AUTOMATIC SUMMARIZATION FOR STUDENT REFLECTIVE RESPONSES

by

Wencan Luo

B.E. in Computer Science and Technology, University of Science & Technology Beijing, 2008

M.E. in Computer Science, Graduate University of the Chinese Academy of Sciences, 2011

Submitted to the Graduate Faculty of
the Kenneth P. Dietrich School of Arts and Sciences, Department of
Computer Science in partial fulfillment
of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2017

UNIVERSITY OF PITTSBURGH

KENNETH P. DIETRICH SCHOOL OF ARTS AND SCIENCES, DEPARTMENT OF COMPUTER SCIENCE

This dissertation was presented

by

Wencan Luo

It was defended on

April 21 2017

and approved by

Diane Litman, Computer Science Department

Rebecca Hwa, Computer Science Department

Jingtao Wang, Computer Science Department

Fei Liu, Computer Science Department, University of Central Florida

Dissertation Director: Diane Litman, Computer Science Department

AUTOMATIC SUMMARIZATION FOR STUDENT REFLECTIVE RESPONSES

Wencan Luo, PhD

University of Pittsburgh, 2017

Educational research has demonstrated that asking students to respond to reflection prompts can improve both teaching and learning. However, summarizing student responses to these prompts is an onerous task for humans and poses challenges for existing summarization methods.

From the input perspective, there are three challenges. First, there is a lexical variety problem due to the fact that different students tend to use different expressions. Second, there is a length variety problem that student inputs range from single words to multiple sentences. Third, there is a redundancy issue since some content among student responses are not useful. From the output perspective, there are two additional challenges. First, the human summaries consist of a list of important phrases instead of sentences. Second, from an instructor's perspective, the number of students who have a particular problem or are interested in a particular topic is valuable.

The goal of this research is to enhance student response summarization at multiple levels of granularity.

At the sentence level, we propose a novel summarization algorithm by extending traditional ILP-based framework with a low-rank matrix approximation to address the challenge of lexical variety.

At the phrase level, we propose a phrase summarization framework by a combination of phrase extraction, phrase clustering, and phrase ranking. Experimental results show the effectiveness on multiple student response data sets.

Also at the phrase level, we propose a quantitative phrase summarization algorithm in order to estimate the number of students who semantically mention the phrases in a summary. We first introduce a new phrase-based highlighting scheme for automatic summarization. It highlights the phrases in the human summaries and also the corresponding semantically-equivalent phrases in student responses. Enabled by the highlighting scheme, we improve the previous phrase-based summarization framework by developing a supervised candidate phrase extraction, learning to estimate the phrase similarities, and experimenting with different clustering algorithms to group phrases into clusters. Experimental results show that our proposed methods not only yield better summarization performance evaluated using ROUGE, but also produce summaries that capture the pressing student needs.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

PREFACE

With the dream to write a program to pass the Turing Test, I came to University of Pittsburgh to pursue a PhD degree. The dream has not come true yet but getting a PhD degree in Natural Language Processing makes me a little bit closer to the dream. It is a great accomplishment for me and I am glad that I am able to push the boundary of world knowledge a little bit. However, it won't happen without many helps from many people.

I first want to thank my wife, Xiaoying Jia. Thank you to company with me during day and night. Thank you to bring our two lovely kids to our family. Thank you for your sacrifice for our family. I know it was hard for you to quit a decent job and came to U.S. with me. I will continue to work hard to make our life better.

I then want to specially thank my advisor, Diane Litman. I still remember the mistakes I made during my first two years. The lesson I learn about time management will be beneficial to me for my entire life. I am glad that I was not struck out. Thank you for the patient, encouragement, and freedom to allow me to do what I want to do. I am proud to be one of your students.

I also want to thank my parents, my friends, my colleagues, my committee, etc. Just to name a few, David Albert, Lingjia Deng, Xiangmin Fan, Mitch Freed, Fei Liu, Zitao Liu, Huy Nguyen, Jiannan Ouyang, Phuong Pham, Zahra Rahimi, Jingtao Wang, Xiang Xiao, Wenting Xiong, Fan Yang, Fan Zhang, Xianwei Zhang, Lin Zhao...

1.0 INTRODUCTION

1.1 MOTIVATION

Automatic text summarization seeks to generate concise, textual summaries from a large collection of text documents. It reduces users' *information overload* and is a desired capability in many scenarios. Since its debut in 1958 (Luhn, 1958), automatic summarization techniques have been broadly applied to a number of areas, for example, judging if a document is relevant to a topic of interest (Mani et al., 2002), clustering new articles on the same event (McKeown et al., 2002; Radev et al., 2005; Vuurens et al., 2015), producing snippets for search engines (Jones et al., 2004; Varadarajan and Hristidis, 2006; Turpin et al., 2007), enabling fast browsing of world wide web pages (Buyukkokten et al., 2001; Zhang et al., 2004), generating online advertising keywords (Thomaidou et al., 2013), generating an overview paper of a research area (Nanba and Okumura, 1999; Teufel and Moens, 2002; Qazvinian and Radev, 2008; Mohammad et al., 2009), extracting popular events in real time among social media data such as tweets (Shen et al., 2013; Schinas et al., 2015), etc.

There are several distinctive ways to perform summarization. *Extractive summarization* is the most popular one and it produces summaries by concatenating sentences taken exactly as they appear in the materials being summarized. Abstractive summarization produces summaries that are expressed in the words of the summary author. Compressive summarization produces summaries from compressed sentences, not necessarily extracts. It is a trade-off between extractive summarization and abstractive summarization. Since state-of-the-art abstractive and compressive summarization approaches (Berg-Kirkpatrick et al., 2011; Li et al., 2013a; Liu et al., 2015; Li et al., 2016b) often have difficulties handling ill-formed sentences and spelling errors, we thus focus on extractive summarization, where an extraction unit tends to carry coherent semantic information and the results are easily interpretable to users.

Summarization can also be categorized into different granularities (Kan, 2015): word, phrase, and sentences. The most popular summarization granularity is sentences (**S**), in which a summary consists of sentences. For example, news documents can be summarized by a single headline. At the same time, a specific application or user need might call for a keyword (**W**) summary, which consists of a set of indicative words mentioned in the input. It is easy to read and browse (Ueda et al., 2000). In addition, producing a summary as a list of keyphrases has received considerable attention (Wu et al., 2005; Liu et al., 2009; Medelyan et al., 2009; Hasan and Ng, 2014), aiming to select important phrases (**P**) from input documents.

	Statistics				
	Tasks	Docs/task	Tokens/sen	Length	Granularity
Student response (Eng)*	36	49	9.1	4 P	P
Student response (Stat2015)*	44	39	6.0	5 S/P	S/P
Student response (Stat2016)*	48	42	4.3	5 S/P	S/P
Student response (CS2016)	46	22	8.8	5 S/P	S/P
Reviews (camera)	3	18	22.7	10 S	S
Reviews (movie)	3	18	24.4	10 S	S
Reviews (peer)	3	18	19.2	10 S	S
News articles (DUC04)*	50	10	22.4	105 W	S

Table 1.1: Selected summarization data sets. Publicly available data sets are marked with an asterisk (*). The statistics involve the number of summarization tasks (Tasks), average number of documents per task (Docs/task), average number of tokens per sentence (Tokens/sen), output summary length (Length), and the granularity of summarization units (Granularity). **W**, **P** and **S** are short for word(s), phrase(s), and sentence(s) respectively.

A summary of summarization data sets and their statistics that we are going to use in this work is presented in Table 1.1. The student response (Eng) was collected by Menekse et al. (2011) using paper-based surveys; student responses (Stat2015, Stat2016, and CS2016) are collected using a mobile application developed by us (Luo et al., 2015; Fan et al., 2015, 2017); data sets of news articles and product reviews are contributed by other researchers. In this work, we mainly focus on automatic summarization techniques for student responses. To our best knowledge, this type of source is new and has not been explored by existing research. Since one of our proposed techniques is not unique to this particular type of data, we will also consider applying it to data sources of news and reviews.

Like other data sources (news, websites, social media), users in the educational domain also face the challenge of *information overload*. Currently, student course feedback is generated daily in both classrooms and online course discussion forums (e.g., *Piazza.com*). Therefore, it is expensive and time consuming for humans to summarize student feedback. This is becoming more severe in large courses (e.g., introductory STEM, MOOCs). Most existing research efforts on reflection prompts focus on post-hoc analysis, learners' self-reflections, and learner-to-instructor feedback. Little effort has been made to facilitate instructor-to-student, student-to-student interactions in a timely manner in large classrooms. It is therefore desirable to automatically summarize the student feedback produced in online and offline environments. In this work, we are considering one particular type of student responses, named "*reflective feedback*" (Boud et al., 2013), which has been shown to enhance interaction between instructors and students by educational research (Van den Boom et al., 2004; Menekse et al., 2011). Specifically, students are presented with such prompts after each lecture and asked to provide responses.

Summarizing student responses is challenging from both the input and the output perspectives, as illustrated in Table 1.2.

From the input perspective, there are at least three challenges. 1) There is a **lexical variety** problem due to the fact that different students tend to use different expressions. For example, in Table 1.2, "bike elements" (S11), "the bicycle" (S13), "part of a bike" (S18, S40), and "bicycle parts" (S36) are different expressions that communicate the same or similar meanings. Similarly, "the main topics of this course" (S12), "what we will learn in this class" (S26), and "what we are going to learn this semester" (S34) are similar to each other but with a different vocabulary. 2) There is a **length variety** problem. A student

Prompt
Describe what you found most interesting in today's class
Student Responses
S1: Professors "student centered learning" approach
S2: Class seems interesting, look forward to the semester
S3: How lacking my ability was to describe in my own words the bonding concepts although I did have a vague of understanding of the differences
S4: The most interesting thing in today's class was learning about the grading scale because I have never heard of a normalized grading scale, and I like the fairness of it
S5: Process of manufacturing
S6: The bonding pre-assignment
S7: Extrusion
S8: I found the group activity most interesting
S9: I thought the hip thing was cool
S10: Process that make materials
S11: I found the properties of bike elements to be most interesting
S12: The main topics of this course seem interesting and correspond with my major (Chemical engineering)
S13: The table discussion at analyzing the bicycle
S14: Processing Vulcanization and floating on molten tin
S15: Separating a single object (light bulb) into the 3 families of materials
S16: How materials are manufactured
S17: This class is interaction based
S18: The process of making different part of a bike
S19: The glass is formed on molten tin
S20: The pre-test
S21: Separating a single object (light bulb) into the 3 families of materials
S22: Having a group to share experience with
S23: The introduction of the different uses of material sciences in life ? Hip replacement
S24: Tungsten is the best element for bulb filaments
S25: The normalization of grades what the grade percentage is made up of
S26: Finding out what we will learn in this class was interesting to me
S27: I like the interaction and the activity
S28: I wasn't aware of what the class was about clearly the introduction during this first class fixed this information which was previously unknown
S29: Class activity with matching was great for learning
S30: I thought it was interesting that only 3 families of materials were mentioned. Do all materials fit into those categories? Or are there others that not studied in this class?
S31: I already had this lecture in MSE 100 but I would say that the bonding test was the most interesting
S32: the application of chemistry. I have never really used it
S33: Differences between characteristics of materials
S34: Most interesting would be what we are going to learn this semester
S35: hip replacement in notes
S36: The activity with the bicycle parts
S37: Different type of materials & the uses
S38: I found that the grade normalizing and how that works the most interesting part
S39: the "educating tomorrow's engineer" page! :)
S40: "part of a bike" activity
S41: The different properties and applications of various materials. Families of materials
S42: I was interested in learning what I will be learning about this semester those categories? Or are there others that not studied in this class?
Human Summary
- Group activity of analyzing bicycle's parts [12]
- Materials processing [6]
- The main topic of this course [4]

Table 1.2: Example student responses and a reference summary created by the teaching assistant. The numbers in the square brackets in the human summary indicate the number of students who semantically mention each phrase. 'S1'–'S42' are student IDs.

response is shorter than other types of sources in terms of the number of tokens, as shown in Table 1.1. Making it even worse, the linguistic units of student inputs range from single words (S7) to multiple sentences (S30, S42). In other types of sources (e.g., product reviews), short sentences (e.g., less than 5 words) are often discarded (Xiong and Litman, 2014). However, for student responses, short ones could also be useful (S5). Student responses have a limited internal structure within a paragraph, therefore, it is not necessarily true that the first sentence or the last sentence is generally more important than others, making position features working for news and scientific articles less useful for student responses (Luo and Litman, 2015). 3) For our particular problem, there is a **redundancy** issue since some content among student responses are not useful. For example, extracting sentences that include phrases such as “to be most interesting” (S11), “was interesting to me” (S26), and “I was interested in” (S42) is a waste of space, given that the prompt is asking “Describe what you found most interesting in today’s class.”

