are not required to generate a final summarization. Instead, we explore ways to measure the overall amount of diversity and novelty in each answer. In addition, prior work in CQA summarization operated at the sentence level. We argue that a more fine-grained analysis should be taken for aspect detection in CQA.

Excedrine migraine, phenergan, dark room, cold compress, 8 hours of sleep
Claratin and zyrtec did nothing for be then my allergist prescribed singular and it worked like a charm
try to get at least 7.5hrs of sleep and regular exercise
Id drink green tea as late as 10 pm at night but end up staying up really late, its a personal choice, you could always try a sleep aid

Table 1: Examples for single sentences in CQA answers that combine different information aspects for the question "*Whats your best migraine cure? I have the worst headache...*".

3. NOVELTY-BASED ANSWER RANKING

In this work we propose an answer ranking solution that focuses on diversity and novelty notions of answers on top of relevance. Our goal is to rank the answers in a way that provides the reader a fast coverage of all the important aspects mentioned in the answers for the given question. Such ranking presents the user with "the whole picture" by reading only the first few answers instead of skipping repeated suggestions or opinions in order to find more relevant but diverse material.

We consider all the answers for a question to be relevant to some extent [3]. This does not mean that all the answer text is relevant to the question. Some answer parts may be irrelevant, such as emotional response ("*I am sorry to hear that Joe.*"), connecting sentences ("*but let me tell you how things should really work*") and personal agenda ("*we should blame the president for such situations*"). In addition, we do not consider all aspects as equal. Some recommendations or solutions are given in several answers, and assuming that "wisdom of the crowd" takes effect, we would like to view them as more important to the asker than others. Our task is therefore to rank higher answers that contain novel and diverse aspects, and to promote those aspects that are repeated in other answers as well.

Looking at the considerations above, we find an analogy between our task and the task of query-focused multi-document summarization [18, 41], as discussed earlier. We therefore present an algorithm that borrows ideas from textual summarization, but employ them differently for our ranking task. Similarly to extractive summarization, we focus on basic textual units as conveying the different information elements in an answer. Yet, unlike the common view of sentences as basic units, we found that often CQA answerers list several pieces of information in a single sentence. Table 1 presents such examples. We therefore focus on propositions, instead of sentences, as our basic units.

At a high level, our algorithm starts by extracting all propositions in each answer. Propositions that are irrelevant to the question are automatically filtered out based on their semantic dissimilarity from the question. The algorithm then measures the similarity between the remaining propositions as a proxy to aspect diversity, and the "importance" of each proposition is assessed based on its occurrences in different answers. Finally, a greedy procedure selects at each round the answer that best combines a set of relevant propositions that are both diverse, novel, and important. Algorithm 1 depicts this procedural overview of our algorithm. In the reminder of this section we will detail each of the steps in our algorithm.

3.1 Proposition extraction

In this step the algorithm extracts the basic semantic units, called *propositions*, which we consider as conveying coherent information given by the answerer. To this end, it starts, similarly to summarization and classic novelty detection in IR [13], by sentence splitting each answer. It then proceeds by syntactically analyzing each

```
Input: question $q$
Input: set of answers $Ans$
$OrderedAns = []$
/* Init                                                    */
foreach $a \in Ans$ do
    $Propositions[a] = extractRelevantPropositions(a, q)$
    foreach $p \in Propositions[a]$ do
        $Novelty[p] = 1$
    end
end
/* Rank                                                    */
while $Ans \neq \emptyset$ do
    $selectedAnswer = select(Ans, Novelty, Propositions)$
    $OrderedAns.add(selectedAnswer)$
    $Ans = Ans \setminus \{selectedAnswer\}$
    foreach $p \in Propositions$ do
        Novelty[p] =
        $updateNovelty(p, Novelty, selectedAnswer)$
    end
end
return $OrderedAns$
```

Algorithm 1: Novelty-based answer ranking overview

sentence. Specifically, the algorithm parses each sentence with the Clear[2] dependency parser [11]. It then splits the dependency tree into sub-trees based on specific edge types that indicate proposition boundaries. These edges include the following kinds of connectives: *ccomp*, *npadvmod* and *conj*. Finally, each sub-tree is turned into a text string by ordering the words in the sub-tree according to their original position in the sentence. As an example of procedure, the last sentence in Table 1 is split into "*Id drink green tea as late as 10 pm at night*", "*end up staying up really late*", "*its a personal choice*" and "*you could always try a sleep aid*".

3.2 Proposition filtering

Proposition filtering is an essential step in our algorithm, it ensures that answers will not be promoted for including irrelevant propositions like recurring empathic statements such as "*its a personal choice*" in the above example.

In order to keep only propositions relevant to the question, we rank them based on their similarity to the question. Yet, simple surface word comparison will not suffice for two reasons. First, the language used in the answer is very different from the question's language, one containing words that help express an information need and the other words that convey a solution to that need. For example, the question "*what should I see in Paris?*" and the answer "*the Eiffel tower is a must*" have no shared terms, but the answer is very relevant. Second, propositions are rather short texts while some questions may be verbose and long, containing several sentences. Therefore, surface level similarity measures such as Cosine or Jaccard will fail to identify relevant propositions.

We address these two issues by mapping the answer and question to a shared latent space and measure their similarity there. To this end, we employ a variant of the Explicit Semantic Analysis (ESA) approach [20]. Under ESA, a text is mapped to a semantic space in which each dimension is a Wikipedia page (concept). This is done by retrieving the most relevant Wikipedia documents using a search engine, given the whole text as a query. Once the vectors containing the top results for the two compared texts are retrieved, cosine similarity between the two vectors is computed to measure their similarity.

