Metadata-Aware Measures for Answer Summarization in Community Question Answering

Mattia Tomasoni *
Dept. of Information Technology
Uppsala University, Uppsala, Sweden
mattia.tomasoni.8371@student.uu.se

Minlie Huang
Dept. Computer Science and Technology
Tsinghua University, Beijing 100084, China
aihuang@tsinghua.edu.cn

Abstract

This paper presents a framework for automatically processing information coming from community Question Answering (cQA) portals with the purpose of generating a trustful, complete, relevant and succinct summary in response to a question. We exploit the metadata intrinsically present in User Generated Content (UGC) to bias automatic multi-document summarization techniques toward high quality information. We adopt a representation of concepts alternative to n-grams and propose two concept-scoring functions based on semantic overlap. Experimental results on data drawn from Yahoo! Answers demonstrate the effectiveness of our method in terms of ROUGE scores. We show that the information contained in the best answers voted by users of cQA portals can be successfully complemented by our method.

1 Introduction

Community Question Answering (cQA) portals are an example of Social Media where the information need of a user is expressed in the form of a question for which a best answer is picked among the ones generated by other users. cQA websites are becoming an increasingly popular complement to search engines: overnight, a user can expect a human-crafted, natural language answer tailored to her specific needs. We have to be aware, though, that User Generated Content (UGC) is often redundant, noisy and untrustworthy (Jeon et al., 2006; Wang et al., 2009b; Suryanto et al., 2009). Interestingly, a great amount of information is embedded in the metadata generated as a byproduct of users' action and interaction on Social Media. Much valuable information is contained in answers other than the chosen best one (Liu et al., 2008). Our work aims to show that such information can be successfully extracted and made available by exploiting metadata to distill cQA content. To this end, we casted the problem to an instance of the query-biased multi-document summarization task, where the question was seen as a query and the available answers as documents to be summarized. We mapped each characteristic that an ideal answer should present to a measurable property that we wished the final summary could exhibit:

- Quality to assess trustfulness in the source,
- Coverage to ensure completeness of the information presented,
- Relevance to keep focused on the user's information need and
- Novelty to avoid redundancy.

Quality of the information was assessed via Machine Learning (ML) techniques under best answer supervision in a vector space consisting of linguistic and statistical features about the answers and their authors. Coverage was estimated by semantic comparison with the knowledge space of a corpus of answers to similar questions which had been retrieved through the Yahoo! Answers API [1]. Relevance was computed as information overlap between an answer and its question, while Novelty was calculated as inverse overlap with all other answers to the same question. A score was assigned to each concept in an answer according to

The research was conducted while the first author was visiting Tsinghua University.

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

Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, pages 760–769,
Uppsala, Sweden, 11-16 July 2010.

the above properties. A score-maximizing summary under a maximum coverage model was then computed by solving an associated Integer Linear Programming problem (Gillick and Favre, 2009; McDonald, 2007). We chose to express concepts in the form of Basic Elements (BE), a semantic unit developed at ISI[2] and modeled semantic overlap as intersection in the equivalence classes of two concepts (formal definitions will be given in section 2.3).

The objective of our work was to present what we believe is a valuable conceptual framework; more advance machine learning and summarization techniques would most likely improve the performances.

The remaining of this paper is organized as follows. In the next section Quality, Coverage, Relevance and Novelty measures are presented; we explain how they were calculated and combined to generate a final summary of all answers to a question. Experiments are illustrated in Section 3, where we give evidence of the effectiveness of our method. We list related work in Section 5, discuss possible alternative approaches in Section 4 and provide our conclusions in Section 6.

2 The summarization framework

2.1 Quality as a ranking problem

Quality assessing of information available on Social Media had been studied before mainly as a binary classification problem with the objective of detecting low quality content. We, on the other hand, treated it as a ranking problem and made use of quality estimates with the novel intent of successfully combining information from sources with different levels of trustfulness and writing ability. This is crucial when manipulating UGC, which is known to be subject to particularly great variance in credibility (Jeon et al., 2006; Wang et al., 2009b; Suryanto et al., 2009) and may be poorly written.

An answer a was given along with information about the user u that authored it, the set TA^q (Total Answers) of all answers to the same question q and the set TA^u of all answers by the same user. Making use of results available in the literature (Agichtein et al., 2008) [3], we designed a Quality feature space to capture the following syntactic, behavioral and statistical properties:

- ϑ, length of answer a
- ς, number of non-stopwords in a with a corpus frequency larger than n (set to 5 in our experiments)
- ϖ, points awarded to user u according to the Yahoo! Answers' points system
- ϱ, ratio of best answers posted by user u