From the output perspective, there are at least two additional challenges. 1) The human summaries consist of a list of important phrases (**phrase scale**). Note, the summary phrases are not necessarily extracted from student responses, which makes our task different from the task of keyphrase extraction. 2) From an instructor’s perspective, the quantitative number of students (**quantity**) who have a particular problem or are interested in a particular topic is extremely valuable, as shown in the human summary’s square brackets in Table 1.2. It assumes the concepts (represented as phrases) mentioned by more students should rank higher in the summary. For example, from the summary, an instructor can know that 12 out of 42 students are interested in “Group activity of analyzing bicycle’s parts.” This is difficult to automate due to the lexical variety, and a better understanding of the student responses is needed. As far as we know, although there is work on quantitative summarization based on keywords or simple bigrams (Yatani et al., 2011; Van Labeke et al., 2013), no existing summarization technique delivers quantitative results together with the summary at the phrase or sentence scale.

To address the challenges above (lexical variety, length variety, redundancy, phrase scale, and quantity), we propose several new approaches to summarize student responses.

At the sentence level, we propose a new approach to summarizing student feedback (Luo et al., 2016b), which extends the standard Integer Linear Programming (ILP) framework by approximating the co-occurrence matrix using a low-rank alternative, to address the challenge of lexical variety. The resulting system allows sentences authored by different students to share co-occurrence statistics. For example, “The activity with the bicycle parts” (S36) will be allowed to partially contain “bike elements” (S11) although the latter did not appear in the sentence. Experiments show that our approach produces better results on the student responses Eng and CS2016 (Table 1.1) in terms of both automatic evaluation and human evaluation. We expect this method is applicable to other data sets since people generally tend to use diverse lexical terms to express the same or similar semantic meanings. Particularly, user-generated content, such as online product reviews are expected to have a high lexical diversity issue like student responses. We therefore perform extensive experiments on these data sets to provide insights on why and when the model works.

At the phrase level, we propose a novel summarization algorithm in order to meet the need of aggregating and displaying reflections in a mobile application, given that the output of human summaries are phrases. It differs from traditional methods in two primary ways (Luo and Litman, 2015). 1) It is an extractive summarization technique at the scale of phrases, in which summaries are created from extracted phrases rather than from sentences. Phrases are easy to read and browse like keywords, and fit better on small devices when compared to sentences. After phrase extraction, long sentences are decomposed into different short phrases, which will be processed together with phrases from short sentences. In addition, only noun phrases are extracted with a syntax parser and thus phrases such as “to be most interesting” (S11) and “was interesting” (S26) are filtered out. In this way, it addresses the length variety and redundancy challenge. 2) We adopt a metric clustering paradigm based on k-medoids with a semantic distance to group extracted phrases; a semantic metric allows similar phrases to be grouped together even if they are in different textual forms, in order to address the lexical variety and quantity challenges.

Also at the phrase level, we propose a quantitative phrase summarization algorithm (Luo et al., 2016a) in order to estimate the number of students who semantically mention the phrases in a summary, addressing the quantity challenge, which is important for instructors. We observe that the proposed phrase summarization (Luo and Litman, 2015) partially addresses this challenge, but it has three limitations. First, noun phrases do not suffice. Other types of phrases such as "how confidence intervals linked with previous topics" are useful and should be allowed. Second, clustering is based on similarity, but the similarity of phrases that do not appear in a background corpus (i.e., the corpus used to learn the similarities) cannot be captured in the previous setting. Lastly, a greedy clustering algorithm k-medoids (Kaufman and Rousseeuw, 1987) was previously used to group candidate phrases. It ignores global information and may suffer from a "collapsing" effect, which leads to the generation of a large cluster with unrelated items (Basu et al., 2013). To address these limitations, we first introduce a new phrase-based highlighting scheme for automatic summarization. In the new scheme, human annotators are instructed to 1) create summary phrases from the student responses, 2) associate a number with each summary phrase which indicates the number of students who raise the issue (henceforth **student supporters**), and 3) highlight the corresponding phrases in both the human summaries and student responses. Enabled by the highlighting scheme, we improve the phrase-based summarization framework proposed by Luo and Litman (2015) by developing a supervised candidate phrase extraction via sequence labeling, learning to estimate the phrase similarities, and experimenting with different clustering algorithms to group phrases into clusters. We further introduce a new metric that offers a promising direction for making progress on developing automatic summarization evaluation metrics. Experimental results show that our proposed methods not only yield better summarization performance evaluated using ROUGE, but also produce summaries that capture the pressing student needs.

1.2 RESEARCH SUMMARY

The goal of this research is to enhance student response summarization at multiple levels of granularity.

At the sentence level, we propose a novel summarization algorithm by extending the ILP-based framework with a low-rank matrix approximation, in which we hypothesize that:

- **H1.1**: *The low-rank matrix approximation is able to capture similar concepts on student* *responses.*

- **H1.2**: *The extended-ILP framework delivers better summarization performance than the traditional ILP-based framework on student responses.*
- **H1.3**: *The extended-ILP framework is applicable to other data sets including news and reviews, and it will yield better summarization performance.*

At the phrase level, we propose a phrase summarization algorithm by a combination of phrase extraction, phrase clustering, and phrase ranking. We hypothesize that:

- **H2**: *The proposed phrase summarization improves summarization performance to student responses.*

Also at the phrase level, we try to improve the phrase summarization enabled by the highlighting scheme, in which we hypothesize that:

- **H3.1**: *Phrase extraction with a supervised sequence labeling model can generate better candidate phrases than using noun phrases only. It thus improves the end-to-end summarization performance.*
- **H3.2**: *Supervised similarity learning can better measure the similarity between phrases and thus improve the performance of summarization.*
- **H3.3**: *The proposed quantitative summarization gives a better estimate of student numbers than the previous clustering-based phrase summarization.*

1.3 CONTRIBUTIONS

This research contributes to both NLP and education researchers.

- For the NLP community, we first propose a new way to address the lexical variety challenge by introducing a low-rank approximation to the co-occurrence matrix. It helps tackle the high lexical diversity issue and we explore different factors that impact the performance of the proposed model. We perform extensive experiments on a number of datasets, ranging from student course feedback, product reviews, to news reports, to provide insights on why and when the model works. Second, we propose a general phrase summarization framework by adapting existing sentence-level summarization techniques. Lastly, we propose a quantitative summarization approach to enhance summaries by associating the number of people who semantically mention the phrases in a summary and propose a new evaluation metric based on color matching measuring how well phrase summaries capture the most pressing student needs.

- For education researchers, we offer a new application using NLP techniques to summarize student responses in order to facilitate the interaction between instructors and students.

1.4 THESIS OUTLINE

This chapter introduces the background of automatic summarization and illustrates the challenges of summarizing student responses. In the following chapters, we present all the evaluation data sets we are going to use and our new summarization approaches which summarize student responses at a sentence level and a phrase level respectively.

In chapter 3, we introduce the evaluation corpora, including data sets from three different sources: student responses from four different courses, one benchmark of news articles, and three sets of reviews.

In chapter 2, we introduce related work about fundamental summarization background, state-of-the-art systems, and summarization evaluation and annotation.

In chapter 4, we first propose a new approach to summarizing student course feedback based on the integer linear programming (ILP) framework. We explore different factors that impact the performance of the proposed model. We perform extensive experiments on a number of data sets to provide insights on why and when the model works. Experimental results show that our approach is promising to summarize student feedback on two courses in terms of both ROUGE scores and human evaluation

In chapter 5, we present a summarization algorithm at a phrase level that differs from traditional methods in two ways (Luo and Litman, 2015). First, since the linguistic units of student inputs range from single words to multiple sentences, our summaries are created from extracted phrases rather than from sentences. Second, the phrase summarization algorithm ranks the phrases by the number of students who semantically mention a phrase in a summary. Experimental results on student responses from all courses show the effectiveness of the proposed approach.

In chapter 6, we first introduce the limitations of the phrase summarization proposed above. To address such limitations, we introduce a new phrase-based highlighting scheme for automatic summarization. Enabled by the highlighting scheme, we improve the phrase-based summarization framework proposed by Luo and Litman (2015) by developing a supervised candidate phrase extraction, learning to estimate the phrase similarities, and experimenting with different clustering algorithms to group phrases into clusters. We further introduce a new metric that offers a promising direction for making progress on developing automatic summarization evaluation metrics. Experimental results show that our proposed methods not only yield better summarization performance evaluated using ROUGE, but also produce summaries that capture the pressing student needs.

Finally, chapter 7 and chapter 8 present the possible future directions and summarize the major contributions of this work.

2.0 RELATED WORK

The challenge of *information overload* has triggered the research of automatic summarization in the community of natural language processing (NLP). It is the task of taking an input of text/speech documents and producing a concise summary of the most important information of the original documents (Nenkova and McKeown, 2011).

Existing studies on summarization can be broadly divided into sentence extraction (Martins and Smith, 2009; Berg-Kirkpatrick et al., 2011; Li et al., 2013a) and document abstraction (Liu et al., 2015; Rush et al., 2015; Durrett et al., 2016; Nallapati et al., 2016). Abstractive approaches build an internal semantic representation from the input text and leverage natural language generation techniques to create a summary (Li et al., 2013a; Liu et al., 2015). An abstract is close to what a human might produce, and it may contain words that are not present in the original. These models draw on recent developments of neural language models and the attention mechanisms (Rush et al., 2015; Chopra et al., 2016; Nallapati et al., 2016). On the downside, a large amount of paired training data (e.g., document+summary), in the scale of millions of data instances, are required to train the models in an end-to-end fashion. This enabling factor can sometimes be difficult to achieve.

Extractive approaches focus on extracting textual units from the input documents. Frequently, sentences are extracted from input documents according to two criteria: the summary, realized as a collection of sentences, is expected to 1) maximize the coverage of important content contained in the original documents, and 2) minimize redundancy in the summary. Because the summary is restricted in length, a compression step can be optionally applied to the sentences to further remove irrelevant or redundant constituents. For example, “FBI says” may be removed from the sentence “Airport shooter did it for ISIS, *FBI says*.” Subordinate clauses, prepositional phrases, adverbs, etc. are often removed in this process. Notable extractive systems include maximal marginal relevance (Carbonell and Goldstein, 1998), submodular functions (Lin and Bilmes, 2010), jointly extract and compress sentences (Zajic et al., 2007), optimize content selection and surface realization (Woodsend and Lapata, 2012), minimize reconstruction error (He et al., 2012), and dual decomposition (Almeida and Martins, 2013).

2.1 MULTIPLE GRANULARITIES

Work on automatic text summarization involves multiple granularities, ranging from keywords, phrases, to sentences. Traditional approaches have largely focused on sentence extraction (Martins and Smith, 2009; Berg-Kirkpatrick et al., 2011; Li et al., 2013a) and document abstraction (Liu et al., 2015; Rush et al., 2015; Durrett et al., 2016; Nallapati et al., 2016). In both cases, the produced summary is expected to be cohesive and coherent. We deviate from this path and seek to directly generate a set of bullet points as a summary.

While summarization systems that extract sentences are dominant, others have published in "summarization" at other levels besides the sentence. For example, Ueda et al. (2000) developed an "at-a-glance" summarization method with handcrafted rules. Recently, keyphrase extraction (Wu et al., 2005; Liu et al., 2009; Medelyan et al., 2009; Hasan and Ng, 2014; Kan, 2015) has received considerable attention, aiming to select important phrases from input documents, which is similar to phrase summarization. In this paper, we propose a general framework to adapt sentence summarization to phrase summarization. However, our task setting differs from those of keyphrase extraction. Of key importance is that each summary phrase is associated with a numerical value, indicating the number of students who raise the issue. This information is critical to course instructors for making informed choices. Intuitively our task setting bears similarity to word/phrase clouds (Yatani et al., 2011; Brooks et al., 2014), where the cloud gives greater prominence to words or phrases that appear frequently in the source text. The downside is that they do not take lexical variety into account or consider semantically-equivalent words/phrases.

2.2 STATE OF THE ART

We will use the following state of the art methods as competitive baselines in my experiments.

MEAD is a centroid-based summarization system that scores sentences based on length, centroid, and position (Radev et al., 2004).

LexRank is a graph-based summarization approach based on eigenvector centrality (Erkan and Radev, 2004).

SumBasic (Vanderwende et al., 2007) is an approach that assumes words occurring frequently in a document cluster have a higher chance of being included in the summary.

ILP-based framework is an important strand of extractive summarization research. It has demonstrated substantial success on summarizing news documents (Gillick et al., 2008, 2009; Woodsend and Lapata, 2012; Li et al., 2013b, 2016a). Previous studies attempted to improve this line of work by generating better estimates of concept weights. Galanis et al. (2012) proposed a support vector regression model to estimate bigram frequency in the summary. Berg-Kirkpatrick et al. (2011) explored a supervised approach to learn parameters using a cost-augmentative SVM. Our work is different from the above approaches in that we focus on improving the word co-occurrence matrix instead of concept weights, which is another important component of the ILP framework.