ESA is a successful semantic representation for texts. Still, standard ESA does not compensate for the difference in languages between questions and answers. To address this issue, we follow a variant of ESA, denoted *CQA-ESA*, in which instead of Wikipedia documents the latent space is defined over a collection of structured documents containing CQA questions and their best answers [37]. Each CQA document is a dimension and, as in standard ESA, the representation of a text is by retrieving the top documents in the collection. CQA-ESA differs from standard ESA in the way documents are retrieved. When representing questions, the documents are retrieved by searching only on the question field of each document, while when answers are provided as queries, documents are retrieved by searching over the answer field of each document. This way, the proper language is used for searching for each type of text, but the final latent space is shared – the document ids. We note that the underlying assumption of this approach is that the best-answer of a question is typically highly relevant to it, and therefore the two fields of a single document convey the same semantics in different "languages".

We use CQA-ESA similarity to the question for ranking the propositions in all answers. Following an empirical study over several dozen questions not in our test-set, we chose to keep the top 90% propositions similar to the question as relevant, filtering out the rest. As document collection we used a random sample of 2 million questions and their best answers from Yahoo Answers. We apply the Lucene[3] search engine, under its default settings, for searching over this collection. The search results are used for CQA-ESA representation of the queried text.

3.3 Answer Diversity and Importance

Motivation and Definitions

As discussed above, we would like to discover which relevant propositions present different aspects that appear in the answers, in order to promote answers that contain a diverse set of aspects. In addition, following the notion of "wisdom of the crowd" we would like to promote answers that include aspects that are shared with other answers as a measure of importance.

We chose to employ semantic similarity between two propositions, $sim(p, o)$, as a measure of diversity. That is, the more similar two propositions are the less diverse the aspects they convey are considered. We use proposition similarity also for importance assessment. Specifically, the more similar a proposition is to many propositions in other answers the more *support* we say it has. We describe two ways to estimate proposition support in Section 3.4.

In order to compute support and diversity we next introduce our semantic similarity function $sim(p, o)$ between two propositions.

Semantic Similarity Function

To compute the semantic similarity between two propositions we experimented with four unsupervised similarity measures:

TF-IDF. The cosine between the TF-IDF vectors of the two propositions, after stop-word removal and stemming. This is a typical surface-word similarity measure [5]. Term frequency of a term is its number of occurrences in the proposition. Document frequency of the term is counted over a collection of randomly sampled 16 million questions and their best answers from Yahoo Answers.

Word2Vec. Since propositions are short, and in general people tend to use different wordings to express the same information, we wanted to utilize measures that may overcome such differences in word selection. The following *Word2Vec* measure employs the

[2] https://code.google.com/p/clearparser/

[3] http://lucene.apache.org

Word2Vec model [29], which maps words to a low dimensional space such that semantically similar words are close to each other in this space. We used a publicly available model trained on part of Google news data-set[4] (about 100 billion words) to map words to a 500 dimension space. We then apply the following function to compute proposition similarity $sim(p, o)$:

$$Coverage(p, o) = \frac{1}{|p|} \sum_{t_p \in p} \max_{t_o \in o} [cosine(w2v(t_p), w2v(t_o))]$$

$$sim(p, o) = \sqrt{Coverage(p, o) \cdot Coverage(o, p)}$$

where p and o are propositions, t_p and t_o are terms in p and o respectively, and $w2v(t)$ is the Word2Vec representation of term t.

ESA. As another measure for semantic proposition similarly beyond surface wording, we represent each proposition with its ESA vector over the Wikipedia dump from Feb 2014, using Lucene as the search engine. The cosine between the two vectors is taken as their similarity measure.

CQA-ESA. Similarly to the ESA similarity measure, we use the CQA-ESA representation of each proposition (see Section 3.2) and compute the cosine between the two CQA-ESA vectors.

During our research we found that the performance of our similarity measures differ from one question to another. In order to get a consistently well performing similarity function we combined them in a supervised way. Specifically, we learned a classifier for predicting whether two propositions represent the same answer aspect for a given question, where the input features of each proposition pair are the four unsupervised similarity measures described above. The training set consists of pairs of propositions together with labels indicating if they capture the same aspect or not. A detailed description of this dataset is presented in Section 4. We used the SVM implementation in Weka[5] as our classifier. As a final similarity score we used the classifier's normalized output between $[0, 1]$. We note that this is the only supervised component in our algorithm.

3.4 Greedy Answer Ranking

Given a question and its corresponding set of answers, we would like to rank the answers based on the diversity and importance of their propositions. We follow a common approach in diversification literature [8, 44, 4, 35] and present a greedy ranking framework. Our algorithm iteratively selects the next answer to be added to the ranked list, (denoted by $select()$ in Algorithm 1), by considering how much each answer includes: (a) diverse aspects; (b) important aspects; (c) novel aspects that did not appear in higher ranked answers. Then, the novelty measure of all propositions is updated based on the amount of "support" they are given by the selected answer (denoted by $updateNovelty()$ in Algorithm 1). This iterative procedure continues as long as there are still answers that are not added to the ranked list, resulting in a complete ordering of all answers. We next present two methods for selecting the next answer, and for updating proposition novelty. These methods define the two variants of our overall ranking algorithm.

3.4.1 Similarity based answer selection

The first method uses the similarity between propositions directly for answer selection. The answer's score is proportional to the amount of "support" it provides for novel, yet uncovered propositions. Formally, each proposition's novelty assessment is maintained by the $Novelty[p]$ property, according to how much the aspect it represents is already represented by other propositions in higher ranked answers. We first define $Support(p, a)$, measuring how well answer a supports the information given in proposition p, based on p's similarity to the answer's propositions. Then, the score of a is determined by summing a's support over all propositions, weighted by their novelty property.

$$Support(p, a) = 1 - \prod_{p_a \in a} (1 - sim(p, p_a)) \qquad (1)$$

$$Score(a) = \sum_{p \in Propositions} Novelty[p] \cdot Support(p, a)$$

We note that $Support(p, a)$ acts as a noisy-or formulation of similarity – it is zero only when all answer's propositions are zero-similar to p, and it supports p when at least one of a's propositions is similar to p.