The features mentioned above determined a space Ψ; An answer a, in such feature space, assumed the vectorial form:

$$\Psi^a = (\ \vartheta,\ \varsigma,\ \varpi,\ \varrho\ )$$

Following the intuition that chosen best answers ($a^\star$) carry high quality information, we used supervised ML techniques to predict the probability of a to have been selected as a best answer $a^\star$. We trained a Linear Regression classifier to learn the weight vector $W = (w_1, w_2, w_3, w_4)$ that would combine the above feature. Supervision was given in the form of a training set Tr^Q of labeled pairs defined as:

$$Tr^Q = \{\langle\ \Psi^a,\ isbest^a\ \rangle\}$$

$isbest^a$ was a boolean label indicating whether a was an $a^\star$ answer; the training set size was determined experimentally and will be discussed in Section 3.2. Although the value of $isbest^a$ was known for all answers, the output of the classifier offered us a real-valued prediction that could be interpreted as a quality score $Q(\Psi^a)$:

$$\begin{aligned} Q(\Psi^a) &\approx P(\ isbest^a = 1 \mid a, u, TA^u,\ ) \\ &\approx P(\ isbest^a = 1 \mid \Psi^a\ ) \\ &= W^T \cdot \Psi^a \end{aligned} \quad (1)$$

The Quality measure for an answer a was approximated by the probability of such answer to be a best answer ($isbest^a = 1$) with respect to its author u and the sets TA^u and TA^q. It was calculated as dot product between the learned weight vector W and the feature vector for answer Ψ^a.

Our decision to proceed in an unsupervised direction came from the consideration that any use of external human annotation would have made it impracticable to build an actual system on larger scale. An alternative, completely unsupervised approach to quality detection that has not undergone experimental analysis is discussed in Section 4.

[2] Information Sciences Institute, University of Southern California, http://www.isi.edu

[3] A long list of features is proposed; training a classifier on all of them would no doubt increase the performances.

2.2 Bag-of-BEs and semantic overlap

The properties that remain to be discussed, namely Coverage, Relevance and Novelty, are measures of semantic overlap between concepts; a concept is the smallest unit of meaning in a portion of written text. To represent sentences and answers we adopted an alternative approach to classical n-grams that could be defined bag-of-BEs. a BE is "a head|modifier|relation triple representation of a document developed at ISI" (Zhou et al., 2006). BEs are a strong theoretical instrument to tackle the ambiguity inherent in natural language that find successful practical applications in real-world query-based summarization systems. Different from n-grams, they are variant in length and depend on parsing techniques, named entity detection, part-of-speech tagging and resolution of syntactic forms such as hyponyms, pronouns, pertainyms, abbreviation and synonyms. To each BE is associated a class of semantically equivalent BEs as result of what is called a transformation of the original BE; the mentioned class uniquely defines the concept. What seemed to us most remarkable is that this makes the concept context-dependent. A sentence is defined as a set of concepts and an answer is defined as the union between the sets that represent its sentences.

The rest of this section gives formal definition of our model of concept representation and semantic overlap. From a set-theoretical point of view, each concepts c was uniquely associated with a set $E^c = \{c_1, c_2 \dots c_m\}$ such that:

$$\forall i, j \quad (c_i \approx^L c) \wedge (c_i \not\equiv c) \wedge (c_i \not\equiv c_j)$$

In our model, the "$\equiv$" relation indicated syntactic equivalence (exact pattern matching), while the "$\approx^L$" relation represented semantic equivalence under the convention of some language L (two concepts having the same meaning). E^c was defined as the set of semantically equivalent concepts to c, called its equivalence class; each concept c_i in E^c carried the same meaning ($\approx^L$) of concept c without being syntactically identical ($\equiv$); furthermore, no two concepts i and j in the same equivalence class were identical.

"Climbing a tree to escape a black bear is pointless because *they can climb* very well."

BE = they|climb
E^c = {climb|bears, bear|go_up, climbing|animals, climber|instincts, trees|go_up, claws|climb...}

Given two concepts c and k:

$$c \bowtie k \begin{cases} c \equiv k & \text{or} \\ E^c \cap E^k \neq \emptyset \end{cases}$$

We defined semantic overlap as occurring between c and k if they were syntactically identical or if their equivalence classes E^c and E^k had at least one element in common. In fact, given the above definition of equivalence class and the transitivity of "$\equiv$" relation, we have that if the equivalence classes of two concepts are not disjoint, then they must bare the same meaning under the convention of some language L; in that case we said that c semantically overlapped k. It is worth noting that relation "$\bowtie$" is symmetric, transitive and reflexive; as a consequence all concepts with the same meaning are part of a same equivalence class. BE and equivalence class extraction were performed by modifying the behavior of the BEwT-E-0.3 framework [4]. The framework itself is responsible for the operative definition of the "$\approx^L$" relation and the creation of the equivalence classes.

