The Computational Linguistics Summarization Pilot Task

Kokil Jaidka[1]*, Muthu Kumar Chandrasekaran[2], Rahul Jha[3], Christopher Jones[4]
Min-Yen Kan[2,5], Ankur Khanna[2], Diego Molla-Aliod[4], Dragomir R. Radev[3],
Francesco Ronzano[6] and Horacio Saggion[6]

Abstract

The Computational Linguistics (CL) Summarization Pilot Task was a pilot shared task to use citations to create summaries of scholarly research publications in the domain of computational linguistics. We describe the background for the task, corpus construction, evaluation methods for the pilot and survey the participating systems and their preliminary results. The experience gleaned from the pilot will assist in the proper organization of future shared task where difficulties with annotations and scale can be addressed.

1 Introduction

This paper describes the evolution and design of the CL pilot task[1] for the summarisation of computational linguistics research papers sampled from the Association of Computational Linguistics' (ACL) anthology. This task was concurrently publicized with TAC 2014, although it is not formally affiliated with the same, and shares its basic structure and guidelines with the more formal TAC 2014 Biomedical Summarization (BiomedSumm) track. A development corpus of training "topics" from CL research papers was released, each comprising a main, cited paper along with associated citing papers. Participants were invited to enter their systems in a task-based evaluation, similar to BiomedSumm.

This paper will describe the participating systems and survey their results from the task-based evaluation.

2 Background

Recent works (Mohammad et al., 2009);(Abu-Jbara and Radev, 2011) in scientific document summarisation have used citation sentences or citances from citing papers to create a multi document summary of the reference paper (RP).

As proposed by (VU, 2010); (Hoang and Kan, 2010) the summarisation can be decomposed into finding the relevant documents, in this case, the citing papers (CPs), then selecting sentences from those papers that cite and justify the citation and finally generate the summary. To help tackle each

[*] Authors appear in alphabetical order, with the exception of the coordinator of the task, whom is given the first authorship.

[1] This research is support in part by the Singapore National Research Foundation under its International Research Centre @ Singapore Funding Initiative and administered by the IDM Programme Office.

of these subproblems, we created gold standard datasets where human annotators identify the citances in each of about 10 randomly sampled citing papers for the RP.

A pilot study conducted in the information science domain indicated that most citations clearly refer to one or more specific aspects of the cited paper (Jaidka et al., 2013). For computational linguistics, we identified that the discourse facets being cited were usually the aim of the paper, methods followed and the results or implications of the work. Accordingly, we used a different set of discourse facets than BiomedSumm which suit CL papers better. Please note that this is a development corpus and only a training set is available for use now. Although, we plan to release a test set of documents for next year's evaluation, we report k fold cross-validated performance over the 10 documents for the systems registered for participation.

3 Corpus Construction

The CL community uses the ACL Anthology Reference Corpus (Bird et al., 2008) to evaluate and report performance of systems. To support further research in scientific document summarisation among the CL community and beyond we plan to build a manually annotated corpus using research papers sampled from the ACL Anthology. As first steps towards this goal we created an annotated development corpus by randomly sampling 10 documents from the ACL anthology.

We now describe our construction in detail. From the current, live ACL Anthology, there are approx 26K (exactly 25961) individual papers that are in the Anthology (as of 18 September 2014; including ones staged for publication but not actually published yet). These only include files that we have PDFs hosted (e.g., LREC is not represented as we don't hold these PDFs in the Anthology, just their metadata). This number is approximate as there are some files that are not publications (frontmatter, author indices) that are included. Culling all files before and including 2006, we get 13.8K (13838) publications, which include conference and journal articles. We randomized this list to remove any ordering affects. Starting from the top of the list, we used Google search [2] on (18 September 2013) to search for the publication - first by using its Anthology ID as a query (e.g.,"H89-2014.pdf") and not productive, re-queried by the title of the paper as a string (e.g., "Some Experiments with a Naive Bayes WSD System"). We look for a Scholar search result that shows # of citations. This was an approximation. We kept any paper with over 10 citations. Some papers had some similar versions that presented different citation rates; however, all of these were dropped anyways due to low citation rate. We vetted the citations from Google Scholar [3] for the citation spread being over 3 years as per citing papers' year of publication (as in Google Scholar). We did not attempt to check for publication years that Google Scholar doesn't report for some publications. We check only the earliest range manually to ensure that the citation is the correct one, as there are usually many examples of later citations. We also vetted that there were at least 10 of these citing sources available for download. Some candidates were dropped due to the few amount of available files that were freely downloadable from the Web. We ran a title search to find the paper in ACL Anthology Network [4] (AAN, February 2013 version). We inspected and listed the citing papers (incoming citations) Anthology ID, title and year where the citing papers were given in reverse chronological order. Note the citation count from Google / Google Scholar and AAN (Feb 2013 release) will differ substantially.