MMR (Carbonell and Goldstein, 1998) is a popular diversity-based summarization method, which can be used as a post-processing step to remove redundancy in the summary.

Clustering has been used to score sentences and has shown good improvement in text summarization (Gung and Kalita, 2012; Yang et al., 2012; Li and Li, 2014). In this work, we are using a metric clustering with semantic similarity to estimate the student coverage at a phrase level. Similarly, both diversity-based summarization (Carbonell and Goldstein, 1998; Zhang et al., 2005; Zhu et al., 2007) and our proposed method aim to estimate and maximize student coverage by minimizing redundancy in the output phrases. Differently, our method performs the redundancy reduction at a cluster level (a group of phrases) rather than penalize redundancy with a greedy iterative procedure sentence by sentence, and not only the information content is considered, but also the information source.

2.3 EVALUATION

There is a debate about how to judge summarization quality. However, ROUGE has been quickly adopted in many research papers and is a standard metric to evaluate the quality of summarization because it is fast and is correlated well to human evaluation (Lin, 2004; Graham, 2015). ROUGE (Lin, 2004) measures the n-gram overlap between system and human reference summaries. The recall, precision, and F-measure of ROUGE-N are computed as follows:

$$R_{R-N} = \frac{\sum_{S \in ReferenceSummaries} \sum_{gram_n \in S} Count_{match}\left(gram_n\right)}{\sum_{S \in ReferenceSummaries} \sum_{gram_n \in S} Count\left(gram_n\right)}, \tag{2.1}$$

$$P_{R-N} = \frac{\sum_{S \in ReferenceSummaries} \sum_{gram_n \in S} Count_{match}\left(gram_n\right)}{\sum_{S \in SystemSummary} \sum_{gram_n \in S} Count\left(gram_n\right)}, \tag{2.2}$$

$$F_{R-N} = \frac{\left(1+\beta^2\right) R_{R-N} P_{R-N}}{R_{R-N} + \beta^2 P_{R-N}}, \tag{2.3}$$

N is the length of the n-gram, $gram_n$ is an n-gram with length n, $Count\left(gram_n\right)$ is the number of n-grams, $Count_{match}\left(gram_n\right)$ is the maximum number of n-grams co-occurring in a candidate summary and a set of reference summaries, and β controls the relative importance of P_{R-N} and R_{R-N}.

At the same time, it is also criticized that ROUGE cannot thoroughly capture the semantic similarity between system and reference summaries. Therefore, many researchers supplement ROUGE with a manual evaluation.

Pyramid (Nenkova and Passonneau, 2004) is a human evaluation method by creating clusters of similar phrases to represent Summary Content Units (SCU) from human reference summaries. This annotation is semantically driven but it is very labor intensive.

Recently, human evaluation using the crowdsourcing platform Amazon Mechanical Turk (AMT)[1] is becoming an alternate method considered by researchers (Gorinski and Lapata, 2015; Kiddon et al., 2016; Durrett et al., 2016).

[1] www.mturk.com

In this work, we mainly use ROUGE as the evaluation metric. We report ROUGE-1 and ROUGE-2 as they are typically used in existing literature and found correlation to human evaluation, especially for informal user-generated content (Liu and Liu, 2010). We also supplement it human evaluation using AMT. Pyramid is not used because it is hard to scale due to its intensive labor and our evaluation is considerably large.

2.4 SUMMARY ANNOTATION

Traditional approaches to summary annotation have been based on either sentence extracts or document abstracts (Loza et al., 2014; Xiong and Litman, 2014; Wang and Ling, 2016). An effective linkage between the document content and human summary on the micro level have been largely absent. Barker et al. (2016) partially address this challenge by linking a summary back to a group of sentences that support the summary. However, this linkage is weak since it tells only that there is one sentence or more supporting the summary within the group, without explicitly telling which one(s).

Approaches such as Pyramid (Nenkova and Passonneau, 2004) have exploited creating Summary Content Units (SCUs) to establish such links and alleviate the challenge. The new highlighting scheme described in this work holds promise for establishing direct links between the phrases in student responses and those in the human summary, allowing us to develop a new evaluation metric based on color matching.

2.5 REFLECTION FROM AN EDUCATION PERSPECTIVE

In this work, we are considering one particular type of student responses, named "*reflective feedback*" (a.k.a. "muddy cards" (Mosteller, 1989a) or "one-minute papers" (Harwood, 1996)) , which has been shown to enhance interaction between instructors and students by educational research (Van den Boom et al., 2004; Menekse et al., 2011). In a typical deployment of reflection prompts, students are given index cards at the end of each lecture and are encouraged to reflect on what was confusing in the lecture. After collecting responses from students, the instructor summarizes the student reflections, identifies major misunderstandings, and plans follow-up actions, such as providing feedback in the following lectures, and tailoring the teaching plan in the future. Previous studies in different domains (Baird et al., 1991; Aleven and Koedinger, 2002; Van den Boom et al., 2004; Menekse et al., 2011; Glassman et al., 2015) consistently confirmed that reflective activities could benefit students by enhancing their retention and comprehension in learning. However, it is time consuming for instructors to summarize and understand of the raw response data (Mosteller, 1989b) and thus providing feedback to students based on such reflection is typically delayed. This is becoming more severe in large courses (e.g., introductory STEM, MOOCs). In this work, we automatically summarize student reflective responses so as to provide immediate summary both to students and instructors and to address the scalability issue to large classrooms.

3.0 DATA SETS

This chapter introduces distinct data sets that we are going to use in this work, including student response data sets from four different courses, three sets of reviews, and one benchmark of news articles. The corpora are summarized in Table 1.1.

As far as we know, the student response Eng was the first kind of student response summarization data set, collected by Menekse et al. (2011). The responses were collected by paper-based surveys after each lecture and the human summaries were created by the teaching assistant. However, this data set is limited to one course and one human annotator. To collect more data for different courses, we design and implement a mobile application, CourseMIRROR (Luo et al., 2015; Fan et al., 2015, 2017) to collect and share student feedback in a large scale. We annotated student responses with human summaries from three different courses (Stat2015, Stat2016 and CS2016) collected by the mobile application, allowing us to test the generalizability of proposed methods.

Although our main focus is to summarize student responses, we expect that our proposed methods can be applied to other types of data. In news articles and online reviews, there is a lexical variety challenge as well. For example, people like to use nicknames like “the Bronx Zoo” or “New York Highlanders” for the baseball team “New York Yankees”. Automatic summarization systems should identify such varieties. In addition, a length variety issue also exists in review data sets. For example, when people want to express how they like a movie, they may use a single word like “A++”, a few words like “love this movie”, a sentence or clause(s) like “Well done, well acted, and well directed”, or multiple sentences. Therefore, we also collect news articles and reviews data sets and want to apply our methods to them for generalizability testing. Unfortunately, they do not have summaries annotated at a phrase level, therefore, we use the news and review data sets only for our sentence-level summarization method.

3.1 STUDENT RESPONSES: ENG

The Eng student response corpus was first collected by Menekse et al. (2011) and a subset is made public by us (Luo and Litman, 2015), available at the link: http://www.coursemirror.com/download/dataset. It consists of student responses collected from 53 undergraduates enrolled in an introduction to materials science and engineering class in Spring 2011 (henceforth **Eng**). The students were asked to complete a survey at the end of each of 25 lectures during a semester, consisting of three carefully designed reflection prompts:

- Point of Interest (POI): "Describe what you found most interesting in today's class."
- Muddiest Point (MP): "Describe what was confusing or needed more detail."
- Learning Point (LP): "Describe what you learned about how you learn."

In total, more than 900 responses were collected for each prompt. If we concatenate all the responses to each lecture and prompt into a "pseudo-document", the document contains 375 words on average. The reference summaries are created by a teaching assistant. She is allowed to create abstract summaries using her own words in addition to selecting phrases directly from the responses. 48.8% of the bigrams in human summaries appear in the responses. Because summary annotation is costly and recruiting annotators with a proper background is nontrivial, 12 out of the 25 lectures are annotated with reference summaries. The summaries include not only the important phrases, but also the number of students who mentioned them (i.e., student supporters). Additional external resources are also available, including the lecture slides and textbook (Callister and Rethwisch, 2010).

An example of student responses to "Point of Interest" and the corresponding human summary is illustrated in Table 1.2. Another example for "Muddiest Point" is shown in Appendix A. The statistics of the student responses and the human's reference summaries are shown in Table 3.1. The phrases summarized by the TA are significantly shorter than the student responses (WC-Student vs. PWC-Human, p<0.01).

		min	max	mean	std
Eng	WC-Student	1	91	9.2	7.3
	PWC-Human	1	26	7.1	4.9
	WC-Human	6	103	29.4	23.2
	PC-Human	2	12	4.2	2.2
Stat2015	WC-Student	1	45	6.2	6.0
	PWC-Human	1	10	3.1	1.7
	WC-Human	5	36	15.1	5.7
	PC-Human	2	5	4.9	0.5
Stat2016	WC-Student	1	86	3.9	4.3
	PWC-Human	1	10	2.7	1.6
	WC-Human	6	24	13.3	3.1
	PC-Human	5	5	5.0	0.0
CS2016	WC-Student	1	91	10.0	10.6
	PWC-Human	1	11	3.3	2.1
	WC-Human	5	35	16.4	5.7
	PC-Human	3	5	4.9	0.3

Table 3.1: Number of words in student responses and human summaries. WC-Student is the **w**ord **c**ount of a **student** response; PWC-Human is the **w**ord **c**ount per **p**hrase in **human** summaries; WC-Human is the **w**ord **c**ount of **human** summaries; PC-Human is the **p**hrase **c**ount of **human** Summaries.

3.2 STUDENT RESPONSES: STAT2015, STAT2016, CS2016

These three data sets were collected by us with the mobile application, CourseMIRROR (Luo et al., 2015; Fan et al., 2015, 2017). The Stat2015 and Stat2016 data sets were from the same course, Statistics for Industrial Engineers, but taught in 2015 and 2016 respectively (henceforth **Stat2015** and **Stat2016**), at the Boğaziçi University in Turkey.[1] The course was taught in English while the official language is Turkish. The CS2016 data set is from a fundamental undergraduate Computer Science course (data structures) at the University of Pittsburgh taught in 2016 (henceforth **CS2016**).

After each lecture, the students were asked to respond to two reflection prompts using CourseMIRROR: 1) "Describe what you found most interesting in today's class," and 2) "Describe what was confusing or needed more detail." For each course, two independent human annotators (native English speakers) with a proper background were recruited to create summaries for each lecture and prompt. The summarization annotation task was paid at a rate of $25 per lecture. For each lecture and prompt, each annotator will create three different types of summarization. When creating the summaries, the annotators are told to imagine themselves as a TA for the course, by assuming what they want to present to the instructor after reading the students' responses. The instruction given to the annotators for each task is introduced as follows.

Task 1: Extractive Summarization. Select five most representative sentences in order as the summary. (Use the sentence index number.)

Task 2: Abstractive Summarization. Given the students' responses, create a short summary using your own words (about 40 words, no specific format other than linear).

Task 3: Phrase Summarization. Create a summary using 5 phrases together with how many students semantically mentioned each phrase. You can use your own phrases.

Annotators are also asked to highlight where the summary phrases come from for the phrase summarization. Here is the instruction: "please also highlight the corresponding phrases in the student responses above which are semantically same to the summary phrases using the highlighted colors in the first row in the table below. The number of highlights for each phrase should match the number of students who semantically mentioned the phrase."

A sample annotated summarization is shown in Appendix B.

In this work, we use only the Phrase Summarization annotations. We leave the opportunities to use other annotations to future work.

[1] Publicly available at http://www.coursemirror.com/download/dataset2 (Luo et al., 2016a)

3.3 PRODUCT AND PEER REVIEWS

The review data sets are provided by Xiong and Litman (2014), consisting of 3 categories. The first one is a subset of product reviews from a widely used data set in review opinion mining and sentiment analysis, contributed by Jindal and Liu (2008). In particular, it randomly sampled 3 sets of reviews from a representative product (digital camera), each with 18 reviews from an individual product type (e.g. “summarizing 18 camera reviews for Nikon D3200”). The second one is movie reviews crawled from IMDB.com by the authors themselves. The third one is peer reviews collected in a college level history class from an online peer-review reciprocal system, SWoRD (Cho, 2008). The average number of sentences per review set is 85 for camera reviews, 328 for movie reviews and 80 for peer review; the average number of words per sentence in the camera, movie, and peer reviews are 23, 24 and 19, respectively. The human summaries were collected in the form of online surveys (one survey per domain) hosted by Qualtrics. Each human summary contains 10 sentences from users’ reviews. Example movie reviews are shown in Table 3.2.

“Forrest Gump” is one of the best movies of all time, guaranteed.

I just love this movie.