2.3 Coverage via concept importance

In the scenario we proposed, the user's information need is addressed in the form of a unique, summarized answer; information that is left out of the final summary will simply be unavailable. This raises the concern of completeness: besides ensuring that the information provided could be trusted, we wanted to guarantee that the posed question was being answered thoroughly. We adopted the general definition of Coverage as the portion of relevant information about a certain subject that is contained in a document (Swaminathan et al., 2009). We proceeded by treating each answer to a question q as a separate document and we retrieved through the Yahoo! Answers API a set TK^q (Total Knowledge) of 50 answers [5] to questions similar to q: the knowledge space of TK^q was chosen to approximate the entire knowledge space related to the queried question q. We calculated Coverage as a function of the portion of answers in TK^q that presented semantic overlap with a.

[4] The authors can be contacted regarding the possibility of sharing the code of the modified version. Original version available from `http://www.isi.edu/publications/licensed-sw/BE/index.html`.

[5] such limit was imposed by the current version of the API. Experiments with a greater corpus should be carried out in the future.

$$C(a,q) = \sum_{c_i \in a} \gamma(c_i) \cdot tf(c_i, a) \quad (2)$$

The Coverage measure for an answer a was calculated as the sum of term frequency $tf(c_i, a)$ for concepts in the answer itself, weighted by a concept importance function, $\gamma(c_i)$, for concepts in the total knowledge space TK^q. $\gamma(c)$ was defined as follows:

$$\gamma(c) = \frac{|TK^{q,c}|}{|TK^q|} \cdot \log_2 \frac{|TK^q|}{|TK^{q,c}|} \quad (3)$$

where $TK^{q,c} = \{d \in TK^q : \exists k \in d,\ k \bowtie c\}$

The function $\gamma(c)$ of concept c was calculated as a function of the cardinality of set TK^q and set $TK^{q,c}$, which was the subset of all those answers d that contained at least one concept k which presented semantical overlap with c itself. A similar idea of knowledge space coverage is addressed by Swaminathan et al. (2009), from which formulas (2) and (3) were derived.

A sensible alternative would be to estimate Coverage at the sentence level.

2.4 Relevance and Novelty via $\bowtie$ relation

To this point, we have addressed matters of trustfulness and completeness. Another widely shared concern for Information Retrieval systems is Relevance to the query. We calculated relevance by computing the semantic overlap between concepts in the answers and the question. Intuitively, we reward concepts that express meaning that could be found in the question to be answered.

$$R(c,q) = \frac{|q^c|}{|q|} \quad (4)$$

where $q^c = \{k \in q : k \bowtie c\}$

The Relevance measure $R(c,q)$ of a concept c with respect to a question q was calculated as the ratio of the cardinality of set q^c (containing all concepts in q that semantically overlapped with c) normalized by the total number of concepts in q.

Another property we found desirable, was to minimize redundancy of information in the final summary. Since all elements in TA^q (the set of all answers to q) would be used for the final summary, we positively rewarded concepts that were expressing novel meanings.

$$N(c,q) = 1 - \frac{|TA^{q,c}|}{|TA^q|} \quad (5)$$

where $TA^{q,c} = \{d \in TA^q : \exists k \in d,\ k \bowtie c\}$

The Novelty measure $N(c,q)$ of a concept c with respect to a question q was calculated as the ratio of the cardinality of set $TA^{q,c}$ over the cardinality of set TA^q; $TA^{q,c}$ was the subset of all those answers d in TA^q that contained at least one concept k which presented semantical overlap with c.

2.5 The concept scoring functions

We have now determined how to calculate the scores for each property in formulas (1), (2), (4) and (5); under the assumption that the Quality and Coverage of a concept are the same of its answer, every concept c part of an answer a to some question q, could be assigned a score vector as follows:

$$\Phi^c = (\ Q(\Psi^a),\ C(a,q),\ R(c,q),\ N(c,q)\ )$$

What we needed at this point was a function S of the above vector which would assign a higher score to concepts most worthy of being included in the final summary. Our intuition was that since Quality, Coverage, Novelty and Relevance were all virtues properties, S needed to be monotonically increasing with respect to all its dimensions. We designed two such functions. Function (6), which multiplied the scores, was based on the probabilistic interpretation of each score as an independent event. Further empirical considerations, brought us to later introduce a logarithmic component that would discourage inclusion of sentences shorter then a threshold t (a reasonable choice for this parameter is a value around 20). The score for concept c appearing in sentence s^c was calculated as:

$$S^{\Pi}(c) = \prod_{i=1}^{4} (\Phi_i^c) \cdot \log_t(length(s^c)) \quad (6)$$

A second approach that made use of human annotation to learn a vector of weights $V = (v_1, v_2, v_3, v_4)$ that linearly combined the scores was investigated. Analogously to what had been done with scoring function (6), the Φ space was augmented with a dimension representing the length of the answer.

$$S^{\Sigma}(c) = \sum_{i=1}^{4} (\Phi_i^c \cdot v_i) + length(s^c) \cdot v_5 \quad (7)$$

In order to learn the weight vector V that would combine the above scores, we asked three human annotators to generate question-biased extractive summaries based on all answers available for a certain question. We trained a Linear Regression

classifier with a set Tr^S of labeled pairs defined as:

$$Tr^S = \{\langle\, (\Phi^c,\ length(s^c)),\ include^c \,\rangle\}$$

$include^c$ was a boolean label that indicated whether s^c, the sentence containing c, had been included in the human-generated summary; $length(s^c)$ indicated the length of sentence s^c. Questions and relative answers for the generation of human summaries were taken from the "filtered dataset" described in Section 3.1.