Once the $selectedAnswer$, the highest scoring answer, is chosen, the $Novelty$ assessment of each proposition is updated by the novelty update function

$$Novelty[p] = Novelty[p] \cdot (1 - Support(p, selectedAnswer))$$

We would like to emphasize two properties of this selection procedure. First, during the re-computation phase, the $Novelty$ property of any proposition p of the selected answer becomes 0, since $Support(p, selectedAns) = 1$. Such propositions will no longer contribute to the score of lower ranked answers in future iterations. Therefore, the $Score(a)$ formula only considers propositions in unranked answers for contribution to an answer's score.

Second, the scoring formula includes a notion of proposition importance since an important proposition is similar to many propositions that capture the same aspect. Hence, if an answer contains important propositions, many novel propositions will contribute to the answer score, more than to an answer with the same number of novel propositions which correspond to uncommon aspects.

We note that our novelty update formula resembles that of the xQuAD algorithm [35], as it downgrades the importance of aspects that are already supported by higher ranking documents. Yet, xQuAD's aspects are sub-queries, variations, or expansions of the original query, while our update model is based on the similarity measurement between answer propositions.

3.4.2 Hierarchical clustering based answer selection

Ideally, we would like to have clusters of propositions, each reflecting a single specific aspect within the answers. The size of the clusters would indicate the importance of each aspect, and the diversity of each answer could be easily derived from the number of its related clusters (aspects).

Such clustering is also in the heart of multi-document summarization, since picking a representative sentence from each cluster would generate a comprehensive yet compact summarization. However, producing high quality partitioning of propositions is not a simple task. Instead, in this method we take a "softer" approach that constructs an agglomerative hierarchical clustering (AHC) tree for the set of propositions.

The hierarchical clustering method utilizes the Cluto[6] tool to construct a hierarchical cluster tree based on the similarity matrix between propositions (using our similarity function). It then uses the resulting cluster tree structure to calculate the support of each

[4] http://tiny.cc/GoogleNews-vectors

[5] http://www.cs.waikato.ac.nz/ml/weka

[6] http://glaros.dtc.umn.edu/gkhome/software

```
Input: Selected answer $selectedAnswer$
Input: novelty assessment $Novelty$
Input: all propositions $Propositions$
Input: Proposition cluster tree $T$
/* Update propositions novelty */
foreach $p_a \in selectedAnswer$ do
    $n_{p_a} = findLeafNodeOf(p_a, T)$
    foreach $p \in Propositions$ do
        $n_p = findLeafNodeOf(p, T)$
        $d = distance(n_{p_a}, nearestCommonAncestor(n_{p_a}, n_p))$
        $Novelty[p] = Novelty[p] \cdot (1 - \frac{1}{2}^d)$
    end
end
```

Algorithm 2: Novelty update for all propositions after selecting an answer using hierarchical clustering tree

proposition and to penalize propositions that are similar to propositions in already selected answers. The answer score function of this method is defined as follows:

$$Score_{AHC}(a) = \frac{\sum_{p \in a} Novelty[p] \cdot Depth(p,T)}{\sqrt{|a|} \cdot \left(1 + \left(1 - \frac{\sum_{p \in a} Novelty[p]}{|a|}\right)\right)} \quad (2)$$

where $Depth(p,T)$ is the distance of node p from the root of the cluster tree, and $|a|$ refers to the number of propositions in a.

We note two properties of this function. First, in contrast to Equation 1, an answer score is based only on its own propositions. However, propositions that are further down in the (unbalanced) tree are considered more important, since they are similar to more propositions, therefore the nominal part captures the average importance of the answer's proposition where each proposition is weighted by its novelty. Second, an answer with non-novel propositions will be penalized since it has a larger denominator. Therefore, this function weighs novelty and importance against each other. Finally, answer scores are normalized by the answer's number of propositions $\sqrt{|a|}$ to allow fair comparison between answers of different length.

After selecting an answer a that maximizes $Score_{AHC}(a)$ to the ranked list, we penalize all propositions that are similar to the propositions in a. Propositions are penalized according to their distance from the selected answer's propositions in the cluster tree: the further away two propositions are, the less similar they are. Specifically, for each proposition p_a in the selected answer we start from its corresponding leaf node in the tree and climb towards the tree root. At each node n on the path we penalize each proposition p for which n is its nearest common ancestor with p_a, by degrading their $Novelty$ property: $Novelty[p] = Novelty[p] \cdot (1 - \frac{1}{2}^d)$, where d is their distance to p_a. This procedure is formalized in Algorithm 2.

Similarly to the previous method, the $Novelty$ of each proposition in the selected answer is set to 0, since its distance from itself is 0. Moreover, the more similar a proposition p is to those in the selected answer, *i.e.* the closer its location is to their locations in the tree, the more p will be penalized.

4. EXPERIMENTAL SETTINGS

To assess the performance of our proposed approach, we compare our two algorithm variants and several baselines on a manually annotated gold-standard test-set. Under this evaluation setting, a gold standard annotation of the aspects in each answer is given, following the methodology in the TREC novelty track [14]. Both our algorithms and the compared baselines are evaluated according to their ability to rank higher diverse answers that contain novel aspects with respect to higher ranked answers. We compared the ranking quality using the $\alpha NDCG$ [13] and *ERR-IA* [10] measures and novel measures we propose specifically for the CQA scenario.

We next detail the gold-standard dataset construction, the compared algorithms and the novel ranking measures.