It truly is amazing...

What an amazing story and moving meaning.

I am not kidding, “Forrest Gump” is a remarkable movie and inspires everyone.

I really just love this movie and it has such a special place in my heart.

And anyone who hasn’t seen it or who thinks that don’t like it I seriously suggest seeing it or seeing it again.

The brilliant humour, the hilarious yet touching acting, the special effects and the uplifting message are totally rewarding.

That movie teaches you so much about life and the meaning of it.

This is one masterpiece of a movie that will not be forgotten about in a long time.

This is a powerful yet charming movie; fun for its special effects and profound in how it keeps you thinking long after it’s over.

It may change your lifeOne hell of a movie; it will be close to my heart forever!

It is something to mull over for a long time.

The performances are just so unforgettable and never get out of your head.

I've watched the movie about once every two years since then.

The lines are so memorable, touching, and sometimes hilarious.We have Forrest Gump (Tom Hanks), not the sharpest tool in the box, his I.Q.

Well done, well acted, and well directed to pythagorean procision. A++

This story is beautiful and will inspire everyone to go the distance and see the world like Forrest did and will never give up on their dreams.10/10

A++

You 'd be a fool to miss it.Bottom Line : 4 out of 4 (own this movie)

Table 3.2: Example movie reviews.

3.4 NEWS ARTICLES: DUC04

Most summarization work focuses on news documents, as driven by the Document Understanding Conferences (DUC) and Text Analysis Conferences (TAC). For comparison, we select DUC 2004[2] to evaluate our approach (henceforth **DUC04**), which is widely used in the literature (Lin, 2004; Hong et al., 2014; Ren et al., 2016; Takase et al., 2016; Wang et al., 2016). It consists of 50 clusters of Text REtrieval Conference (TREC) documents, from the following collections: AP newswire, 1998-2000; New York Times newswire, 1998-2000; Xinhua News Agency (English version), 1996-2000. Each cluster contained on average 10 documents. The task is to create a short summary ($\leq$ 665 bytes) of each cluster. Example news sentences are shown in Table 3.3.

[2] http://duc.nist.gov/duc2004/

Samaranch expressed surprise at allegations made by the IOC executive board member Marc Hodler of Switzerland that agents were offering to sell I.O.C. members' votes for payments from bidding cities.

Moving quickly to tackle an escalating corruption scandal, IOC leaders questioned Salt Lake City officials Friday in the first ever investigation into alleged vote-buying by an Olympic city.

Acting with unusual speed, the International Olympic Committee set up a special investigative panel that immediately summoned the organizers of the 2002 Salt Lake Games to address the bribery allegations.

It's the most serious case of alleged ethical misconduct investigated by the IOC since former U.S. member Robert Helmick was accused of conflict of interest in 1991.

This is the first time the IOC has ever investigated possible bribery by bidding cities, despite previous rumors and allegations of corruption in other Olympic votes.

Hodler said a group of four agents, including one IOC member, have been involved in promising votes for payment.

Samaranch Sunday ruled out taking the Games from Salt Lake City.

I can't be stronger in saying I don't consider it a possibility whatsoever of the games being withdrawn from Salt Lake City.

The chief investigator refused to rule out the possibility of taking the games away from Salt Lake City - though that scenario is considered highly unlikely.

Table 3.3: Example sentences from news.

3.5 USAGE OF DATA SETS

The usage of data sets is summarized in Table 3.4.

	H1.1	H1.2	H1.3	H2	H3.1	H3.2	H3.3
Student Response (Eng)		✔	✔	✔			
Student Response (Stat2015)	✔	✔	✔	✔	✔	✔	✔
Student Response (Stat2016)	✔	✔	✔	✔	✔	✔	✔
Student Response (CS2016)	✔	✔	✔	✔	✔	✔	✔
Review (camera)			✔				
Review (movie)			✔				
Review (peer)			✔				
News articles (DUC04)			✔				

Table 3.4: Usage of data sets.

3.6 SUMMARY

In this chapter, we introduced all the data sets that we are going to use, including student responses from four courses, three set of reviews and one benchmark set of news articles. We also introduced how they will be used. For the sentence-level summarization, we will use all the data sets. For the phrase-based summarization, we will use all student response data sets. For the quantitative phrase summarization, we will use the three student response data sets that have the phrase-highlighting annotations.

4.0 SENTENCE SUMMARIZATION BY AN EXTENDED-ILP FRAMEWORK

This chapter introduces a new approach to summarizing student course feedback at the sentence level by extending the Integer Linear Programming (ILP) framework (Luo et al., 2016b).

As mentioned in §1, one of the challenges of summarizing student responses is its lexical variety. For example, in Table 1.2, "bike elements" (S11) and "bicycle parts" (S36), "the main topics of this course" (S12) and "what we will learn in this class" (S26) are different expressions that communicate the same or similar meanings. In particular, we observe 97% of the bigrams appear only once or twice in the student response data sets (§3.1,§3.2), whereas in a typical news data set (DUC 2004), it is about 80%.

The high lexical diversity issue can cause problems to the ILP framework. With high lexical diversity, the word co-occurrence matrix does not faithfully reflect if certain concepts (instead of words) are contained in the sentences, thus causing confusion to redundancy removal. To tackle this challenge, we propose a new approach to automatic summarization, which extends the standard ILP framework by approximating the co-occurrence matrix with a low-rank alternative. The resulting system allows different sentences to share co-occurrence statistics. For example, "The activity with the *bicycle parts*" will be allowed to partially contain "*bike elements*" although the latter did not appear in the sentence. The low-rank matrix approximation offers a domain-specific way of calculating "partial counts." It is not constrained by out-of-vocabulary terms and is a more principled approach than heuristically calculating similarities of word embeddings.

The contributions for this work are two-fold. First, we propose a novel improvement to the ILP framework for automatic summarization by introducing a low-rank approximation to the word co-occurrence matrix. It helps tackle the high lexical diversity issue. Second, we explore different factors that impact the performance of the proposed model. We perform extensive experiments on a number of datasets, ranging from student course feedback and product reviews to news reports, to provide insights on why and when the model works.

4.1 ILP FORMULATION

Let $\mathcal{D}$ be a set of documents that consist of M sentences in total. Let $y_j \in \{0, 1\}$, $j = \{1, \cdots, M\}$ indicate if a sentence j is selected ($y_j = 1$) or not ($y_j = 0$) in the summary. Similarly, let N be the number of unique concepts in $\mathcal{D}$. $z_i \in \{0, 1\}$, $i = \{1, \cdots, N\}$ indicate the appearance of concepts in the summary. Each concept i is assigned a weight of w_i, often measured by the number of sentences or documents that contain the concept. The ILP-based summarization approach (Gillick and Favre, 2009) searches for an optimal assignment to the sentence and concept variables so that the selected summary sentences maximize coverage of important concepts. The relationship between concepts and sentences is captured by a co-occurrence matrix $A \in \mathbb{R}^{N \times M}$, where $A_{ij} = 1$ indicates the i-th concept appears in the j-th sentence, and $A_{ij} = 0$ otherwise. In the literature, bigrams are frequently used as a surrogate for concepts (Gillick et al., 2008; Berg-Kirkpatrick et al., 2011). We follow the convention and use 'concept' and 'bigram' interchangeably in the thesis.

Two sets of linear constraints are specified to ensure the ILP validity: (1) a concept is selected if and only if at least one sentence carrying it has been selected (Eq. 4.2), and (2) all concepts in a sentence will be selected if that sentence is selected (Eq. 4.3). Finally, the selected summary sentences are allowed to contain a total of L words or less (Eq. 4.4).

$$\max_{\boldsymbol{y},\boldsymbol{z}} \quad \textstyle\sum_{i=1}^{N} w_i z_i \tag{4.1}$$

$$s.t. \quad \textstyle\sum_{j=1}^{M} A_{ij}\, y_j \geq z_i \tag{4.2}$$

$$A_{ij}\, y_j \leq z_i \tag{4.3}$$

$$\textstyle\sum_{j=1}^{M} l_j y_j \leq L \tag{4.4}$$

$$y_j \in \{0,1\}, z_i \in \{0,1\} \tag{4.5}$$

The above ILP can be transformed to matrix representation:

$$\max_{\boldsymbol{y},\boldsymbol{z}} \quad \boldsymbol{w}^\top \boldsymbol{z} \tag{4.6}$$

$$s.t. \quad \boldsymbol{A}\,\boldsymbol{y} \geq \boldsymbol{z} \tag{4.7}$$

$$\boldsymbol{A}\,\mathrm{diag}(\boldsymbol{y}) \leq \boldsymbol{Z} \tag{4.8}$$

$$\boldsymbol{\eta}^\top \boldsymbol{y} \leq L \tag{4.9}$$

$$\boldsymbol{y} \in \{0,1\}^M \tag{4.10}$$

$$\boldsymbol{z} \in [0,1]^N \tag{4.11}$$

We use boldface letters to represent vectors and matrices. $\boldsymbol{Z} = [\boldsymbol{z}, ..., \boldsymbol{z}] \in \mathbb{R}^{N \times M}$ is an auxiliary matrix created by horizontally stacking the concept vector $\boldsymbol{z} \in \mathbb{R}^N$ M times. Constraint set (Eq. 4.8) specifies that a sentence is selected indicates that all concepts it carries have been selected. It corresponds to $N \times M$ constraints of the form $A_{i,j}\, y_j \leq z_i$, where $i \in [N], j \in [M]$. As far as we know, this is the first-of-its-kind matrix representation of the ILP framework. It clearly shows the two important components of this framework, including 1) the concept-sentence co-occurrence matrix $\boldsymbol{A}$, and 2) concept weight vector $\boldsymbol{w}$. Existing work focus mainly on generating better estimates of concept weights ($\boldsymbol{w}$), while we focus on improving the co-occurrence matrix $\boldsymbol{A}$.

4.2 OUR APPROACH

Because of the lexical diversity problem, we suspect the co-occurrence matrix $\boldsymbol{A}$ may not establish a faithful correspondence between sentences and concepts. A concept may be conveyed using multiple bigram expressions; however, the current co-occurrence matrix only captures a binary relationship between sentences and bigrams. For example, we ought to give partial credit to "bicycle parts" given that a similar expression "bike elements" appears in the sentence. Domain-specific synonyms may be captured as well. For example, the sentence "I tried to follow along but I couldn't *grasp the* concepts" is expected to partially contain the concept "understand the", although the latter did not appear in the sentence.

The existing matrix $\boldsymbol{A}$ is highly sparse. Only 3.7% of the entries are non-zero in the student response data sets on average (§6.1). We therefore propose to *impute* the co-occurrence matrix by filling in missing values (i.e., matrix completion). This is accomplished by approximating the original co-occurrence matrix using a low-rank matrix. The low-rankness encourages similar concepts to be shared across sentences. The low-rank approximation process makes two notable changes to the existing ILP framework. First, it extends the domain of A_{ij} from binary to a continuous scale $[0, 1]$ (Eq. 4.2), which offers a better sentence-level semantic representation. The binary concept variables (z_i) are also relaxed to continuous domain $[0, 1]$ (Eq. 4.11), which allows the concepts to be "partially" included in the summary.

Concretely, given the co-occurrence matrix $\boldsymbol{A} \in \mathbb{R}^{N \times M}$, we aim to find a low-rank matrix $\hat{\boldsymbol{A}} \in \mathbb{R}^{N \times M}$ whose values are close to $\boldsymbol{A}$ at the observed positions. Our objective function is

$$\min_{\hat{A} \in \mathbb{R}^{N \times M}} \frac{1}{2} \sum_{(i,j) \in \Omega} (A_{ij} - \hat{A}_{ij})^2 + \lambda \left\| \hat{A} \right\|_*, \tag{4.12}$$

where Ω represents the set of observed value positions. $\|\hat{A}\|_*$ denotes the trace norm of $\hat{A}$, i.e., $\|\hat{A}\|_* = \sum_{i=1}^{r} \sigma_i$, where r is the rank of $\hat{A}$ and σ_i are the singular values. By defining the following projection operator P_Ω,

$$[P_\Omega(\hat{A})]_{ij} = \begin{cases} \hat{A}_{ij} & (i,j) \in \Omega \\ 0 & (i,j) \notin \Omega \end{cases} \tag{4.13}$$

our objective function (Eq. 4.12) can be succinctly represented as

$$\min_{\hat{A}\in\mathbb{R}^{N\times M}} \frac{1}{2}\|P_\Omega(A) - P_\Omega(\hat{A})\|_F^2 + \lambda\|\hat{A}\|_*, \tag{4.14}$$

where $\|\cdot\|_F$ denotes the Frobenius norm.