The concept score for the same BE in two separate answers is very likely to be different because it belongs to answers with their own Quality and Coverage values: this only makes the scoring function context-dependent and does not interfere with the calculation the Coverage, Relevance and Novelty measures, which are based on information overlap and will regard two BEs with overlapping equivalence classes as being the same, regardless of their score being different.

2.6 Quality constrained summarization

The previous sections showed how we quantitatively determined which concepts were more worthy of becoming part of the final machine summary M. The final step was to generate the summary itself by automatically selecting sentences under a length constraint. Choosing this constraint carefully demonstrated to be of crucial importance during the experimental phase. We again opted for a metadata-driven approach and designed the length constraint as a function of the lengths of all answers to q (TA^q) weighted by the respective Quality measures:

$$length^M = \sum_{a \in TA^q} length(a) \cdot Q(\Psi^a) \quad (8)$$

The intuition was that the longer and the more trustworthy answers to a question were, the more space was reasonable to allocate for information in the final, machine summarized answer M.

M was generated so as to maximize the scores of the concepts it included. This was done under a maximum coverage model by solving the following Integer Linear Programming problem:

$$\text{maximize:} \quad \sum_i S(c_i) \cdot x_i \quad (9)$$

$$\text{subject to:} \quad \sum_j length(j) \cdot s_j \leq length^M$$

$$\sum_j y_j \cdot occ_{ij} \geq x_i \quad \forall i \quad (10)$$

$$occ_{ij},\ x_i,\ y_j \in \{0,1\} \quad \forall i,j$$

$$occ_{ij} = 1 \text{ if } c_i \in s_j, \quad \forall i,j$$

$$x_i = 1 \text{ if } c_i \in M, \quad \forall i$$

$$y_j = 1 \text{ if } s_j \in M, \quad \forall j$$

In the above program, M is the set of selected sentences: $M = \{s_j : y_j = 1, \quad \forall j\}$. The integer variables x_i and y_j were equals to one if the corresponding concept c_i and sentence s_j were included in M. Similarly occ_{ij} was equal to one if concept c_i was contained in sentence s_j. We maximized the sum of scores $S(c_i)$ (for S equals to S^Π or S^Σ) for each concept c_i in the final summary M. We did so under the constraint that the total length of all sentences s_j included in M must be less than the total expected length of the summary itself. In addition, we imposed a consistency constraint: if a concept c_i was included in M, then at least one sentence s_j that contained the concept must also be selected (constraint (10)). The described optimization problem was solved using lp_solve [6].

We conclude with an empirical side note: since solving the above can be computationally very demanding for large number of concepts, we found performance-wise very fruitful to skim about one fourth of the concepts with lowest scores.

3 Experiments

3.1 Datasets and filters

The initial dataset was composed of 216,563 questions and 1,982,006 answers written by 171,676 user in 100 categories from the Yahoo! Answers portal[7]. We will refer to this dataset as the "unfiltered version". The metadata described in section 2.1 was extracted and normalized; quality experiments (Section 3.2) were then conducted. The unfiltered version was later reduced to 89,814 question-answer pairs that showed statistical and linguistic properties which made them particularly adequate for our purpose. In particular, trivial, factoid and encyclopedia-answerable questions were

[6] the version used was lp_solve 5.5, available at `http://lpsolve.sourceforge.net/5.5`

[7] The reader is encouraged to contact the authors regarding the availability of data and filters described in this Section.

removed by applying a series of patterns for the identification of complex questions. The work by Liu et al. (2008) indicates some categories of questions that are particularly suitable for summarization, but due to the lack of high-performing question classifiers we resorted to human-crafted question patterns. Some pattern examples are the following:

- {Why,What is the reason} [...]
- How {to,do,does,did} [...]
- How {is,are,were,was,will} [...]
- How {could,can,would,should} [...]

We also removed questions that showed statistical values outside of convenient ranges: the number of answers, length of the longest answer and length of the sum of all answers (both absolute and normalized) were taken in consideration. In particular we discarded questions with the following characteristics:

- there were less than three answers [8]
- the longest answer was over 400 words (likely a copy-and-paste)
- the sum of the length of all answers outside of the $(100,\ 1000)$ words interval
- the average length of answers was outside of the $(50,\ 300)$ words interval

At this point a second version of the dataset was created to evaluate the summarization performance under scoring function (6) and (7); it was generated by manually selecting questions that arouse subjective, human interest from the previous 89,814 question-answer pairs. The dataset size was thus reduced to 358 answers to 100 questions that were manually summarized (refer to Section 3.3). From now on we will refer to this second version of the dataset as the "filtered version".