To report the final list of citing papers, we strived to provide at least 3 citing papers for each paper. To do so, we defined the following criteria in order or priority):

1. Non-list citation (i.e., at least one citation for the target paper not of the form [X,a,b,c]);
2. The oldest and newest citations within AAN; and,
3. Citations from different years.

We thus provided the oldest and newest citation regardless of criteria 1) and 3) and included a randomized sample of up to 8 additional citing paper

[2] http://www.google.com.sg

[3] http://www.scholar.google.com

[4] http://clair.eecs.umich.edu/aan/index.php

IDs that met either criteria 1) and 3). To do this, we started by first randomizing the list of citing papers and enumerating up to 8 additional citing papers. At this point, the citing papers were listed as either “old(est)”, “new(est)”, 1-8 (additional citing papers), or “sub” (substitute backup citing paper in case of disqualification of one of the 1-8 additional papers due to criterion 1). We combed through the lists of the additional 1-8 citing papers per target paper, from the top to the bottom of the randomized list. We unilaterally collected the top most oldest and top most newest paper, in case of ties. For the remaining (up to) 8 papers, we examined the citing paper’s PDF file to ensure that at least one citation made was of a single citation format (e.g., [X] and not [X,a,b,c]). Any invalidated files were marked with ”list” (citation) mark. The resulting final list was divided among the annotators to add human annotations using the same scheme used by annotators of the BiomedSumm track’s corpus.

Given a reference paper and up to 10 citing papers, annotators from National University of Singapore and Nanyang Technological University were instructed to find citations to the reference paper (RP) in the Citing Papers (CP). Annotators followed instructions used for annotation of corpus for the BiomedSumm to encourage cross participation across the two tasks. Specifically, the citation text, citation marker, reference text, and discourse facet were marked for each citation of the RP found in the CP.

4 The Task

In this task, we explore a new form of structured summary: a faceted summary of the traditional self-summary (the abstract) and the community summary (the collection of citances). As a third component, we propose to group the citances by the facets of the text that they refer to. We propose that by identifying first, the cited text span, and second, the facet of the paper (Aim, Method, Result or Implication), we can create a faceted summary of the paper by clustering all cited/citing sentences together by facet.

The CL-Summ Task is defined as follows:

Given: A topic consisting of a Reference Paper (RP) and upto ten Citing Papers (CPs) that all contain citations to the RP. In each CP, the text spans (i.e., citances) have been identified that pertain to a particular citation to the RP.

Task 1a: For each citance, identify the spans of text (cited text spans) in the RP that most accurately reflect the citance. These are of the granularity of a sentence fragment, a full sentence, or several consecutive sentences (no more than 5).

Task 1b: For each cited text span, identify what facet of the paper it belongs to, from a predefined set of facets.

Evaluation: Task 1 will be scored by overlap of text spans in the system output vs the gold standard created by human annotators.

5 Participating teams

The following 9 teams expressed an interest in participating while **BUG** 3 teams have submitted their findings which are included in this paper:

1. Taln.UPF, from Universitat Pompeu Fabra, Spain. They have proposed to adapt available summarisation tools to scientific texts.

2. clair_umich from University of Michigan, Ann Arbor, USA.

3. IITKGP_sum, from Indian Institute of Technology, Kharagpur, India. They plan to use citation network structure and citation context analysis to summarise the scientific articles.