4.1 Dataset Construction

Our manually constructed dataset[7] consists of a random sample of 110 questions from the Health top category in Yahoo Answers, each with at least 10 answers. For each sampled question the authors manually split each answer into propositions and annotated those that are relevant to the question. The relevant propositions in all answers were then manually aggregated into clusters, where each cluster represents a specific aspect mentioned in the answers. This aspect cluster is referred to as the gold-standard aspect mapping for the question. From this gold standard annotation the "importance" of each aspect is taken to be the size of its cluster. Additionally, the aspect distribution in each answer is straightforwardly induced. Overall, we analyzed 1426 answers and labeled 2775 relevant propositions referring to 838 different information aspects (7.6 aspects per question on average). Interestingly, relevant propositions cover (on average) only about 30% of the answers text.

We evaluated the performance of all tested algorithms using 5-fold cross validation. The training parts were used to train our supervised similarity function (see Subsection 3.3), and to tune the parameters of the baseline algorithms.

4.2 Tested Algorithms

We implemented six baseline algorithms in order to analyze the behavior of CQA answer ranking and to compare their performance to our proposed ranking approach. The first baseline is a simple random ranking of the answers (denoted *RandomRanker*). It serves as a lower bound for the performance of the algorithms under the evaluation measures we utilize.

The second baseline, denoted *Votes*, ranks the answers by the feedback they received from other users. Specifically, we subtract the down-votes (thumbs-down) from the up-votes (thumbs-up) that each answer received. This score was viewed in prior work as an implicit quality assessment measure and was used as ground truth for learning to rank answers by their quality [16].

The next two baselines address only the relevance aspect of ranking: *BM25* [34] and *ESA* (over Wikipedia) [20]. For BM25, the query is the question's text and the documents to be ranked are the different answers. Document frequency of the terms for BM25 was estimated over a collection of 16 million question/best-answer pairs. ESA was implemented as described in Section 3.3. We note that using ESA over CQA instead of Wikipedia did not improve the results.

We are not aware of prior work that addresses novelty or diversification in answer ranking within the CQA settings (*i.e.* ranking manually provided answers for a target question). Instead, we implemented two ranking approaches from related fields as baselines (Section 2.2). The first algorithm, denoted *LDARanker*, was proposed for diversifying product reviews [23]. We implemented this algorithm including the specific ngram partitioning of the texts [19]. We learned 1000 topics for the LDA model on a collection of 1 million CQA questions from the Health top category in Yahoo Answers, in order to match our test-set domain.

The second related baseline we implemented is the answer diversification algorithm of [1], denoted *WebAnswerRanker*. Given an input complex query, this algorithm attempts to diversify the set of

[7] This dataset will be publicly available.

retrieved answers from a large answer collection, showing significant improvement over prior work (see Section 2.2). We applied this algorithm in our settings, using TF-IDF for measuring question/answer similarity and n-gram similarity for answer/answer similarity (following experiments which showed that this is the best performing configuration on our data-set).

We tested two variants of our approach against the above baselines: a) *SimRanker*, which uses Equation 1 for answer selection and the corresponding novelty update (Subsection 3.4.1); and b) *HCRanker*, which uses the hierarchical clustering based answer selection (Equation 2) and AHC-based novelty update (Subsection 3.4.2). Both variants make use of the supervised similarity function detailed in Section 3.3.

4.3 Performance Measures

Several IR metrics were proposed that consider novelty/diversity on top of relevancy. For this experiment we utilize $\alpha NDCG$ [13, 12] and the topic-aware ERR (*ERR-IA*) [10]. These two metrics were designed for typical IR scenario in which many documents are irrelevant. They therefore aim at balancing between relevance and novelty. In the CQA scenario on the other hand, most answers are relevant. Therefore, we propose two additional metrics that directly assess the effort required by a user to scan the ranked answer list through the number of redundant aspects the user would encounter. The two metrics differ in their treatment of aspect importance: one is oblivious to it while the other focuses on it.

Novelty-focused evaluation metric

This metric assesses how efficient a given answer ranking is in covering the gold-standard aspects. For each recall point r (fraction of aspects covered) a cost function is first computed:

$$NoveltyCost(A, r) = \sum_{i=1}^{m(r)} \left(1 + \beta \cdot (1 - \frac{|NovelAspects(a_i)|}{|Aspects(a_i)|})\right)$$

$$m(r) = \min_{m} \frac{|\cup_{i=1}^{m} Aspects(a_i)|}{|\cup_{a \in A} Aspects(a)|} \geq r$$

A is the ranked answer list $\{a_i\}_1^{|A|}$, $m(r)$ is the minimal rank position for which the accumulative aspects in the rank reaches recall r in terms of aspect coverage, and β controls the effect of novelty. $Aspects(a_i)$ returns the gold-standard aspects that are mentioned in the answer a_i while $NovelAspects(a_i)$ returns only those aspects in a_i that are not covered by higher ranked answers. This cost function puts emphasize on the amount of new aspects an answer introduces by penalizing answers with repeated aspects. In the extreme, a novel answer which reveals only new aspects contributes 1 to the score while an answer with no novel aspects at all contributes $(1 + \beta)$ to the score. In our experiments, we set $\beta = 0.5$.

For each question we computed the minimal possible cost for the recall point r according to the above cost function by evaluating all the permutations over the answers set. We then normalize the cost of the ranked list A as follows:

$$minNoveltyCost(A, r) = \min_{A' \in \pi(A)} [NoveltyCost(A', r)]$$

$$NormNoveltyCost(A, r) = \frac{minNoveltyCost(A, r)}{NoveltyCost(A, r)}$$

where $\pi(A)$ is the set of all permutations of the answer list A. Note that while lower $NoveltyCost(A, r)$ values correspond to a better ranking, $NormNoveltyCost(A, r)$ exhibits how good is this ranking relative to the best possible ranking, and here higher values correspond to a better ranking.