3.2 Quality assessing

In Section 2.1 we claimed to be able to identify high quality content. To demonstrate it, we conducted a set of experiments on the original unfiltered dataset to establish whether the feature space Ψ was powerful enough to capture the quality of answers; our specific objective was to estimate the amount of training examples needed to successfully train a classifier for the quality assessing task. The Linear Regression[9] method was chosen to determine the probability $Q(\Psi^a)$ of a to be a best answer to q; as explained in Section 2.1, those probabilities were interpreted as quality estimates. The evaluation of the classifier's output was based on the observation that given the set of all answers TA^q relative to q and the best answer $a^\star$, a successfully trained classifier should be able to rank $a^\star$ ahead of all other answers to the same question. More precisely, we defined Precision as follows:

$$\frac{|\{q \in Tr^Q :\ \forall a \in TA^q,\ Q(\Psi^{a^\star}) > Q(\Psi^a)\}|}{|Tr^Q|}$$

where the numerator was the number of questions for which the classifier was able to correctly rank $a^\star$ by giving it the highest quality estimate in TA^q and the denominator was the total number of examples in the training set Tr^Q. Figure 1 shows the precision values (Y-axis) in identifying best answers as the size of Tr^Q increases (X-axis). The experiment started from a training set of size 100 and was repeated adding 300 examples at a time until precision started decreasing. With each increase in training set size, the experiment was repeated ten times and average precision values were calculated. In all runs, training examples were picked randomly from the unfiltered dataset described in Section 3.1; for details on Tr^Q see Section 2.1. A training set of 12,000 examples was chosen for the summarization experiments.

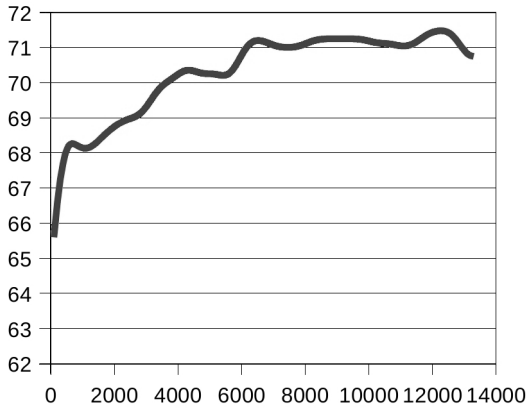

Figure 1: Precision values (Y-axis) in detecting best answers $a^\star$ with increasing training set size (X-axis) for a Linear Regression classifier on the unfiltered dataset.

[8] Being too easy to summarize or not requiring any summarization at all, those questions wouldn't constitute an valuable test of the system's ability to extract information.

[9] Performed with Weka 3.7.0 available at `http://www.cs.waikato.ac.nz/~ml/weka`

System	$a^\star$ (baseline)	S^Σ	S^Π
ROUGE-1_R	51.7%	67.3%	67.4%
ROUGE-1_P	62.2%	54.0%	71.2%
ROUGE-1_F	52.9%	59.3%	66.1%
ROUGE-2_R	40.5%	52.2%	58.8%
ROUGE-2_P	49.0%	41.4%	63.1%
ROUGE-2_F	41.6%	45.9%	57.9%
ROUGE-L_R	50.3%	65.1%	66.3%
ROUGE-L_P	60.5%	52.3%	70.7%
ROUGE-L_F	51.5%	57.3%	65.1%

Table 1: Summarization Evaluation on filtered dataset (refer to Section 3.1 for details). ROUGE-L, ROUGE-1 and ROUGE-2 are presented; for each, Recall (R), Precision (P) and F-1 score (F) are given.

3.3 Evaluating answer summaries

The objective of our work was to summarize answers from cQA portals. Two systems were designed: Table 1 shows the performances using function S^Σ (see equation (7)), and function S^Π (see equation (6)). The chosen best answer $a^\star$ was used as a baseline. We calculated ROUGE-1 and ROUGE-2 scores[10] against human annotation on the filtered version of the dataset presented in Section 3.1. The filtered dataset consisted of 358 answers to 100 questions. For each questions q, three annotators were asked to produce an extractive summary of the information contained in TA^q by selecting sentences subject to a fixed length limit of 250 words. The annotation resulted in 300 summaries (larger-scale annotation is still ongoing). For the S^Σ system, 200 of the 300 generated summaries were used for training and the remaining were used for testing (see the definition of Tr^S Section 2.5). Cross-validation was conducted. For the S^Π system, which required no training, all of the 300 summaries were used as the test set.

S^Σ outperformed the baseline in Recall (R) but not in Precision (P); nevertheless, the combined F-1 score (F) was sensibly higher (around 5 points percentile). On the other hand, our S^Π system showed very consistent improvements of an order of 10 to 15 points percentile over the baseline on all measures; we would like to draw attention on the fact that even if Precision scores are higher, it is on Recall scores that greater improvements were achieved. This, together with the results obtained by S^Σ, suggest performances could benefit from the enforcement of a more stringent length constraint than the one proposed in (8). Further potential improvements on S^Σ could be obtained by choosing a classifier able to learn a more expressive underlying function.