4. CCS2014, from the IDA Center for Computing Sciences, USA. They will employ a language model based on the sections of the document to find referring text and related sentences in the cited document.

5. TabiBoun14, from the Boazii University, Turkey. They plan to modify an existing system for CL papers, wherein they use LIBSVM as a classification tool for face classification. They also plan to use the cosine similarity metric to compare text spans.

6. PolyAF, from The Hong Kong Polytechnic University.

7. TXSUMM, from University of Houston, Texas. Their system consists of applying similarity

kernels in an attempt to better discriminate between candidate text spans (with sentence granularity). They are using an extractive procedure with ranking algorithms.

8. MQ, from Macquarie University, Australia. They plan to use the same system that was used for the BiomedSumm track, with the exception that they will not incorporate domain knowledge (UMLS). For task 1a they will use similarity metrics to extract the top n sentences from the documents. For task 1b we will use a logistic regression classifier. For task 2 we will incorporate the distances from task 1 to rank the sentences.

9. A team from IHMC, USA

6 The MQ System - Finding the Best Fit to a Citance

Given the text of a citance, the MQ system ranks the sentences of the reference paper according to its similarity to the citance. Every sentence and its citance was modeled as a vector and compared using cosine similarity. The team experimented with different forms of representing the information in the vectors, and different forms of using the similarity scores to perform the final sentence ranking.

6.1 Baseline - Using *tf.idf*

For the baseline system, the *tf.idf* of all lowercased words was used, without removing stop words. Separate *tf.idf* statistics were computed for each reference paper, using the set of sentences in the paper and the citance text of all citing papers.

6.2 Adding texts of the same topic

Since the amount of text used to compute the *tf.idf* in Section 6.1 was relatively little, the complete text of all citing papers was added, under the presumption that citing papers are presumably of the same topic as the reference paper. By adding this text we hope to include complementary information that can be useful for extending and computing the *idf* component.

6.3 Adding context

In order to extend the information of each sentence in the reference paper and further add to the approach in Section 6.2, the text from the reference papers was added within a context window of 20 sentences by including the neighouring sentences, centered in the target sentence.

6.4 Re-ranking using MMR

The last experiment used Maximal Marginal Relevance (MMR) (**?**) to rank the sentences. All sentences were represented as *tf.idf* vectors of extended information as described in Section 6.3. Then, the final score of a sentence was the combination of the similarity with the citance and similarity of the other sentences of the summary according to the formula shown in Figure 1. A value of $\lambda = 0.97$ was chosen.

7 The clair_umich System - Comparing Overlap of Word Synsets

7.1 Data Preprocessing

The original SciSumm corpus contained data for 10 papers sampled from the ACL Anthology. For each of these papers, citing sentences were extracted from all its citing papers. Each citing sentence was then matched to a text segment in the original paper creating the final annotated dataset. The original source text for the papers in the SciSumm corpus was not sentence segmented, which made it difficult to compute evaluation metrics.

Data preprocessing of the SciSumm corpus was done in the following way - First, sentences from the reference papers were segmented and then matched to each of these source sentences to the SciSumm annotation files. This yielded a fixed set of source sentences from the original files, a subset of which were matched to each citing sentence. In this way, given a citing sentence, matching sentences from the source paper were compared to the gold standard sentences matched from the source paper and compute precision / recall.

The average number of source sentences matched for each citing sentence was 1.28 (with standard deviation 1.92). The maximum number of source sentences matched for a citing sentence was 7. Given that the total number of source sentences for papers ranged from between 100 to 600, this made it a very challenging classification problem.

$$\text{MMR} = \arg \max_{D_i \in R \setminus S} \left[\lambda(\text{sim}(D_i, Q)) - (1 - \lambda) \max_{D_j \in S} \text{sim}(D_i, D_j) \right]$$

Where:

- Q is the citance text.
- R is the set of sentences in the document.
- S is the set of sentences that haven been chosen in the summary so far.

Figure 1: Maximal Marginal Relevance (MMR)

7.2 Baseline System

Like the MQ system, the team first created a baseline system based on TF*IDF cosine similarity. For any citing sentence, the system computed the TF*IDF cosine similarity with all the sentences in the source paper, thus the IDF values differed across each of the 10 source papers.