The final metric score of a ranked list A is taken as the average of all normalized costs at recall points $\{0.1, 0.2, \dots, 1.0\}$:

$$NoveltyMetric(A) = \frac{1}{10} \sum_{r=0.1}^{r=1.0} NormNoveltyCost(A, r)$$

Support-focused evaluation metric

The novelty-focused metric views all aspects as equally important. Yet, as we discussed previously, some aspects are mentioned more than others in the answers and reflect a common recommendation or opinion. In this work we view such aspects as more "important", measuring it by the number of propositions that contain information about the aspect.

In the following proposed metric, important aspects contribute more to recall. For this purpose, recall is computed as the fraction of propositions belonging to the aspects covered by the list of answers. We follow a similar methodology as in the novelty-focused metric, by defining a cost function for a ranked list of answers A and a recall point r:

$$SupportCost(A, r) = \sum_{i=1}^{m(r)} \left(1 + \beta \cdot (1 - \frac{|Props(NovelAspects(a_i))|}{|Props(Aspects(a_i))|})\right)$$

$$m(r) = \min_{m} \frac{|\cup_{i=1}^{m} Props(Aspects(a_i))|}{|\cup_{a \in A} Props(Aspects(a))|} \geq r$$

where $Props()$ returns the list of gold-standard propositions corresponding to an input list of gold-standard aspects. The final metric is computed just like the novelty-focused metric by analogously defining *minSupportCost*, *NormSupportCost* and *SupportMetric*.

5. RESULTS

We compared the performance of all tested algorithms across the four metrics: $\alpha NDCG$, *ERR-IA*, $NoveltyMetric$ and $SupportMetric$. Table 2 summarizes the results. Statistical significance for the difference between our algorithms and the baselines is marked by the '+' sign for $p < 0.05$ using pairwise t-test. We note that the results in the table are macro averages over the parameters of each measure. Specifically, we averaged all α values in $\alpha NDCG$ metric on the range $[0, 1]$ with jumps of 0.25, and averaged all recall r values in novelty-based and support-based metrics, as discussed in Section 4.3. We used $\frac{15}{16}$ for relevant topic weight in *ERR-IA*. A more detailed analysis of the results, drawing the performance graph for each parameter range in each measure, is shown in Figures 1, 2 and 3 for $\alpha NDCG$, $NoveltyMetric$ and $SupportMetric$ respectively[8].

From the results we can see that the random baseline already places a rather high bar with respect to similar values in traditional IR tasks. This is especially true under $\alpha NDCG$ and *ERR-IA*, which supports our observation that most answers are relevant. Under our proposed metrics, which weigh novelty and diversification elements more, the random baseline is lower. This is also the reason why all systems perform similarly when considering only

[8] In all three figures our algorithms are plotted in solid lines, *Random* is plotted in dotted lines, LDA and $WebAnswer$ are plotted in mixed dashed and dots lines and *BM25*, *ESA* and *Votes* baselines are plotted in dashed lines.

Metric	$\alpha NDCG$	*ERR-IA*	Novelty	Support
RandomRanker	0.61	0.64	0.49	0.55
Votes	0.67	0.69	0.55	0.61
BM25	0.68	0.71	0.57	0.64
ESA	0.67	0.67	0.57	0.60
LDARanker	0.63	0.51	0.54	0.56
WebAnswerRanker	0.76	0.77	0.62	0.67
SimRanker	**0.80**$^+$	**0.81**	**0.68**$^+$	**0.71**
HCRanker	**0.79**	**0.81**	**0.67**$^+$	**0.71**

Table 2: Performance results of all algorithms across all metrics. $'+'$ signs mark statistically significant results between our algorithms and all the baselines.