[10]Available at `http://berouge.com/default.aspx`

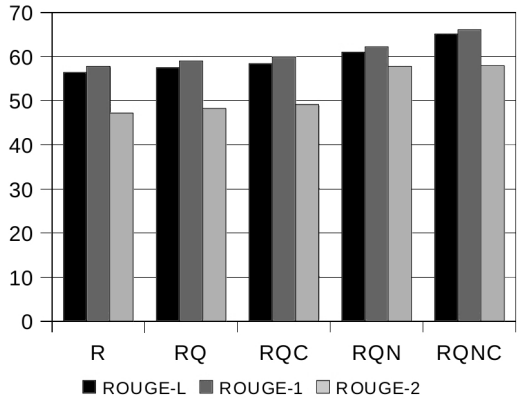

Figure 2: Increase in ROUGE-L, ROUGE-1 and ROUGE-2 performances of the S^Π system as more measures are taken in consideration in the scoring function, starting from Relevance alone (R) to the complete system (RQNC). F-1 scores are given.

In order to determine what influence the single measures had on the overall performance, we conducted a final experiment on the filtered dataset to evaluate (the S^Π scoring function was used). The evaluation was conducted in terms of F-1 scores of ROUGE-L, ROUGE-1 and ROUGE-2. First only Relevance was tested (R) and subsequently Quality was added (RQ); then, in turn, Coverage (RQC) and Novelty (RQN); Finally the complete system taking all measures in consideration (RQNC). Results are shown in Figure 2. In general performances increase smoothly with the exception of ROUGE-2 score, which seems to be particularly sensitive to Novelty: no matter what combination of measures is used (R alone, RQ, RQC), changes in ROUGE-2 score remain under one point percentile. Once Novelty is added, performances rise abruptly to the system's highest. A summary example, along with the question and the best answer, is presented in Table 2.

4 Discussion and Future Directions

We conclude by discussing a few alternatives to the approaches we presented. The $length^M$ constraint for the final summary (Section 2.6), could have been determined by making use of external knowledge such as TK^q: since TK^q represents

HOW TO PROTECT YOURSELF FROM A BEAR? http://answers.yahoo.com/question/index?qid=20060818062414AA7VldB
BEST ANSWER Great question. I have done alot of trekking through California, Montana and Wyoming and have met Black bears (which are quite dinky and placid but can go nuts if they have babies), and have been half an hour away from (allegedly) the mother of all grizzley s whilst on a trail through Glacier National park - so some other trekkerers told me... What the park wardens say is SING, SHOUT, MAKE NOISE...do it loudly, let them know you are there..they will get out of the way, it is a surprised bear wot will go mental and rip your little legs off..No fun permission: anything that will confuse them and stop them in their tracks...I have been told be an native american buddy that to keep a bottle of perfume in your pocket...throw it at the ground near your feet and make the place stink: they have good noses, them bears, and a mega concentrated dose of Britney Spears Obsessive Compulsive is gonna give em something to think about...Have you got a rape alarm? Def take that...you only need to distract them for a second then they will lose interest..Stick to the trails is the most important thing, and talk to everyone you see when trekking: make sure others know where you are.
SUMMARIZED ANSWER […] In addition if the bear actually approaches you or charges you.. still stand your ground. Many times they will not actually come in contact with you, they will charge, almost touch you than run away. […] The actions you should take are different based on the type of bear. for example adult Grizzlies can t climb trees, but Black bears can even when adults. They can not climb in general as thier claws are longer and not semi-retractable like a Black bears claws. […] I truly disagree with the whole play dead approach because both Grizzlies and Black bears are oppurtunistic animals and will feed on carrion as well as kill and eat animals. Although Black bears are much more scavenger like and tend not to kill to eat as much as they just look around for scraps. Grizzlies on the other hand are very accomplished hunters and will take down large prey animals when they want. […] I have lived in the wilderness of Northern Canada for many years and I can honestly say that Black bears are not at all likely to attack you in most cases they run away as soon as they see or smell a human, the only places where Black bears are agressive is in parks with visitors that feed them, everywhere else the bears know that usually humans shoot them and so fear us. […]

Table 2: A summarized answer composed of five different portions of text generated with the S^{Π} scoring function; the chosen best answer is presented for comparison. The richness of the content and the good level of readability make it a successful instance of metadata-aware summarization of information in cQA systems. Less satisfying examples include summaries to questions that require a specific order of sentences or a compromise between strongly discordant opinions; in those cases, the summarized answer might lack logical consistency.

the total knowledge available about q, a coverage estimate of the final answers against it would have been ideal. Unfortunately the lack of metadata about those answers prevented us from proceeding in that direction. This consideration suggests the idea of building TK^q using similar answers in the dataset itself, for which metadata is indeed available. Furthermore, similar questions in the dataset could have been used to augment the set of answers used to generate the final summary with answers coming from similar questions. Wang et al. (2009a) presents a method to retrieve similar questions that could be worth taking in consideration for the task. We suggest that the retrieval method could be made Quality-aware. A Quality feature space for questions is presented by Agichtein et al. (2008) and could be used to rank the quality of questions in a way similar to how we ranked the quality of answers.