7.3 Supervised System

The supervised system used knowledge based features derived from WordNet, syntactic dependency based features, and distributional features in addition to the simple lexical features like cosine similarity. These features are described below.

Lexical Features Two lexical features were used - tf*idf and the LCS (Longest Common Subsequence) between the citing sentence (C) and source sentence S, which is computed as:

$$\frac{|LCS|}{min(|C|, |S|)}$$

Knowledge Based Features The system also used set of features based on Wordnet similarity. Six wordnet based word similarity measures were combined to obtain six knowledge based sentence similarity features using the method proposed in (**?**). The wordnet based word similarity measures used are path similarity, WUP similarity (**?**), LCH similarity (**?**), Resnik similarity (**?**), Jiang-Conrath similarity (**?**), and Lin similarity (**?**).

Given each of these similarity measures, the similarities between two sentences was computed by first creating a set of senses for each of the words in each of the sentences. Given these two sets of senses, the similarity score between citing sentence C and source sentence S was calculated as follows:

$$sim_{wn}(C, S) = \frac{(\omega + \sum_{i=1}^{|\phi|} \phi_i) * (2|C||S|)}{|C| + |S|}$$

Here ω is the number of shared senses between C and S. The list ϕ contains the similarities of non-shared words in the shorter text, ϕ_i is the highest similarity score of the ith word among all the words of the lower text (**?**).

Syntactic Features An additional feature based on similarity of dependency structures was used, by applying the method described in (**?**). The Stanford parser was used to obtain dependency parse all the citing sentences and source sentences. Given a candidate sentence pair, two syntactic dependencies were considered equal if they have the same dependency type, govering lemma, and dependent lemma. If R_c and R_s are the set of all dependency relations in C and S, the dependency overlap score was computed using the formula:

$$sim_{dep}(C, S) = \frac{2 * |R_c \cap R_s| * |R_c||R_s|}{|R_c| + |R_s|}$$

8 The UPF System

8.1 Pre-processing / documents preparation:

The UPF system carried out the following set of preprocessing steps on the papers of each topic:

- Sentence segmentation:To identify candidate sentences that will be validated or rejected in the following pre-processing steps;

- Tokenizer and POS tagger:Using the open-source GATE software

- Sentence sanitizer: To remove incorrectly annotated sentences, relying on a set of rules and heuristics;

- Document structural analyzer: To classify each sentence as belonging to one of the following document structural categories: Abstract, Introduction, Result_Discussion, Experimental_Procedure, Supplemental_Data, Material_Method, Conclusion, Acknowledgement_Funding, and Reference;

- Sentence TFIDF vector computation: To associate to each sentence a TFIDF vector where the IDF values are computed over all the papers of the related topic (up to 10 citing paper and one reference paper).

8.2 Algorithm for identifying reference paper text spans for each citance

- For each citance its global citatance context span was considered as the union of the citance context spans marked by human annotators (in this case, there was only one available human annotation, so no union was required).

- Then we select the sentences of the citing paper that overlap totally or partially the global citatance context span; these sentences are referred to as the citance context sentences (CtxSent1,..., CtxSentN),

- We characterize the citance by the document structural category associated with most of its citance context sentences (CtxSent1,..., CtxSentN). In case of tie in the number of occurrences of document structural categories among all the citance context sentences, we choose the document structural category that is most frequent in the citing paper. In case of persisting tie, we select the document structural category that is most frequent in the whole set of citing and reference papers.

- We associate to each reference paper sentence (RefSent) a score equal to the sum of its TFIDF vector cosine similarity with each citance context sentence (CtxSent1,..., CtxSentN).

- Then, we weight the score of RefSent by the relative relevance in the whole training corpus of this kind of link between document structural categories. In order to do so, we consider the document structural category previously associated to the citation together with the document structural category of the RefSent. For instance, if there is a citance associated to the INTRO that references a RefSent belonging to the Abstract and we know that in the whole training corpus this situation occurs in 6.5% of citance-referenced sentence pairs, we multiply the RefSent score for 0.065, obtaining the final RefSent score.