Following Mazumder et al. (2010), we optimize Eq. 4.14 using the proximal gradient descent algorithm. The update rule is

$$\hat{A}^{(k+1)} = \text{prox}_{\lambda\rho_k}\Big(\hat{A}^{(k)} + \rho_k\big(P_\Omega(A) - P_\Omega(\hat{A})\big)\Big), \tag{4.15}$$

where ρ_k is the step size at iteration k and the proximal function $\text{prox}_t(\hat{A})$ is defined as the singular value soft-thresholding operator, $\text{prox}_t(\hat{A}) = U \cdot \text{diag}((\sigma_i - t)_+) \cdot V^\top$, where $\hat{A} = U\text{diag}(\sigma_1, \cdots, \sigma_r)V^\top$ is the singular value decomposition (SVD) of $\hat{A}$ and $(x)_+ = \max(x, 0)$.

Since the gradient of $\frac{1}{2}\|P_\Omega(A) - P_\Omega(\hat{A})\|_F^2$ is Lipschitz continuous with $L = 1$ (L is the Lipschitz continuous constant), we follow Mazumder et al. (2010) to choose fixed step size $\rho_k = 1$, which has a provable convergence rate of $O(1/k)$, where k is the number of iterations.

4.3 EXPERIMENTS

In this section, we evaluate the proposed method intrinsically in terms of whether the co-occurrence matrix after the low-rank approximation is able to capture similar concepts on student response data sets, and also extrinsically in terms of the end task of summarization on all corpora. In the following experiments, summary length is set to be the average number of words in human summaries or less. For the matrix completion algorithm, we perform grid search (on a scale of [0, 5] with stepsize 0.5) to tune the hyper-parameter λ (Eq. 4.12) with a leave-one-lecture-out (for student responses) or leave-one-task-out (for others) cross-validation.

Sentence	**Assoc. Bigrams**
the printing needs to better so it can be easier to read	*the graph*
graphs make it *easier to* understand concepts	*hard to*
the naming system for the 2 *phase regions*	*phase diagram*
I tried to follow along but I couldn't *grasp the* concepts	*understand the*
no problems except for the specific equations used to determine properties from the stress - *strain graph*	*strain curves*
why delete the first entry in the *linked bag* instead of just moving the pointers from the node before the deleted node to the node after	*linked list*
You make *a movie* that romanticizes the '50's, '60's and '70's, and with enough publicity and a good enough soundtrack ...	*the film*
U.S. officials have said the construction ...	*united states*
American officials have said spy satellites ...	*united states*
It also sought to cast *Gates* as an obsessed man who feared the tiny Netscape Communications Corp. and its potential threat to his domination of the market for *Internet browsers*, the software used to navigate the World Wide Web.	*that microsoft*

Table 4.1: Associated bigrams that do not appear in the sentence, but after Matrix Completion, yield a decent correlation (cell value greater than 0.9) with the corresponding sentence.

4.3.1 Intrinsic evaluation

When examining the imputed sentence-concept co-occurrence matrix, we notice some interesting examples that indicate the effectiveness of the proposed approach, shown in Table 4.1.

We want to investigate whether the matrix completion (MC) helps to capture similar concepts (i.e., bigrams) (H1 in §1.2). Recall that, if a bigram i is similar to another bigram in a sentence j, the sentence j should assign a partial score to the bigram i after the low-rank approximation. For instance, "The activity with the bicycle parts" should give a partial score to "bike elements" since it is similar to "bicycle parts". Note that, the co-occurrence matrix $\boldsymbol{A}$ measures whether a sentence includes a bigram or not. Without matrix completion, if a bigram i does not appear in a sentence j, $A_{ij} = 0$. After matrix completion, $\hat{A}_{ij}$ ($\hat{A}$ is the low-rank approximation matrix of A) becomes a continuous number ranging from 0 to 1 (negative values are truncated). Therefore, $\hat{A}_{ij} > 0$ does not necessarily mean the sentence contains a similar bigram, since it might also give positive scores to non-similar bigrams. To solve this issue, we propose two different ways to test whether the matrix completion really helps to capture similar concepts.

- H1.1a: A bigram receives more partial score in a sentence that contains similar bigram(s) to it than a sentence that does not. That is, if a bigram i is similar to one of bigrams in a sentence j^+, but not similar to any bigram in another sentence j^-, then after matrix completion, $\hat{A}_{ij^+} > \hat{A}_{ij^-}$.
- H1.1b: A sentence gives more partial scores to bigrams that are similar to its own bigrams than bigrams that are different from its own. That is, if a sentence j has a bigram that is similar to i^+, but none of its bigrams is similar to i^-, then, after matrix completion, $\hat{A}_{i^+j} > \hat{A}_{i^-j}$.

In order to test these two hypotheses, we need to construct gold-standard pairs of similar bigrams and pairs of different bigrams, which can be automatically obtained with the phrase-highlighting data (Table 6.1). We first extract a candidate bigram from a phrase if and only if a single bigram can be extracted from the phrase. In this way, we discard long phrases if there are multiple candidate bigrams among them in order to avoid ambiguity as we cannot validate which of them match another target bigram. A bigram is defined as two words and at least one of them is not a stopword. We then extract every pair of candidate bigrams that are highlighted as the same color as similar bigrams. Similarly, we extract every pair of candidate bigrams that are highlighted as different colors as different bigrams. For example, “bias reduction” is a candidate phrase, which is similar to “bias correction” since they are in the same color.

To test H1.1a, given a bigram i, a bigram i^+ that is similar to it, and a bigram i^- that is different from it, we can select the bigram i, and the sentence j^+ that contains i^+, and the sentence j^- that contains i^-. We ignore j^- if it contains any other bigram that is similar to i to eliminate the compounded case that both similar and different bigrams are within one sentence. Note, if there are multiple sentences containing i^+, we consider each of them. In this way, we construct a triple $\langle i, j^+, j^- \rangle$, and test whether $\hat{A}_{ij^+} > \hat{A}_{ij^-}$. To test H1.1b, for each pair of similar bigrams $\langle i, i^+ \rangle$, and different bigrams $\langle i, i^- \rangle$, we select the sentence j that contains i so that we construct a triple $\langle i^+, i^-, j \rangle$, and test whether $\hat{A}_{i^+j} > \hat{A}_{i^-j}$. We also filtered out j that contains similar bigram(s) to i^- to remove the compounded effect. In this way, we collected a gold-standard data set to test the two hypotheses above as shown in Table 4.2.

Corpus	**bigrams**	**similar pairs**	**different pairs**	$\langle i, j^+, j^- \rangle$	$\langle i^+, i^-, j \rangle$
Stat2015	516	198	698	404	279
Stat2016	1,673	638	1,928	1,188	228
CS2016	613	168	412	235	46

Table 4.2: A gold-standard data set was extracted from three student response corpora that have phrase-highlighting annotation. Statistics include: the number of bigrams, the number of pairs of similar bigrams and pairs of different bigrams, the number of tuples $\langle i, j^+, j^- \rangle$, and the number of $\langle i^+, i^-, j \rangle$. i is a bigram, j^+ is a sentence with a bigram similar to i, and j^- is a sentence with a bigram different from i. j is a sentence, i^+ is a bigram that is similar to a bigram in j, and i^- is a bigram that is different from any bigram in j.

The results are shown in Table 4.3. $\hat{A}_{ij^+} > \hat{A}_{ij^-}$ significantly on all three courses. That

	Stat2015		Stat2016		CS2016	
H1.1a	$\hat{A}_{ij+}$	$\hat{A}_{ij-}$	$\hat{A}_{ij+}$	$\hat{A}_{ij-}$	$\hat{A}_{ij+}$	$\hat{A}_{ij-}$
	0.122^*	0.056	0.108^*	0.038	0.238^*	0.089
H1.1b	$\hat{A}_{i+j}$	$\hat{A}_{i-j}$	$\hat{A}_{i+j}$	$\hat{A}_{i-j}$	$\hat{A}_{i+j}$	$\hat{A}_{i-j}$
	0.147	0.151	0.132^*	0.074	0.186	0.149

Table 4.3: Hypothesise testing: whether the matrix completion (MC) helps to capture similar concepts. * means $p < 0.05$ using a two-tailed paired t-test.

is, a bigram does receive more partial score in a sentence that contains similar bigram(s) to it than a sentence that does not. Therefore, H1.1a holds. For H1.1b, we only observe $\hat{A}_{i+j} > \hat{A}_{i-j}$ significantly on Stat2016 and there is no significant difference between $\hat{A}_{i+j}$ and $\hat{A}_{i-j}$ on the other two courses. First, the gold-standard data set is still small in the sense that only a limited portion of bigrams in the entire data set are evaluated. Second, the assumption that phrases annotated by different colors are not necessarily unrelated is too strong. For example, “hypothesis testing” and “h0 and h1” are in different colors in the example of Appendix B, but one is a subtopic of the other. An alternative way to evaluate the hypothesis is to let humans judge whether two bigrams are similar or not, which we leave to future work.

4.3.2 Extrinsic evaluation

Our proposed approach is compared against a range of baselines. They are 1) MEAD (Radev et al., 2004), a centroid-based summarization system that scores sentences based on length, centroid, and position; 2) LexRank (Erkan and Radev, 2004), a graph-based summarization approach based on eigenvector centrality; 3) SumBasic (Vanderwende et al., 2007), an approach that assumes words occurring frequently in a document cluster have a higher chance of being included in the summary; 4) ILP (Berg-Kirkpatrick et al., 2011), a baseline ILP framework without matrix completion.

For the ILP-based approaches, we use bigrams as concepts (bigrams consisting of only stopwords are removed[1]) and term frequency as concept weights. We leverage the co-occurrence statistics both within and across the entire corpus[2]. We also filtered out bigrams that appear only once in each corpus, yielding better ROUGE scores with lower computational cost. The results without using this low-frequency filtering are shown in the Appendix C for comparison. In Table 4.4, we present summarization results evaluated by ROUGE (Lin, 2004) and human judges.[3]

ROUGE. It is a standard evaluation metric that compares system and reference summaries based on n-gram overlaps. In this work, we report recall, precision and F-measure[4] of R-1, and R-2 scores, which respectively measure the overlap of unigrams and bigrams. First, there is no winner for all data sets. MEAD is the best one on camera; SumBasic is best on Stat2016 and mostly on Stat2015; ILP is best on DUC04. Our method ILP+MC is best on peer review and mostly on Eng and CS2016. Second, compared with ILP, our method works better on Eng, CS2016, movie and peer. Back to our H1.2 in §1.2, the extended-ILP framework does not deliver better summarization performance than the traditional ILP-based framework on all student responses in terms of ROUGE scores. For H1.3 in §1.2, the extended-ILP framework cannot be directly applicable to news and camera review.

Human Evaluation. Because ROUGE cannot thoroughly capture the semantic similarity between system and reference summaries, we further perform human evaluation. For each task, we present a pair of system outputs in a random order, together with one human summary to five Amazon turkers. If there are multiple human summaries, we will present each human summary and the pair of system outputs to turkers. For student responses,

[1] Bigrams with one stopword are not removed because 1) they are informative ("a bike", "the activity", "how materials'); 2) such bigrams appear in multiple sentences and are thus helpful for matrix imputation.

[2] We construct one single matrix for each entire corpus except DUC04. For example, the co-occurrence matrix for Eng includes 1492 distinct sentences and 9239 unique bigrams, from all lectures and prompts. For DUC04, we construct a matrix for each document cluster instead of the entire corpus due to its high computational cost.