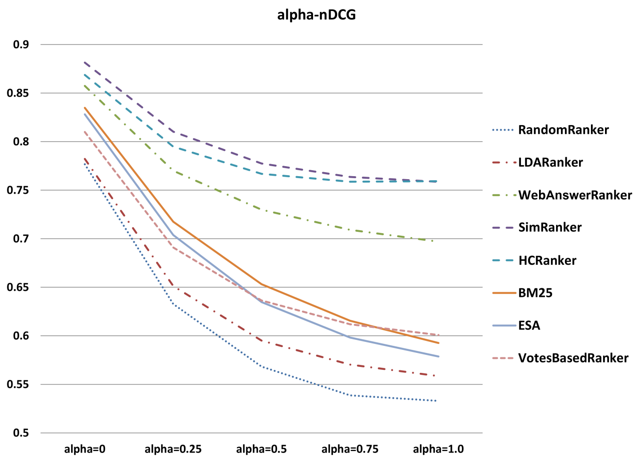

Figure 1: $\alpha NDCG$ results at different α values

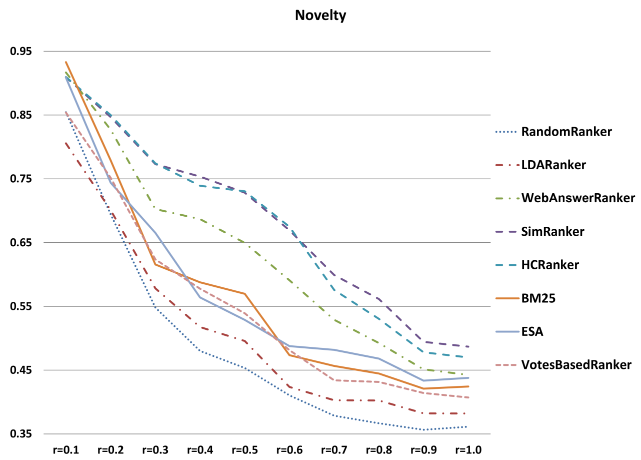

Figure 2: $NoveltyMetric$ results at different recall values

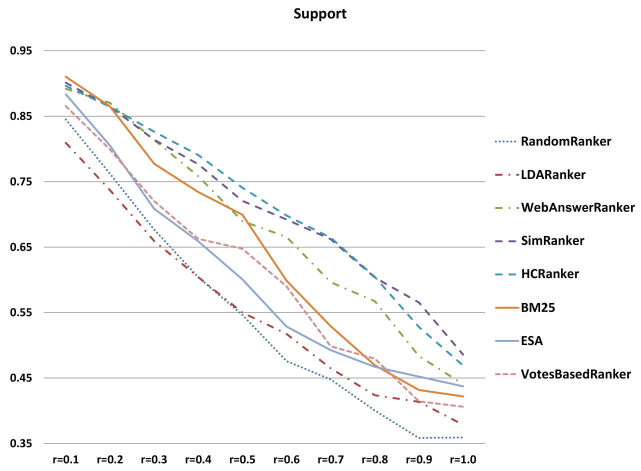

Figure 3: $SupportMetric$ results at different recall values

relevance, under $\alpha NDCG$ with $\alpha = 0$ (Fig. 1). Yet, even under such conservative evaluation, which ignores novelty, our algorithms perform better than the baselines since they introduce more diversity in the top results. When α is increased a clearer picture is revealed. The gap between our algorithms and the baselines keeps on increasing, showing the superior ability of our proposition-based approach to recognize novel aspects in answers. This is compared to the more "global" approaches taken by *LDARanker* and *WebAnswerRanker*. At the extreme, for $\alpha = 1$, *SimRanker* and *HCRanker* achieve a relative improvement over the best performing baseline of more than 11%. A similar trend is observed under the *ERR-IA* metric.

In terms of novelty-based ranking baselines, *WebAnswerRanker* is the best performing baseline. On the other hand, *LDARanker* performs even worse than the purely-relevance-based *BM25* ranker. This result emphasizes the need to model aspects at a fine-grained textual level. In our case, the LDA model was learned over all of the Health corpus and regards only high level topics. But, typically, each question page discusses one or two such high level semantic topics, and therefore *LDARanker* cannot distinguish between answers with respect to the aspects related to the high level topic. This also shows that answer ranking is quite different than product review ranking, in which it is common to find quite a few high level topics in reviews for any specific product and therefore to rank them at this high level view.

Interestingly, all algorithms seem to arrange well the most diverse answers at the top ($r = 0.1$ in Figs. 2 and 3). But this is a phenomenon of the CQA data, not of algorithmic capabilities. It is indicated by the fact that *RandomRanker* can perform at this recall point almost as well as the best performing algorithm. On the other hand, looking at the two figures we can observe the limitation of the greedy ranking scheme that all tested algorithms use. All are doing very well at low recall points, but their performance compared to an optimal solution drops quickly. At the extreme point of $r = 1$, when compared against optimal ranking, all algorithms perform poorly. This shows that early greedy decisions are far from optimal when looking at "the whole picture", and that in general, novelty-based answer ranking, at high recall level, is far from being solved.

While at the extreme recall points all algorithms perform similarly, the overall difference between our approach and the baseline is clearly evident: both our algorithms outperform all the baselines across all four metrics under all parameter settings. This is a strong indication of the capability of our approach to promote novel and diversified answers in a robust way. In general, we see that $SimRanker$ slightly outperforms $HCRanker$, but not in a statistically significant way. Diving more deeply, the main difference between our algorithms and the baselines is at mid recall ranges ($r \in [0.3, 0.9]$). With respect to $NoveltyMetric$ (Fig. 2), a notable gap is maintained at this range with the peak reached when

Question: My alarm clock wakes me at 7 o'clock every morning. why do always wake up tired, angry and irritable?

SimRanker answers:

1: You probably are not sleeping well during the night. You might have sleep apnea which often has the same affect. You should ask your doctor about it. Also, try cutting out all caffiene, sugar and spicy foods about 4 or 5 hours before bed. Do you watch tv at night. That can make matters worse for some. Try reading instead. And even though I am doing it right now, the computer is not good either. ... If this doesn't help, you might want to try setting your alarm for 6:30 instead. People sleep in cycles. You might be falling into an REM cycle right at 7 o'clock and those are the worst to wake up from. You might want to try resetting your internal clock by going to bed a little earlier. For me, I've had to resort to Ambien CR. It's a wonderful little pill...

2: Are you watching T.V. before you go to bed? Drinking sodas, coffee, eating sugar? ... Keep your body from working too hard ... and try some soothing techniques like warm milk and reading. You wake up feeling unrested because you're not getting enough sleep, and once you are asleep, it's no fun to wake up. Just try to go to bed earlier if possible and you should slowly rotate your circadian rhythm.

3: ... Try taking a hot relaxing bath just before bed, and maybe sit reading until you feel tired in bed. ... Also try going to bed a little earlier, and try not to oversleep. Another tip would be to not have large lie ins at the weekend as this will knock your rythym out of synch...

4: well you might not be getting enough sleep and your body needs more. ... so just take a shower and it makes it better

5: Try to stay awake for 24 hours, then go to bed early. ... Also dont eat too late at night, and dont drink caffeine in the evenings.

WebAnswerRanker answers:

1: Same as #2 of SimRanker | 2: Same as #3 of SimRanker | 3: Same as #4 of SimRanker

4: that makes two of us, not been up too long myself and i've got a mood like a thunder storm this morning

5: Maybe you stay up too late on here. Go to bed earlier.

Figure 4: Example of a question with 2 sets of top 5 answers ordered by $SimRanker$ and $WebAnswerRanker$. The propositions of each specific information aspect are colored in a separate color (human gold-standard annotation).

half of the novel aspects are presented ($r = 0.5$). This shows that our algorithm is nicely tuned for aspect novelty.