- We choose the 3 reference paper sentences (RefSents) with the highest final RefSent score as the reference paper text spans.

8.3 Algorithm for identifying the discourse facet of the cited text spans

A linear-kernel SVM classifier was trained to associate each citance with one of the five text facets considered in Task 1b. Each citance was characterized by lexical and semantic features extracted from the sentences belonging to the citance context together with the sentences of the reference paper selected as outcome of Task 1a. Some of the features exploited were:

- relative number of sentences belonging to each document structural category

- relative number of sentences belonging to the citance context or reference paper

- relative number of POS

- presence of key lexical patterns

9 Evaluation and Results

Two of the teams have submitted their results so far, and the evaluation is based on the ROUGE metric (**?**). ROUGE is a popular evaluation method for summarisation systems that compares the text output of the system against a set of target summaries. Since ROUGE uses the actual contents words, and not the offset information, we expect that this metric will give non-zero results for cases when a system chooses a sentence that is similar to, but not exactly, the one chosen by the annotator.

The MQ system was an unsupervised system while clair_umich system was supervised. clair_umich reports cross validated performance over the 10 topics while MQ evaluated their system over all 10 topics in a single run. For the MQ system, the output is the set of selected sentences, and the target summaries are the sentences given by the annotators. For the clair_umich system, the ROUGE-L scores were computed for each citing sentence in each annotation file separately and then averaged for a topic.

The following paragraphs describe the results for Task 1a, 1b, and the bonus Task 2 which was attempted by the MQ system.

9.1 Task 1a: For each citance, identify the spans of text (cited text spans) in the RP

Table 2 shows the ROUGE-L F1 scores of each individual reference document from the SciSumm dataset.

9.2 Task 2: Generate a structured summary of the RP and all of the community discussion of the paper represented in the citances

The MQ team performed an additional test to see whether information from the citances were useful for building an extractive summary, as is the case with the BiomedSumm data (**?**). They implemented extractive summarisation systems with and without information from the citances. The summarisers without information from the citances scored each sentence as the sum of the *tf.idf* values of the sentence elements. They tried the *tf.idf* approach described in Section refsec:tfidf.

The summarisers with information from the citances scored each candidate sentence i on the basis of rank(i,c) obtained in task 1a, which has values between 0 (first sentence) and n (last sentence) and represents the rank of sentence i in citance c:

$$\text{score}(i) = \sum_{c \in \text{citances}} 1 - \frac{\text{rank}(i, c)}{n}$$

The summaries were evaluated using ROUGE-L, where the model summaries are the abstracts of the corresponding papers. Since paper X96-1048 of the SciSumm data did not have an abstract, it was omitted from this experiment. Table 3 shows the breakout of ROUGE-L F1 scores per document.

10 Discussion

10.1 Comparison with the BioMedSumm task

Table 4 compares the results of the MQ system's experiments with the SciSumm data, against the results from the BiomedSumm data. In all results the systems were designed to return 3 sentences, as specified in the shared task. All short sentences (under 50 characters) were ignored, to avoid including headings or mistakes made by the sentence segmentation algorithm.

The results show an improvement in both domains, with the exception that MMR does not improve over the run that uses *tf.idf* over context in SciSumm, whereas there is an improvement in BiomedSumm. The absolute values are better in the BiomedSumm data, and looking at the confidence intervals it can be presumed that the difference between the best and the worst run is statistically significant in the BiomedSumm data. The results in the SciSumm data are poorer in general and there are no statistically significant differences. However, this may be an artifact of the small size of the corpus. Overall, the improvement of results in SciSumm mirrors that of the BiomedSumm data, so it can be suggested that on adding more information to the models that compute *tf.idf*, the results improve. It is expected that alternative approaches, which gather related information to be added for computing the vector models will produce even better results. The results with MMR appears to be contradictory across the two domains but the difference is so small that it might not be statistically significant even when we add more evaluation data.