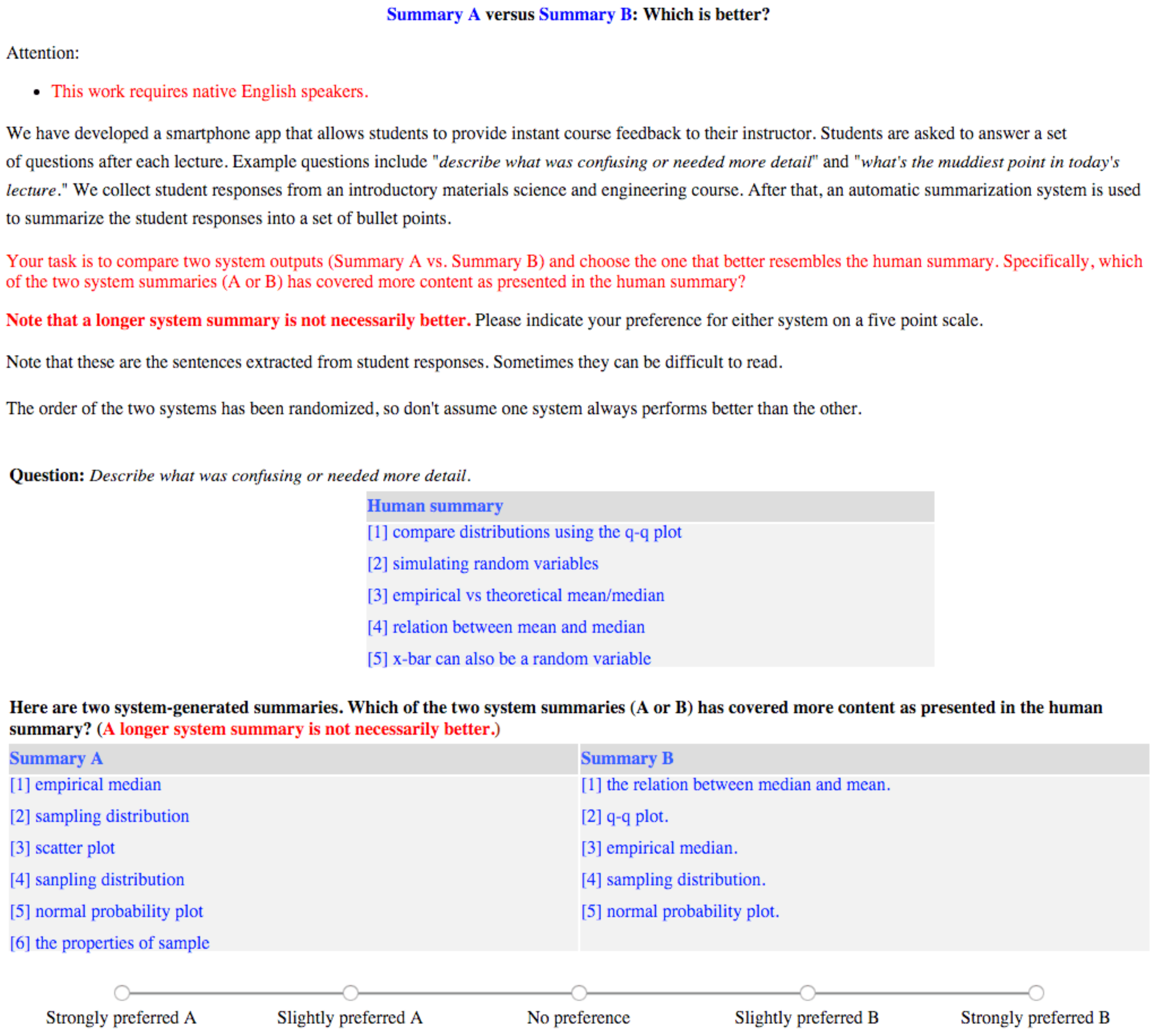
[3] The results on Eng are slightly different from the results published by Luo et al. (2016b) as we used leave-one-lecture-out cross-validation instead of 3 cross-validation to select the parameter λ. We also changed the order of student responses by grouping same responses together, affecting the position feature in MEAD.

[4] Some of F-measures are slightly lower than P/R because of the averaging effect and can be illustrated in one example. Suppose we have P1=0.1, R1=0.4, F1=0.16 and P2=0.4, R2=0.1, F2=0.16. Then the macro-averaged P/R/F are: P=0.25, R=0.25, F=0.16. In this case, the F-measure is lower than both P and R.

		R-1			**R-2**			**Human**
	System	R	P	F	R	P	F	**Preference**
Eng	MEAD	.192*	.179*	.161*	.052*	.054*	.046*	-
	LexRank	.303*	.286*	.262*	.093	.097	.087	-
	SumBasic	.387	**.337**	**.323**	.090*	.089*	.082*	26.9%
	ILP	.364*	.329	.308	.123	.124	.110	24.1%
	ILP+MC	**.392**	.335	.322	**.130**	**.127**	**.114**	**29.4%**
Stat2015	MEAD	.225*	.217*	.213*	.073*	.073*	.071*	-
	LexRank	.334*	.346	.325*	.154	.147	.142	-
	SumBasic	**.457***	**.424***	**.427***	**.193**	.169	**.174**	**30.7%**
	ILP	.405	.396	.390	.186	**.175**	**.174**	29.2%*
	ILP+MC	.401	.372	.375	.183	.164	.167	29.6%
Stat2016	MEAD	.364*	.419*	.378*	.172*	.213	.181	-
	LexRank	.397*	.431	.407*	.191	.209	.195	-
	SumBasic	**.554***	**.569***	**.557***	**.295***	**.298***	**.294***	**32.9%**
	ILP	.482	.516	.496	.262*	.283*	.270*	29.1%
	ILP+MC	.457	.489	.465	.214	.230	.218	28.0%
CS2016	MEAD	.221*	.190*	.195*	.056*	.050*	.050*	-
	LexRank	.285*	.296*	.282*	.085*	.089*	.084*	-
	SumBasic	**.408**	.408	**.398**	.141	.144	.139	31.5%
	ILP	.374	.408	.382	.141	.155	.144	24.4%*
	ILP+MC	.398	**.409**	.395	**.154**	**.156**	**.151**	**32.7%**
camera	MEAD	**.475**	**.478**	**.474**	**.207**	**.217**	**.211**	-
	LexRank	.439	.456	.446	.181	.188	.184	-
	SumBasic	**.475**	.472	.473	.168	.166*	.167	23.9%*
	ILP	.457	.466	.460	.165	.165	.165	**36.9%**
	ILP+MC	.447	.449	.447	.157	.158	.157	32.5%
movie	MEAD	.394	.408	.398	.131	.136	.132	-
	LexRank	.434*	.428	.417	**.147**	**.141**	**.139**	-
	SumBasic	**.441**	**.437**	**.437**	.098	.097	.097	27.6%*
	ILP	.435	.424	.427	.091*	.087*	.088*	**38.2%**
	ILP+MC	.436	.427	.429	.106	.100	.102	21.8%
peer	MEAD	.469	.494	.479	.242	.255	.248	-
	LexRank	.444	.461	.451	.196	.214	.204	-
	SumBasic	.473	.470	.471	.154*	.149	.151	23.3%
	ILP	.466	.469	.466	.199	.196	.197	**34.4%**
	ILP+MC	**.491**	**.496**	**.492**	**.261**	**.262**	**.260**	22.2%
DUC04	MEAD	.352	.354	.351	.076	.076	.076	-
	LexRank	.354	.364	.358	.076	.078	.077	-
	SumBasic	.364*	.365	.365*	.066	.066	.066	24.9%*
	ILP	**.377***	**.381***	**.379***	**.092***	**.093***	**.092***	27.3%*
	ILP+MC	.342	.351	.346	.072	.074	.072	**31.1%**

Table 4.4: Summarization results evaluated by ROUGE and human judges. Best results are shown in **bold** for each data set. * indicates that the performance difference with ILP+MC is statistically significant ($p < 0.05$) using a two-tailed paired t-test. Underline means that ILP+MC is better than ILP.

we also present the prompt. An example Human Intelligence Task (HIT) is illustrated in Fig. 4.1. Additional example HITs can be found in Appendix D.



Summary A versus Summary B: Which is better?

Attention:

- This work requires native English speakers.

We have developed a smartphone app that allows students to provide instant course feedback to their instructor. Students are asked to answer a set of questions after each lecture. Example questions include "*describe what was confusing or needed more detail*" and "*what's the muddiest point in today's lecture*." We collect student responses from an introductory materials science and engineering course. After that, an automatic summarization system is used to summarize the student responses into a set of bullet points.

Your task is to compare two system outputs (Summary A vs. Summary B) and choose the one that better resembles the human summary. Specifically, which of the two system summaries (A or B) has covered more content as presented in the human summary?

Note that a longer system summary is not necessarily better. Please indicate your preference for either system on a five point scale.

Note that these are the sentences extracted from student responses. Sometimes they can be difficult to read.

The order of the two systems has been randomized, so don't assume one system always performs better than the other.

Question: *Describe what was confusing or needed more detail.*

Human summary
[1] compare distributions using the q-q plot
[2] simulating random variables
[3] empirical vs theoretical mean/median
[4] relation between mean and median
[5] x-bar can also be a random variable

Here are two system-generated summaries. Which of the two system summaries (A or B) has covered more content as presented in the human summary? (A longer system summary is not necessarily better.)

Summary A	**Summary B**
[1] empirical median	[1] the relation between median and mean.
[2] sampling distribution	[2] q-q plot.
[3] scatter plot	[3] empirical median.
[4] sanpling distribution	[4] sampling distribution.
[5] normal probability plot	[5] normal probability plot.
[6] the properties of sample	

○ Strongly preferred A ○ Slightly preferred A ○ No preference ○ Slightly preferred B ○ Strongly preferred B

Figure 4.1: An example HIT from Stat2015, 'System A' is ILP+MC and 'System B' is SumBasic.

The turkers are asked to indicate their preference for system A or B based on the semantic resemblance to the human summary on a 5-Likert scale ('Strongly preferred A', 'Slightly preferred A', 'No preference', 'Slightly preferred B', 'Strongly preferred B'). They are rewarded $0.04 per task. We use two strategies to control the quality of the human evaluation. First, we require the turkers to have a HIT approval rate of 90% or above. Second, we insert some quality checkpoints by asking the turkers to compare two summaries of same text content but in different sentence orders. Turkers who did not pass these tests are filtered out. Due to budget constraints, we conduct pairwise comparisons for three systems. The total number of comparisons is 3 system-system pairs × 5 turkers × (36 tasks × 1 human summaries for Eng + 44×2 for Stat2015 + 48×2 for Stat2016 + 46×2 for CS2016 + 3×8 for camera + 3×5 for movie + 3×2 for peer + 50 × 4 for DUC04) = 8,355. The number of tasks for each corpus is shown in Table 1.1. To elaborate as an example, for Stat2015, there are 22 lectures and 2 prompts for each lecture. Therefore, there are 44 tasks (22×2) in total. In addition, there are 2 human summaries for each task. We selected three competitive systems (SumBasic, ILP, and ILP+MC) and therefore we have 3 system-system pairs (ILP+MC vs. ILP, ILP+MC vs. SumBasic, and ILP vs. SumBasic) for each task and each human summary. Therefore, we have 44×2×3=264 HITs for Stat2015. Each HIT will be done by 5 different turkers, resulting in 264×5=1,320 comparisons.

We calculate the percentage of "wins" (strong or slight preference) for each system among all comparisons with its counterparts. Results are reported in the last column of Table 4.4[5]. ILP+MC is preferred significantly[6] more often than ILP on Stat2015, CS2016, and DUC04. There is no significant difference between ILP+MC and SumBasic on student response data sets. Interestingly, a system with better ROUGE scores does not necessarily mean it is more preferred by humans. For example, ILP is preferred more on all three review data sets. Regarding the inter-annotator agreement, we find 48.5% of the individual judgements agree with the majority votes. The agreement scores decomposed by data sets and system pairs are shown in Table 4.5. Overall, the agreement scores are pretty low, compared to an agreement score achieved by randomly clicking (45.7%)[7]. It has several possibilities. The first one is that many turkers did click randomly (39 out of 160 failed our quality checkpoints). Unfortunately, we did not check all the turkers as we inserted the checkpoints randomly. The second possibility is that comparing two system summaries is difficult for humans, and

[5] The sum of the percentage is not 100% because there are "no preference" choices.

[6] For the significance test, we convert a preference to a score ranging from -2 to 2 ('2' means 'Strongly preferred' to a system and '-2' means 'Strongly preferred' to the counterpart system), and use a two-tailed paired t-test with $p < 0.05$ to compare the scores. Similar significant results can be observed if using a 3-point Likert scale ('preferred A', 'no preference', 'preferred B'), except that the difference between ILP and ILP+MC is not significant for Stat2015, but significant for CS2016 and movie.

[7] The random agreement score on a 5-Likert scale can be verified by a simulation experiment.

thus it has a low agreement score. Xiong and Litman (2014) also found that it is hard to make humans agree on the choice of summary sentences. A third possibility is that turkers needed to see the raw input sentences which are not shown in a HIT.

	ILP+MC vs. ILP	ILP+MC vs. SumBasic	SumBasic vs. ILP
Eng	51.1%	49.4%	50.9%
Stat2015	49.9%* (ILP+MC)	50.0%	51.2%
Stat2016	48.0%	49.2%	51.2%
CS2016	51.3%* (ILP+MC)	51.5%	50.6%* (SumBasic)
camera	49.2%	47.5%* (ILP+MC)	46.7%* (ILP)
movie	45.3%	50.7%* (SumBasic)	44.0%* (SumBasic)
peer	53.3%	43.3%	50.0%* (ILP)
DUC04	48.4%* (ILP)	46.4%* (ILP+MC)	44.0%

Table 4.5: Inter-annotator agreement measured by the percentage of individual judgements agreeing with the majority votes. * means the human preference to the two systems are significantly different and the system in parenthesis is the winner. Underline means that it is lower than random choices (45.7%).

An interesting pattern we found regarding the length of output summaries is that our approach produces longer summaries in terms of number of sentences, as shown in Table 4.6, although the length in terms of number of words is approximately the same for all methods for a particular corpus. Note that, for camera, movie and peer reviews, the human summary length is 10 sentences, and SumBasic and ILP+MC produce more sentences than ILP. It is hard for people to judge which system summaries is closer to a human summary when the summaries are long (216, 242, and 190 words for camera, movie, and peer reviews respectively). Examples are shown in Appendix D. For inter-annotator agreement, 50.3% of judgements agree with the majority votes for student response data sets, 47.6% for reviews, and only 46.3% for news documents. We hypothesize that for these long summaries, people may prefer short system summaries, and for short summaries, people may prefer long system summaries. We leave the examination of this finding to future work.