When promoting "important" aspects via their support (Fig. 3), a more interesting result is drawn. While both our algorithms are best performing, at a significant portion of the recall spectrum *WebAnswerRanker* and more interestingly *BM25* achieve comparable results, at least with $SimRanker$. A possible reason for the relatively good performance of *BM25*, which considers only relevance, is that relevant aspects are also well supported, since answerers feel "obliged" to address these common aspects. Similarly, *WebAnswerRanker*, whose underlying target is to find answers that mostly agree with other answers, provides good results. However, to achieve recall values beyond 0.6 the ranked answers need to cover also infrequent aspects, a task that *WebAnswerRanker* and more notably *BM25* find somewhat harder to perform.

In this experiment, the quality-based baseline *Votes* performed similarly to the relevance only models *ESA* and *BM25*. As expected, user votes contain good indication of answer relevance to the question, and *Votes* scores significantly above the *Random* baseline. However, *Votes* computes a quality value for each answer separately. It therefore cannot distinguish between high quality answers with few novel aspects or that are lacking aspect diversity from those that introduce novel perspectives. For instance, users may mark similar answers as high quality, because both are appealing, but they do not contribute additional information one on top of the other. Indeed, *Votes* did not perform as well as the novelty aware algorithms. This result emphasizes the difference between the tasks of quality-based answer ranking, which was the focus of prior work, and novelty-based answer ranking, which we introduce in this paper.

As a coda, we present an example in Fig. 4, which helps to highlight the performance differences between our approach (using *SimRanker*) and the best performing baseline (*WebAnswerRanker*). In the example, 10 relevant aspects appear in the answers that were provided for the question. *SimRanker* covers 7 of them already in the first answer, adds another 2 in the second one, and another one in the third answer. *WebAnswerRanker*, also covers 7 aspects in the first answer, adds another one in the second, and doesn't cover the 2 remaining in the first 5 at all. Moreover, its forth answer doesn't contain any relevant aspects at all. In terms of support, the first answer of *SimRanker* covers 74% of all aspects, while *WebAnswerRanker* covers 77%, but after the second answer *SimRanker* covers 89% of all aspects, while *WebAnswerRanker* covers 82%. Finally, *SimRanker* covers 100% within 3 answers while *WebAnswerRanker* stays at 82%. This illustrates the capability of *SimRanker* to balance between supported aspects and novel ones, compared to *WebAnswerRanker*, which mainly addresses support.

6. DISCUSSION

Though both our ranking algorithms are based on the same approach, some differences in their performance are worth noting. The similarity-based ranker $SimRanker$ directly uses the similarity function between propositions to compute answer support. This low level usage of the similarity function makes it vulnerable to similarity value distribution issues. Specifically, we have inspected our four unsupervised similarity functions and found that their values are rarely larger than 0.2 even for highly similar propositions. While the supervised measure enabled us to normalize and leverage the different similarity perspectives captured by each individual unsupervised measure, using each of these unsupervised measures alone within $SimRanker$ provided rather poor results on our testset. Therefore, this task would benefit from future exploration of similarity measures for very short texts that, on top of comparing their words, also leverage the immediate local context surrounding each text.

Our hierarchical clustering based ranker $HCRanker$ attempts to transform the continuous similarity score into a high-level relational semantic structure. This approach is motivated from viewing aspects as semantically separated discussion topics within answers, and it is inspired from clustering approaches in multi-document summarization [38, 30]. Yet, this transformation may suffer from information loss, since exact similarity values are not available anymore, as well as from the inherent difficulty in inducing high level semantic concepts from raw text. Specifically, observing the variance in performance, we noticed that this method is less stable and while for some questions it highly outperformed the other tested methods, in other cases its performance downgraded significantly. In the future we want to keep researching the construction and applicability of high level concept similarity graphs for text ranking in general and answer ranking specifically.

7. CONCLUSIONS

We introduced the task of novelty-based answer ranking for CQA question pages. We argued that under the CQA settings this task requires different approaches, since unlike standard document retrieval, most answers for a CQA question are relevant. We proposed a novel algorithm that looks at answer ranking as trying to cover as many aspects in a potential "summary" of all answers as possible with fewest answers. To this end, our algorithm regards syntactic propositions as the basic text units. It then computes the similarity between propositions between and within answers for assessing diversification, novelty and importance quality. We also computed each proposition's similarity to the question for assessing relevance. Finally, answers are greedily ranked based on the amount of novel propositions each answer contains, as well as their importance within all answers, taking into account their relevance to the question. To measure the performance of our algorithm we compared it to prior work that considered novelty in ranking. Under a gold standard manual evaluation, our algorithm significantly outperformed the compared works.

We would like to further investigate the importance of novelty in CQA. Specifically, which types of questions would benefit more from such ranking. In addition, we would like to incorporate novelty elements for measuring user reputation, as an additional quality notion of the answerer. Finally, we want to explore how novelty assessment can be combined with other answer quality signals such as user reputation, temporal features and writing style in order to analyze which are more important to viewers of a CQA page.

8. REFERENCES

[1] P. Achananuparp, X. Hu, T. He, C. C. Yang, Y. An, and L. Guo. Answer diversification for complex question answering on the web. In *PAKDD*, 2010.

[2] P. Achananuparp, C. C. Yang, and X. Chen. Using negative voting to diversify answers in non-factoid question answering. In *CIKM*, 2009.

[3] L. A. Adamic, J. Zhang, E. Bakshy, and M. S. Ackerman. Knowledge sharing and yahoo answers: everyone knows something. In *WWW*, 2008.

[4] R. Agrawal, S. Gollapudi, A. Halverson, and S. Ieong. Diversifying search results. In *WSDM*, 2009.

[5] J. Allan, C. Wade, and A. Bolivar. Retrieval and novelty detection at the sentence level. In *SIGIR*, 2003.