10.2 Tweaking the Parameters - the clair_umich Baseline

For any citing sentence, the TF*IDF cosine similarity was computed with all the sentences in the source paper, and any sentences that had a cosine similarity higher than a given threshold were added to the matched sentences. Table 5 shows the precision/recall for different values of the cosine threshold:

MQ System			clair_umich System		
R	P	F1	R	P	F1
0.335	0.212	0.223	0.0	0.0	0.738

Table 1: Task 1a performance for the participating systems expressed as ROUGE-L score averaged over all topics

Paper ID	MQ System	clair_umich System
C90-2039	0.235	0.801
C94-2154	0.288	0.842
E03-1020	0.239	0.846
H05-1115	0.350	0.736
H89-2014	0.332	0.770
J00-3003	0.196	0.644
J98-2005	0.101	0.658
N01-1011	0.221	0.689
P98-1081	0.200	0.776
X96-1048	0.248	0.622

Table 2: Task 1a ROUGE-L F1 scores for individual topics

Paper ID	*tf.idf*	task1a *tf.idf*	task1a MMR
C90-2039_TRAIN	0.347	0.315	0.293
C94-2154_TRAIN	0.095	0.123	0.120
E03-1020_TRAIN	0.189	0.189	0.196
H05-1115_TRAIN	0.134	0.306	0.321
H89-2014_TRAIN	0.294	0.319	0.320
J00-3003_TRAIN	0.221	0.382	0.367
J98-2005_TRAIN	0.221	0.216	0.233
N01-1011_TRAIN	0.187	0.268	0.284
P98-1081_TRAIN	0.241	0.210	0.206
Average	0.214	0.259	0.260

Table 3: ROUGE-L F1 results for summaries generated by the MQ system

	SciSumm				BiomedSumm			
Run	R	P	F1	CI	R	P	F1	CI
tf.idf	0.316	0.198	0.211	0.185–0.240	0.273	0.326	0.279	0.265–0.293
topics	0.324	0.201	0.217	0.191–0.245	0.288	0.357	0.300	0.285–0.316
context	0.339	0.214	0.225	0.197–0.255	0.291	0.372	0.308	0.293–0.323
MMR	0.335	0.212	0.223	0.195–0.251	0.290	0.375	0.308	0.293–0.323

Table 4: ROUGE-L results of the MQ system runs for task 1a

Similarity Threshold	Precision	Recall	F1-score
0.01	0.027	0.641	0.051
0.05	0.048	0.426	0.087
0.1	0.060	0.235	0.095
0.2	0.079	0.081	0.080
0.3	0.062	0.032	0.042
0.4	0.022	0.085	0.012
0.5	0.007	0.002	0.003

Table 5: Precision/Recall for different values of the cosine threshold for the baseline clair_umich system

The F1-score seems to reach a maxima at a similarity threshold of about 0.1. The recall at the threshold of 0.1 is about 0.23, while the precision is only 0.06. This suggests that initial progress can be made on this problem by first removing these spurious matches that have high lexical similarity. We present some error analysis in the next section.

Error Analysis for the Baseline System A number of errors made by the baseline system are due to source sentences that match the words but differ slightly in their information content. Here is an example.

Citing text: "use the BNC to build a co-occurrence graph for nouns, based on a co-occurrence frequency threshold"

True positives:

- "Following the method in (Widdows and Dorow, 2002), we build a graph in which each node represents a noun and two nodes have an edge between them if they co-occur in lists more than a given number of times."

False positives:

- "Based on the intuition that nouns which co-occur in a list are often semantically related, we extract contexts of the form Noun, Noun,... and/or Noun, e.g. "genomic DNA from rat, mouse and dog"."

- "To detect the different areas of meaning in our local graphs, we use a cluster algorithm for graphs (Markov clustering, MCL) developed by van Dongen (2000)."

- "The algorithm is based on a graph model representing words and relationships between them."

Even though the false positive sentences contain the same lexical items (nouns, co-occurrence, graph), they differ slightly in the facts presented. Detection of such subtle differences in meaning might be challenging for an automated system.

Another set of difficult sentences is when the citing sentence says something that is implied by the sentence in the source paper. For example:

Citing text: "The line of our argument below follows a proof provided in ... for the maximum likelihood estimator based on nite tree distributions"

False negatives:

- "We will show that in both cases the estimated probability is tight."