The Quality assessing component itself could be built as a module that can be adjusted to the kind of Social Media in use; the creation of customized Quality feature spaces would make it possible to handle different sources of UGC (forums, collaborative authoring websites such as Wikipedia, blogs etc.). A great obstacle is the lack of systematically available high quality training examples: a tentative solution could be to make use of clustering algorithms in the feature space; high and low quality clusters could then be labeled by comparison with examples of virtuous behavior (such as Wikipedia's Featured Articles). The quality of a document could then be estimated as a function of distance from the centroid of the cluster it belongs to. More careful estimates could take the position of other clusters and the concentration of nearby documents in consideration.

Finally, in addition to the chosen best answer, a DUC-styled query-focused multi-document summary could be used as a baseline against which the performances of the system can be checked.

5 Related Work

A work with a similar objective to our own is that of Liu et al. (2008), where standard multi-document summarization techniques are employed along with taxonomic information about questions. Our approach differs in two fundamental aspects: it took in consideration the peculiarities of the data in input by exploiting the nature of UGC and available metadata; additionally, along with relevance, we addressed challenges that are specific to Question Answering, such as Coverage and Novelty. For an investigation of Coverage in the context of Search Engines, refer to Swaminathan et al. (2009).

At the core of our work laid information trustfulness, summarization techniques and alternative concept representation. A general approach to the broad problem of evaluating information credibility on the Internet is presented by Akamine et al. (2009) with a system that makes use of semantic-aware Natural Language Preprocessing techniques. With analogous goals, but a focus on UGC, are the papers of Stvilia et al. (2005), Mcguinness et al. (2006), Hu et al. (2007) and

Zeng et al. (2006), which present a thorough investigation of Quality and trust in Wikipedia. In the cQA domain, Jeon et al. (2006) presents a framework to use Maximum Entropy for answer quality estimation through non-textual features; with the same purpose, more recent methods based on the expertise of answerers are proposed by Suryanto et al. (2009), while Wang et al. (2009b) introduce the idea of ranking answers taking their relation to questions in consideration. The paper that we regard as most authoritative on the matter is the work by Agichtein et al. (2008) which inspired us in the design of the Quality feature space presented in Section 2.1.

Our approach merged trustfulness estimation and summarization techniques: we adapted the automatic concept-level model presented by Gillick and Favre (2009) to our needs; related work in multi-document summarization has been carried out by Wang et al. (2008) and McDonald (2007). A relevant selection of approaches that instead make use of ML techniques for query-biased summarization is the following: Wang et al. (2007), Metzler and Kanungo (2008) and Li et al. (2009). An aspect worth investigating is the use of partially labeled or totally unlabeled data for summarization in the work of Wong et al. (2008) and Amini and Gallinari (2002).

Our final contribution was to explore the use of Basic Elements document representation instead of the widely used n-gram paradigm: in this regard, we suggest the paper by Zhou et al. (2006).

6 Conclusions

We presented a framework to generate trustful, complete, relevant and succinct answers to questions posted by users in cQA portals. We made use of intrinsically available metadata along with concept-level multi-document summarization techniques. Furthermore, we proposed an original use for the BE representation of concepts and tested two concept-scoring functions to combine Quality, Coverage, Relevance and Novelty measures. Evaluation results on human annotated data showed that our summarized answers constitute a solid complement to best answers voted by the cQA users.

We are in the process of building a system that performs on-line summarization of large sets of questions and answers from Yahoo! Answers. Larger-scale evaluation of results against other state-of-the-art summarization systems is ongoing.

Acknowledgments

This work was partly supported by the Chinese Natural Science Foundation under grant No. 60803075, and was carried out with the aid of a grant from the International Development Research Center, Ottawa, Canada. We would like to thank Prof. Xiaoyan Zhu, Mr. Yang Tang and Mr. Guillermo Rodriguez for the valuable discussions and comments and for their support. We would also like to thank Dr. Chin-yew Lin and Dr. Eugene Agichtein from Emory University for sharing their data.

References

Eugene Agichtein, Carlos Castillo, Debora Donato, Aristides Gionis, and Gilad Mishne. 2008. Finding high-quality content in social media. In Marc Najork, Andrei Z. Broder, and Soumen Chakrabarti, editors, *Proceedings of the International Conference on Web Search and Web Data Mining, WSDM 2008, Palo Alto, California, USA, February 11-12, 2008*, pages 183–194. ACM.