[6] J. Bian, Y. Liu, E. Agichtein, and H. Zha. Finding the right facts in the crowd: Factoid question answering over social media. In *WWW*, 2008.

[7] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, Mar. 2003.

[8] J. Carbonell and J. Goldstein. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *SIGIR*, 1998.

[9] W. Chan, X. Zhou, W. Wang, and T.-S. Chua. Community answer summarization for multi-sentence question with group l1 regularization. In *ACL*, 2012.

[10] O. Chapelle, D. Metlzer, Y. Zhang, and P. Grinspan. Expected reciprocal rank for graded relevance. In *CIKM*, 2009.

[11] J. D. Choi and M. Palmer. Getting the most out of transition-based dependency parsing. In *ACL*, 2011.

[12] C. L. Clarke, N. Craswell, I. Soboroff, and A. Ashkan. A comparative analysis of cascade measures for novelty and diversity. In *WSDM*, 2011.

[13] C. L. Clarke, M. Kolla, G. V. Cormack, O. Vechtomova, A. Ashkan, S. Büttcher, and I. MacKinnon. Novelty and diversity in information retrieval evaluation. In *SIGIR*, 2008.

[14] C. L. A. Clarke, N. Craswell, and I. Soboroff. Overview of the TREC 2009 web track. In *TREC*, 2009.

[15] W. B. Croft and V. Dang. Term level search result diversification. In *SIGIR*, 2013.

[16] D. H. Dalip, M. A. Gonçalves, M. Cristo, and P. Calado. Exploiting user feedback to learn to rank answers in q&a forums: A case study with stack overflow. In *SIGIR*, 2013.

[17] V. Dang and W. B. Croft. Diversity by proportionality: An election-based approach to search result diversification. In *SIGIR*, 2012.

[18] H. Daumé III and D. Marcu. Bayesian query-focused summarization. In *ACL*, 2006.

[19] S. Deligne and F. Bimbot. Language modeling by variable length sequences: Theoretical formulation and evaluation of multigrams. In *ICASSP*, 1995.

[20] E. Gabrilovich and S. Markovitch. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *IJCAI*, 2007.

[21] J. Jeon, W. B. Croft, J. H. Lee, and S. Park. A framework to predict the quality of answers with non-textual features. In *SIGIR*, 2006.

[22] P. Jurczyk and E. Agichtein. Discovering authorities in question answer communities by using link analysis. In *CIKM*, 2007.

[23] R. Krestel and N. Dokoohaki. Diversifying product review rankings: Getting the full picture. In *WI*, 2011.

[24] X. Li and W. B. Croft. Improving novelty detection for general topics using sentence level information patterns. In *CIKM*, 2006.

[25] C.-Y. Lin and E. Hovy. From single to multi-document summarization: A prototype system and its evaluation. In *ACL*, 2002.

[26] Q. Liu, E. Agichtein, G. Dror, Y. Maarek, and I. Szpektor. When web search fails, searchers become askers: Understanding the transition. In *SIGIR*, 2012.

[27] Y. Liu, S. Li, Y. Cao, C.-Y. Lin, D. Han, and Y. Yu. Understanding and summarizing answers in community-based question answering services. In *COLING*, 2008.

[28] I. Mani. Multi-document summarization by graph search and matching. In *AAAI*, 1997.

[29] T. Mikolov, K. Chen, G. S. Corrado, and J. Dean. Efficient estimation of word representations in vector space, 2013.

[30] A. Nenkova and K. McKeown. A survey of text summarization techniques. In *Mining Text Data*, pages 43–76. Springer US, 2012.

[31] V. Pande, T. Mukherjee, and V. Varma. Summarizing answers for community question answer services. In I. Gurevych, C. Biemann, and T. Zesch, editors, *GSCL*, volume 8105 of *Lecture Notes in Computer Science*, pages 151–161. Springer, 2013.

[32] D. R. Radev and K. R. McKeown. Generating natural language summaries from multiple on-line sources. *Comput. Linguist.*, 24(3):470–500, 1998.

[33] D. Rafiei, K. Bharat, and A. Shukla. Diversifying web search results. In *WWW*, 2010.

[34] S. Robertson and H. Zaragoza. *The probabilistic relevance framework: BM25 and beyond*. Now Publishers Inc, 2009.

[35] R. L. Santos, C. Macdonald, and I. Ounis. Exploiting query reformulations for web search result diversification. In *WWW*, 2010.

[36] C. Shah and J. Pomerantz. Evaluating and predicting answer quality in community qa. In *SIGIR*, 2010.

[37] A. Shtok, G. Dror, Y. Maarek, and I. Szpektor. Learning from the past: Answering new questions with past answers. In *WWW*, 2012.

[38] A. Siddharthan, A. Nenkova, and K. McKeown. Syntactic simplification for improving content selection in multi-document summarization. In *COLING*, 2004.

[39] I. Soboroff and D. Harman. Novelty detection: The trec experience. In *HLT*, 2005.

[40] M. A. Suryanto, E. P. Lim, A. Sun, and R. H. L. Chiang. Quality-aware collaborative question answering: Methods and evaluation. In *WSDM*, 2009.

[41] J. Tang, L. Yao, and D. Chen. Multi-topic based query-oriented summarization. In *SDM*, 2009.

[42] X. Tu, X.-J. Wang, D. Feng, and L. Zhang. Ranking community answers via analogical reasoning. In *WWW*, 2009.

[43] D. Vallet and P. Castells. Personalized diversification of search results. In *SIGIR*, 2012.

[44] C. X. Zhai, W. W. Cohen, and J. Lafferty. Beyond independent relevance: Methods and evaluation metrics for subtopic retrieval. In *SIGIR*, 2003.

[45] Z.-M. Zhou, M. Lan, Z.-Y. Niu, and Y. Lu. Exploiting user profile information for answer ranking in cqa. In *WWW*, 2012.