	Eng	Stat2015	Stat2016	CS2016	camera	movie	peer	DUC04
MEAD	1.6*	1.3*	2.2*	1.1*	3.0*	1.7*	3.3*	2.5*
LexRank	2.8*	2.4*	3.0*	1.9*	7.0	5.3*	6.0*	3.4*
SumBasic	6.0	5.6	5.8*	4.2	14.7	19.7*	12.3*	7.7
ILP	4.8*	3.6*	3.7*	2.6*	14.0*	17.7	12.0*	5.2*
ILP+MC	6.4	5.6	5.3	4.3	17.3	31.3	16.7	7.5

Table 4.6: Summarization output length measured by number of sentences. * means it is significantly different to ILP+MC ($p < 0.05$) using a two-tailed paired t-test.

Table 4.7 presents example system outputs. This offers intuitive understanding to our proposed approach.

4.4 ANALYSIS OF INFLUENTIAL FACTORS

In this section, we want to investigate the impact of the low-rank approximation process to the ILP framework. Therefore, in the following experiments, we focus on the direct comparison with the ILP and ILP+MC and leave the comparison to other baselines as future work. The proposed method achieved better summarization performance on Eng, CS2016, movie, and peer than the ILP baseline. Unfortunately, it doses not work as expected on two courses for student responses (Stat2015 and Stat2016), review camera and news documents. This leaves the research question when and why the proposed method works better. In order to investigate what are key factors that impact the performance, we would like to perform additional experiments using synthesized data sets.

A variety of attributes that might impact the performance are summarized in Table 4.8, categorized into two types. The **input** attributes are extracted from the input original documents and the **summaries** attributes are extracted from human summaries and the input documents as well. Here are some important attributes we expect to have a big

Prompt

Describe what you found most interesting in today's class

Reference Summary

- unit cell direction drawing and indexing
- real world examples
- importance of cell direction on materials properties

System Summary (ILP Baseline)

- drawing and indexing unit cell direction
- it was interesting to understand how to find apf and fd from last weeks class
- south pole explorers died due to properties of tin

System Summary (ILP+MC)

- crystal structure directions
- surprisingly i found nothing interesting today .
- unit cell indexing
- vectors in unit cells
- unit cell drawing and indexing
- the importance of cell direction on material properties

Table 4.7: Example reference and system summaries.

impact on the performance.

- $M * N$ is the size of the summarization task, represented by the size of the co-occurrence matrix A, as shown in Eq. 4.2 and Eq. 4.3. Generally, the bigger the matrix, the more difficult it is to find an optimal solution of low-rank approximation as there are more parameters. Note, A is an $N \times M$ matrix, where N is the number of unique concepts, and M is the number of sentences.
- For the sparsity ratio s, if the matrix is too sparse, there will not be enough information within A to have a good estimate of the completed matrix after imputation. In contrast,

	id	description
Input	1	• **genre**: belonging to student response/review/news
	2	• **T**: number of tasks
	3	• **au**: number of authors
	4	• **M*N**: size of A
	5	• **M**: number of sentences in total
	6	• **N**: number of bigrams in total
	7	• **M/T**: number of sentences per task
	8	• **N/T**: number of bigrams per task
	9	• **N/M**: number of bigrams per sentence
	10	• **W/T**: number of words per task
	11	• **W/M**: number of words per sentence
	12	• **s**: sparsity ratio, ratio of 0 cells in A per task
	13	• **b=1**: ratio of bigrams appear only once
	14	• **b>1**: ratio of bigrams appear more than once
	15	• H: Shannon's diversity index, defined as $H = -\sum_i p_i \ln p_i$, where p_i is the frequency of bigram i divided by total number of bigrams in a task
Summaries	16	• **L**: length of human summaries in number of words
	17	• **hs**: number of human summaries per task
	18	• **r**: compression ratio, length of human summaries compared to length of input documents
	19	• $\alpha_{b>0}$: abstraction ratio, how many of bigrams in **human summaries** appeared in the original documents at least once
	20	• $\alpha_{b=0}$: ratio of bigrams in human summaries that are not in the input
	21	• $\alpha_{b=1}$: ratio of bigrams in human summaries that are in the input only once
	22	• $\alpha_{b>1}$: ratio of bigrams in human summaries that are in the input more than once
	23	• $\beta_{b=1}$: ratio of bigrams in the **input** appear only **once** but selected by human(s)
	24	• $\beta_{b=2}$: ratio of bigrams in the input appear **twice** and selected by human(s)
	25	• $\beta_{b=3}$: ratio of bigrams in the input appear **three times** and selected by human(s)
	26	• $\beta_{b=4}$: ratio of bigrams in the input appear **four times** and selected by human(s)
	27	• $\beta_{b>1}$: ratio of bigrams in the input appear **more than once** and selected by human(s)

Table 4.8: Attributes description, extracted from the input and the human reference summaries.

id	name	Eng	Stat2015	Stat2016	CS2016	camera	IMDB	peer	DUC04
1	genre	response	response	response	response	review	review	review	news
2	T	36	44	48	46	3	3	3	50
3	au	37.7	39.3	42.2	22.4	18.0	18	18	10
4	M*N	13.8	10.8	7.2	7.4	0.9	15.7	0.7	2291.7
5	M	1492	1696	1660	1162	255	985	241	11566
6	N	9239	6366	4329	6409	3716	15934	2934	198140
7	M/T	41.4	38.5	34.6	25.3	85.0	328.3	80.3	231.3
8	N/T	256.6	144.7	90.2	139.3	1238.7	5311.3	978.0	3962.8
9	N/M	6.2	3.8	2.6	5.5	14.6	16.2	12.2	17.1
10	W/T	375.4	233.1	149.3	223.1	1927.0	8014.0	1543.7	5171.6
11	W/M	9.1	6.0	4.3	8.8	22.7	24.4	19.2	22.4
12	s	97.2%	96.6%	96.0%	95.4%	98.5%	99.6%	98.5%	99.4%
13	$b = 1$	90.3%	90.1%	87.6%	94.0%	94.7%	92.6%	91.1%	85.5%
14	$b > 1$	9.7%	9.9%	12.4%	6.0%	5.3%	7.4%	8.9%	14.5%
15	H	5.282	4.590	4.007	4.703	6.894	8.314	6.617	7.844
16	L	30	15	13	16	216	242	190	105
17	hs	1	2	2	2	8	5	2	4
18	r	0.088	0.076	0.109	0.083	0.131	0.031	0.135	0.024
19	$\alpha_{b>0}$	48.8%	46.5%	56.4%	45.8%	96.7%	97.6%	95.9%	37.0%
20	$\alpha_{b=0}$	51.2%	53.5%	43.6%	54.2%	3.3%	2.4%	4.1%	63.0%
21	$\alpha_{b=1}$	34.1%	18.1%	20.9%	25.6%	84.9%	76.4%	77.1%	15.9%
22	$\alpha_{b>1}$	14.7%	28.4%	35.5%	20.2%	11.8%	21.2%	18.8%	21.1%
23	$\beta_{b=1}$	3.3%	2.7%	4.3%	3.7%	45.8%	11.2%	23.3%	1.7%
24	$\beta_{b=2}$	8.5%	16.5%	28.2%	25.1%	65.3%	20.4%	40.3%	7.2%
25	$\beta_{b=3}$	12.5%	39.0%	58.8%	57.4%	79.3%	31.8%	53.8%	13.7%
26	$\beta_{b=4}$	33.3%	61.1%	76.9%	50.0%	90.9%	42.9%	50.0%	22.1%
27	$\beta_{b>1}$	12.3%	28.0%	45.2%	37.0%	70.0%	27.7%	46.0%	12.0%

Table 4.9: Attributes extracted from the input and the human reference summaries. The numbers in the row of $M * N$ are divided by 10^6. The description of each attribute is shown in Table 4.8.

if the matrix is not sparse at all (e.g., all authors use the same term for a concept), there will be no benefit to performing low-rank approximation.

- The Shannon's diversity index H measures the degree of bigram diversity. The more diverse the bigram distribution, the smaller the corresponding Shannon entropy.
- The abstraction ratios $\alpha_{b=0}$, $\alpha_{b=1}$, $\alpha_{b>1}$ capture in what degree annotators use words from the input or use their own.
- $\beta_{b=1}$, $\beta_{b=2}$, $\beta_{b=3}$, $\beta_{b=4}$, $\beta_{b>1}$ intend to capture how humans create the summaries in terms of whether more frequent bigrams are more likely to be selected by humans.

The attributes extracted from the corpora are shown in Table 4.9. Note, a bigram that appears more often in original documents has a better chance to be included in human summaries as indicated by $\beta_{b=1}$, $\beta_{b=2}$, $\beta_{b=3}$, and $\beta_{b=4}$. This verifies our choice to cut low-frequency bigrams.

According to the ROUGE scores, our method works better on Eng, CS2016, movie, and peer (Table 4.4). If we group each attribute into two groups, corresponding to whether ILP+MC works better, we do not find significant differences among these attributes. To further understand which factors impact the performance and have more predictive power, we train a binary classification decision tree by treating the 4 working corpora as positive examples and the remaining 4 as negative examples.

According to the decision tree model, there is only one decision point in the tree: $\alpha_{b=1}$, the ratio of bigrams in human summaries that are in the input only once. Generally, our proposed method works if $\alpha_{b=1} > 23.2\%$, except for camera. When $\alpha_{b=1}$ is low, it means that annotators either adopt concepts that appear multiple times or just use their own. In this case, the frequency-based weighting (i.e., w_i in Eq. 4.1) can capture the concepts that appear multiple times. On the other hand, when $\alpha_{b=1}$ is high, it means that a big number of bigrams appeared only once in the input document. In this case, annotators have difficulty selecting a representative one due to the ambiguous choice. Therefore, we hypothesize,

- **H1.4**: The ILP framework benefits more from low-rank approximation when $\alpha_{b=1}$ is higher.

To test the predictive power of this attribute, we want to test it on new data sets.

Unfortunately, creating new data sets with gold-standard human summaries is expensive and time-consuming, and the new data set may not have the desired property within a certain range of $\alpha_{b=1}$. Therefore, we propose to manipulate the ratio and create new data sets using the existing data sets without additional human annotation. $\alpha_{b=1}$ can be represented as follows,

$$\alpha_{b=1} = \frac{\sum_i \sigma_i \cdot \phi_{w_i=1}}{\sum_i \sigma_i}$$

where

$$\sigma_i = \begin{cases} 1 & \text{if bigram } i \text{ appears in the human summary} \\ 0 & else \end{cases}$$

$$\phi_{w_i=1} = \begin{cases} 1 & \text{if } w_i = 1, w_i \text{ is the weight of the bigram i} \\ 0 & else \end{cases}$$

There are two different ways to control the ratio, both involving removing input sentences with certain constraints.

- To increase this ratio, we remove sentences with bigrams that appear multiple times so that there will be more bigrams that appear once (i.e., increase $\sigma_i \cdot \phi_{w_i=1}$) and thus increase the numerator. For example, if a bigram in a human summary appears in two input sentences (e.g., S1 and S2), we can randomly remove one of them (either S1 or S2) to make the bigram appear only once in the input. Note that we keep sentences that have bigrams appearing multiple times and a bigram appearing only once as well, so that we guarantee that all the input sentences with a unique bigram in human summaries are kept and removing other sentences can only increase the ratio.
- To decrease this ratio, we remove the sentences with bigrams that appear only once in order to decrease the numerator. This will reduce the bigram frequency w_i from 1 to 0. Similarly, we keep sentences that contain bigrams appearing multiple times so that removing sentences will not increase the ratio.

In this way, we obtained different levels of $\alpha_{b=1}$ by deleting sentences. The ROUGE scores on the synthesized corpus are shown in Table 4.10.