Susumu Akamine, Daisuke Kawahara, Yoshikiyo Kato, Tetsuji Nakagawa, Kentaro Inui, Sadao Kurohashi, and Yutaka Kidawara. 2009. Wisdom: a web information credibility analysis system. In *ACL-IJCNLP '09: Proceedings of the ACL-IJCNLP 2009 Software Demonstrations*, pages 1–4, Morristown, NJ, USA. Association for Computational Linguistics.

Massih-Reza Amini and Patrick Gallinari. 2002. The use of unlabeled data to improve supervised learning for text summarization. In *SIGIR '02: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 105–112, New York, NY, USA. ACM.

Dan Gillick and Benoit Favre. 2009. A scalable global model for summarization. In *ILP '09: Proceedings of the Workshop on Integer Linear Programming for Natural Langauge Processing*, pages 10–18, Morristown, NJ, USA. Association for Computational Linguistics.

Meiqun Hu, Ee-Peng Lim, Aixin Sun, Hady Wirawan Lauw, and Ba-Quy Vuong. 2007. Measuring article quality in wikipedia: models and evaluation. In *CIKM '07: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 243–252, New York, NY, USA. ACM.

Jiwoon Jeon, W. Bruce Croft, Joon Ho Lee, and Soyeon Park. 2006. A framework to predict the quality of

answers with non-textual features. In *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 228–235, New York, NY, USA. ACM.

Liangda Li, Ke Zhou, Gui-Rong Xue, Hongyuan Zha, and Yong Yu. 2009. Enhancing diversity, coverage and balance for summarization through structure learning. In *WWW '09: Proceedings of the 18th international conference on World wide web*, pages 71–80, New York, NY, USA. ACM.

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 (Coling 2008)*, pages 497–504, Manchester, UK, August. Coling 2008 Organizing Committee.

Ryan T. McDonald. 2007. A study of global inference algorithms in multi-document summarization. In Giambattista Amati, Claudio Carpineto, and Giovanni Romano, editors, *ECIR*, volume 4425 of *Lecture Notes in Computer Science*, pages 557–564. Springer.

Deborah L. Mcguinness, Honglei Zeng, Paulo Pinheiro Da Silva, Li Ding, Dhyanesh Narayanan, and Mayukh Bhaowal. 2006. Investigation into trust for collaborative information repositories: A wikipedia case study. In *In Proceedings of the Workshop on Models of Trust for the Web*, pages 3–131.

Donald Metzler and Tapas Kanungo. 2008. Machine learned sentence selection strategies for query-biased summarization. In *Proceedings of SIGIR Learning to Rank Workshop*.

Besiki Stvilia, Michael B. Twidale, Linda C. Smith, and Les Gasser. 2005. Assessing information quality of a community-based encyclopedia. In *Proceedings of the International Conference on Information Quality*.

Maggy Anastasia Suryanto, Ee Peng Lim, Aixin Sun, and Roger H. L. Chiang. 2009. Quality-aware collaborative question answering: methods and evaluation. In *WSDM '09: Proceedings of the Second ACM International Conference on Web Search and Data Mining*, pages 142–151, New York, NY, USA. ACM.

Ashwin Swaminathan, Cherian V. Mathew, and Darko Kirovski. 2009. Essential pages. In *WI-IAT '09: Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology*, pages 173–182, Washington, DC, USA. IEEE Computer Society.

Changhu Wang, Feng Jing, Lei Zhang, and Hong-Jiang Zhang. 2007. Learning query-biased web page summarization. In *CIKM '07: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 555–562, New York, NY, USA. ACM.

Dingding Wang, Tao Li, Shenghuo Zhu, and Chris Ding. 2008. Multi-document summarization via sentence-level semantic analysis and symmetric matrix factorization. In *SIGIR '08: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 307–314, New York, NY, USA. ACM.

Kai Wang, Zhaoyan Ming, and Tat-Seng Chua. 2009a. A syntactic tree matching approach to finding similar questions in community-based qa services. In *SIGIR '09: Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 187–194, New York, NY, USA. ACM.

Xin-Jing Wang, Xudong Tu, Dan Feng, and Lei Zhang. 2009b. Ranking community answers by modeling question-answer relationships via analogical reasoning. In *SIGIR '09: Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 179–186, New York, NY, USA. ACM.

Kam-Fai Wong, Mingli Wu, and Wenjie Li. 2008. Extractive summarization using supervised and semi-supervised learning. In *COLING '08: Proceedings of the 22nd International Conference on Computational Linguistics*, pages 985–992, Morristown, NJ, USA. Association for Computational Linguistics.

Honglei Zeng, Maher A. Alhossaini, Li Ding, Richard Fikes, and Deborah L. McGuinness. 2006. Computing trust from revision history. In *PST '06: Proceedings of the 2006 International Conference on Privacy, Security and Trust*, pages 1–1, New York, NY, USA. ACM.

Liang Zhou, Chin Y. Lin, and Eduard Hovy. 2006. Summarizing answers for complicated questions. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC), Genoa, Italy*.