Here, the citing text mentions a proof from source paper, but to match the sentence in the source paper, the system needs to understand that the act of showing something in a scientific paper constitutes a proof.

11 Limitations of this development corpus

There were several limitations in the dataset which were identified in the process of annotating and parsing the corpus for use by the participating systems; these are discussed below.

- The use of "..." where text spans are snippets:The use of "..." follows the BioMedSumm standard practice of indicating discontiguous texts. In Citation Text and Reference Text fields, the "..." means that there is a gap between two text spans (citation spans or reference spans). They may be on different pages, so the gap might be a text. There might be a formula or a figure there, or some text encoding which is not a part of the annotation. However, this notation caused mismatches for sentences which used text from different parts of the same sentence.

- Small size of the training corpus:The corpus comprised only a training set of 10 topics, each with upto 10 citing documents. In this small dataset, participants were asked to conduct a 10-fold cross validation. The small size of the data set meant that there were no statistically significant results, but significance could only be guessed at, from the overall trend of the data.

- Errors in parsing the file: Some of the older PDF files, when parsed to text or xml, had such as misspelt words, spaces within words, sentences in the wrong place and so on. Unfortunately these errors were OCR parsing errors, and not in our control. It was recommended that the participants should configure their string matching to be lenient enough to tackle such problems.

- Errors in citation/reference offset numbers: In the original annotations, citation/reference offset numbers were character-based, and relative to an xml encoding which was not shared in the task, and did not match with the offset numbers on the text-only, cleaned version of the document. Although the text versions of the source documents were shared with the intention to help the participants, this often made their tasks more difficult if their system was geared towards numerical and not system matching. A solution was found for reference offsets by revising them to sentence id numbers based on available XML files from the clair_umich system's pre-processing stage; however, the citation offsets remain character-based.

- Text encoding: Often, the text was not in UTF-8 format as expected. Some participating teams, like the UPF, solved this by running the universal charset tool provided by Google Code over all the text and annotations in order to determine the right file encoding to use. It was found that some of the files were also in WINDOWS-1252 and GB18030 formats.

- Errors in file construction: An automatic, opensource software was used to map the citation annotations from a software, Protege, to a text file. However, participants identified several errors in the output - especially in cases where there was one-to-many mapping between citations and references. Besides this, several annotation texts had no annotation id (Citance Number field).

12 Conclusions

The results of the experiments reported in this paper suggest that information from related papers may be useful to find the sentences of the reference paper that best match the citances.

Our experiments also suggest that information from the citances may be useful for building an extractive summary. This conclusion is compatible with prior research that suggest that, in general, information from citing papers may be useful for building summaries, as was stated in the original goals of the BiomedSumm and CL-Summ shared tasks.

References

Amjad Abu-Jbara and Dragomir Radev. 2011. Coherent citation-based summarization of scientific papers. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 500–509. Association for Computational Linguistics.

Steven Bird, Robert Dale, Bonnie J Dorr, Bryan R Gibson, Mark Joseph, Min-Yen Kan, Dongwon Lee, Brett Powley, Dragomir R Radev, and Yee Fan Tan. 2008. The acl anthology reference corpus: A reference dataset for bibliographic research in computational linguistics. In *LREC*.

Cong Duy Vu Hoang and Min-Yen Kan. 2010. Towards automated related work summarization. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 427–435. Association for Computational Linguistics.

Kokil Jaidka, Christopher SG Khoo, and Jin-Cheon Na. 2013. Literature review writing: how information is selected and transformed. In *Aslib Proceedings*, volume 65, pages 303–325. Emerald Group Publishing Limited.

Saif Mohammad, Bonnie Dorr, Melissa Egan, Ahmed Hassan, Pradeep Muthukrishan, Vahed Qazvinian, Dragomir Radev, and David Zajic. 2009. Using citations to generate surveys of scientific paradigms. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 584–592. Association for Computational Linguistics.

HOANG CONG DUY VU. 2010. Towards automated related work summarization.