			R-1			**R-2**		
	$\alpha_{b=1}$	**System**	R	P	F	R	P	F
Eng	26.5	ILP	.341	.318	.295	.112	.114	.102
		ILP+MC	**.378**$^+$	**.324**	**.311**	.112	.114	.100
	34.1	ILP	.364	.329	.308	.123	.124	.110
		ILP+MC	**.392**$^+$	**.335**	**.322**	**.130**	**.127**	**.114**
	36.0	ILP	.358	.327	.306	.119	.120	.107
		ILP+MC	**.397**$^+$	**.339**	**.327**	**.126**	**.123**	**.111**
Stat2015	11.9	ILP	.401	.395	.387	.183	.173	.172
		ILP+MC	.362$^-$	.340$^-$	.341$^-$	.161	.149$^-$	.149$^-$
	18.1	ILP	.405	.396	.390	.186	.175	.174
		ILP+MC	.401	.372	.375	.183	.164	.167
	21.0	ILP	.394	.391	.382	.172	.161	.160
		ILP+MC	.372	.352$^-$	.352	.156	.147	.147
Stat2016	13.2	ILP	.467	.500	.480	.252	.271	.259
		ILP+MC	.463	.486	.471	.212$^-$	.222$^-$	.215$^-$
	20.9	ILP	.482	.516	.496	.262	.283	.270
		ILP+MC	.457	.489	.465	.214$^-$	.230$^-$	.218$^-$
	23.7	ILP	.455	.488	.468	.244	.265	.252
		ILP+MC	**.462**	.480	.467	.213	.220$^-$	.214$^-$
CS2016	11.0	ILP	.362	.395	.369	.138	.150	.140
		ILP+MC	**.386**	.395	**.382**	.138	.140	.135
	25.6	ILP	.374	.408	.382	.141	.155	.144
		ILP+MC	**.398**	**.409**	**.395**	**.154**	**.156**	**.151**
	34.2	ILP	.296	.330	.306	.091	.108	.097
		ILP+MC	**.335**$^+$	**.347**	**.334**	**.102**	.106	**.102**
camera	78.7	ILP	.453	.460	.456	.166	.166	.166
		ILP+MC	.418	.430	.423	.137	.142	.139
	84.9	ILP	.457	.466	.460	.165	.165	.165
		ILP+MC	.447	.449	.447	.157	.158	.157
	85.8	ILP	.452	.465	.457	.156	.159	.158
		ILP+MC	**.458**	**.469**	**.462**	**.166**	**.170**	**.168**
movie	71.9	ILP	.439	.430	.432	.116	.111	.113
		ILP+MC	.423	.417	.417	.101$^-$	.098$^-$	.099$^-$
	76.4	ILP	.435	.424	.427	.091	.087	.088
		ILP+MC	**.436**	**.427**	**.429**	**.106**$^+$	**.100**$^+$	**.102**$^+$
	76.8	ILP	.435	.427	.428	.109	.105	.106
		ILP+MC	.408	.402	.402	.100	.097	.098
peer	71.3	ILP	.467	.465	.465	.206	.201	.203
		ILP+MC	.431	.447	.439	.163	.170	.166
	77.1	ILP	.466	.469	.466	.199	.196	.197
		ILP+MC	**.491**	**.496**	**.492**	**.261**	**.262**	**.260**
	78.7	ILP	.488	.479	.482	.242	.229	.234
		ILP+MC	.456	.466	.460	.204	.207	.205
DUC04	13.9	ILP	.376	.380	.378	.092	.093	.092
		ILP+MC	.349$^-$	.350$^-$	.349$^-$	.074$^-$	.074$^-$	.074$^-$
	15.9	ILP	.377	.381	.379	.092	.093	.092
		ILP+MC	.342$^-$	.351$^-$	.346$^-$	.072$^-$	.074$^-$	.072$^-$
	16.5	ILP	.375	.379	.377	.093	.094	.094
		ILP+MC	.349$^-$	.351$^-$	.349$^-$	.074$^-$	.075$^-$	.074$^-$

Table 4.10: ROUGE scores on synthesized corpora. **Bold** scores indicate our approach ILP+MC is better than ILP. $^+$ and $^-$ mean a score is significantly better and worse respectively ($p < 0.05$) using a two-tailed paired t-test.

Our hypothesis **H1.4** is partially valid. When increasing the ratio, ILP+MC has a relative advantage gain over ILP. For example, for Stat2015, ILP+MC is not significantly worse than ILP any more when increasing the ratio from 11.9 to 18.1. For camera, ILP+MC becomes better than ILP when increasing the ratio from 84.9 to 85.8. For Stat2016, CS2016, Eng, more improvements or significant improvements can be found for ILP+MC compared to ILP when increasing the ratio. However, for movie and peer review, ILP+MC is worse than ILP when increasing the ratio.

We have investigated a number of attributes that might impact the performance of our proposed method. Unfortunately, we do not have a conclusive answer when our method will work better. However, we would like to share some thoughts about it.

First, our proposed method works better on two student responses courses (Eng and CS2016), but not the other two (Stat2015 and Stat2016). An important factor we ignored is that the students from the other two courses are not native English speakers, resulting in significantly shorter responses ($4.3 < 6.0 < 8.8, 9.1$, $p < 0.01$, Table 4.9, the row with id=11). With shorter sentences, there will be less context to leverage the low-rank approximation.

Second, our proposed method works better on movie and peer reviews, but not camera reviews. As pointed out by Xiong (2015), both movie reviews and peer reviews are potentially more complicated than the camera reviews, as the review content consists of both the reviewer's evaluations of the subject (e.g., a movie or paper) and the reviewer's references of the subject, where the subject itself is full of content (e.g., movie plot, papers). In contrast, such references in product reviews are usually the mentions of product components or properties, which have limited variations. This characteristic makes review summarization more challenging in these two domains.

4.5 SUMMARY

We made the first effort to summarize student feedback using an Integer Linear Programming framework with a low-rank matrix approximation, and applied it to different types of data sets including news articles, product and peer reviews. Our approach allows sentences to share co-occurrence statistics and alleviates sparsity issue. Our experiments showed that the proposed approach performs better against a range of baselines on the student response Eng and CS2016 on ROUGE scores, but not other courses. Therefore, H1.1 is partially confirmed and further investigation is needed. For H1.2, the extended-ILP framework does not deliver better summarization performance than the traditional ILP-based framework on all student responses in terms of ROUGE scores. For H1.3, the extended-ILP framework cannot be directly applicable to news and camera review.

We also investigated a variety of attributes that might impact the performance on a range of data sets. Unfortunately, we did not have a conclusive answer when our method will work better.

5.0 PHRASE SUMMARIZATION

This chapter introduces a novel summarization method at a phrase level (Luo and Litman, 2015). It assumes that the concepts (represented as phrases) mentioned by more students should get more attention from the instructor. Based on this assumption, we introduce the notion of *student coverage*, defined as the number of students who semantically mention a particular phrase (i.e., student supporters). The more student coverage a phrase has, the more important it is.

It differs from traditional methods in two primary ways. First, it is an extractive summarization technique at the scale of phrases, in which summaries are created from extracted phrases rather than from sentences. Phrases are easy to read and browse like keywords, and fit better on small devices when compared to sentences. Long sentences are decomposed into different short phrases, which will be treated the same as phrases from short sentences. In addition, only noun phrases are extracted and thus phrases such as “to be most interesting” and “was interesting” are filtered out, addressing the length variety and redundancy challenge. Second, we adopt a metric clustering paradigm (k-medoids) with the latent semantic analysis similarity to group extracted phrases, allowing similar phrases to be grouped together even if they are in different textual forms, addressing the lexical variety and quantity challenges.

5.1 PROPOSED METHOD

We formulate our task as a standard extractive summarization problem. Unlike standard sentence-level extraction where the input and output are sentences, the input of our task ranges from words or phrases to full sentences. The output is a list of important phrases and the summary length (either # of phrases or words) is no more than L.

The proposed algorithm involves three stages: *candidate phrase extraction*, *phrase clustering*, and *phrase ranking*.

5.1.1 Candidate phrase extraction

We extract noun phrases (NPs) from the input using a syntax parser from the Senna toolkit (Collobert, 2011), preserving the most important content from the original responses without losing too much context information compared to keywords. For example, "The main topics of this course" (S12 in Table 1.2) is extracted as a candidate phrase. Only NPs are considered because all reflection prompts used in the task are asking about "what", and knowledge concepts are usually represented as NPs.

Due to the noisy data, malformed phrases are excluded, including single stop words (e.g. "it", "I", "there", "nothing") and phrases starting with a punctuation mark (e.g. "'t", "+ indexing").

5.1.2 Phrase clustering

Phrases are more meaningful and less ambiguous compared to keywords given the fact that it is difficult for a user to figure out the actual meanings when given only a list of keywords, as it loses the order of words. For example, what does it mean by listing the keywords: 'to', 'related', 'freedom', 'degrees', 'concepts', and 'of'?[1] However, phrases suffer from the sparsity problem as they are longer, especially in our data set when 89.9% of the phrases appeared only once. The challenge is the fact that students use different words for the same meaning (e.g., "bicycle parts" and "bike elements").

We use a clustering paradigm with a semantic distance metric to address this issue. Among different clustering algorithms, k-medoids (Kaufman and Rousseeuw, 1987) fits well for our problem. First, it works with an arbitrary distance matrix between datapoints, allowing pairwise semantic similarity-based distance between phrases to be used, yielding

[1] It means "concepts related to degrees of freedom".

metric clustering. Second, it is robust to noise and outliers because it minimizes a sum of pairwise dissimilarities instead of squared Euclidean distances. It shows better performance than an LDA-based approach to group students' short answers for the purpose of semi-automated grading (Basu et al., 2013). Since k-medoids picks a random set of seeds to initialize as the cluster centers (called medoids), the clustering algorithm runs 100 times and the cluster with the minimal within-cluster sum of the distances is retained to reduce random effects.

Distance metric. The semantic similarity is implemented using SEMILAR (Rus et al., 2013), using the *latent semantic analysis* trained on the Touchstone Applied Science Associates corpus (Ştefănescu et al., 2014). The distance matrix D is constructed from the similarity matrix S by applying the following transformation: $D = e^{-S}$, which is similar to the common heat kernel but without normalization[2].

Number of clusters. For setting the number of clusters without tuning, we adopted a method from Wan and Yang (2008), by letting $K = \sqrt{V}$. where K is the number of clusters and V is the number of candidate phrases instead of the number of sentences.

5.1.3 Phrase ranking

In order to estimate the *student coverage*, phrases are clustered with the algorithm introduced above. We assume the phrases in a cluster are semantically similar to each other and any phrase in a cluster can represent it as a whole. Therefore the coverage of a phrase is assumed to be the same as the coverage of a cluster, which is a union of the students covered by each phrase in the cluster.

We explore two ways to select the most representative phrase in a cluster. The first way to score the extracted candidate phrases is by using LexRank (Erkan and Radev, 2004), a graph-based algorithm for computing relative importance of textual units (working for both sentences and phrases). The top-ranked phrase in the cluster is added to the output summary. This process starts from the cluster that has the most estimated student coverage and repeats for the next cluster until the length limit is reached. The second method is to

[2] This is not normalized to the range between 0 and 1 since we only care about the relative distance.

select the medoid phrase instead of using LexRank to rank the phrases in a cluster to form the summary.

Note that when the student coverage is the same between two clusters, the score of the top-ranked phrases in the clusters according to LexRank is used to break the tie: the higher, the better.

5.2 EXPERIMENTS

We use the ROUGE evaluation metric (Lin, 2004) and report R-1 (unigrams) and R-2 (bigrams), including the recall (R), precision (P) and F-Measure (F). These scores measure the overlap between human-generated summaries and a machine-generated summary. We also perform human evaluation, as we did in §4.3.2.

We design and compare a number of other summarization methods[3] to evaluate the proposed phrase summarization approach.

Keyphrase extraction. Maui (Medelyan et al., 2009) is selected as the baseline, which is one of the state-of-the-art keyphrase extraction methods.

Sentence to phrase summarization. Existing sentence summarization techniques can be used for phrase summarization by extracting candidate phrases and treating them as sentences. Within this framework, we adapt MEAD (Radev et al., 2004) and LexRank (Erkan and Radev, 2004) to our task. We also include the original MEAD[4] (summarizing at a sentence level) for comparison (named as OriMEAD).

Diversity-based summarization. We applied the MMR (Carbonell and Goldstein, 1998), a popular diversity-based summarization method as a post-processing step to the MEAD (**MEAD+MMR**) and LexRank (**LexRank+MMR**) baselines.[5]

[3] For MEAD and LexRank, the results in this chapter are different from the results in §4.3.2 since the summary output length limit is number of phrases here instead of number of words.

[4] The default Length parameter in MEAD is changed to 1 from its default value 9 and the position feature is removed, yielding better performance.

[5] For each MMR based baseline, the parameter is optimized with a grid search on the development data set.

	R-1			R-2			Human
	R	P	F	R	P	F	Preference
Keyphrase	.171	.364	.211	.057	.134	.071	-
OriMEAD	**.397**	.185	.219	.117	.069	.073	-
MEAD	.341	.269	.265	.122	.102	.099	-
MEAD+MMR	.360	.279	.277	**.130**	.106	.104	**30.8%**
LexRank	.325	.355	.307	.107	.110	.102	-
LexRank+MMR	.328	.367	.312	.111	.126	.110	19.7%
Cluster+Medoid	.279	**.473***†	.327	.078	.129	.091	18.6%
Cluster+LexRank	.319	.448*†	**.340**†	.122	**.176***†	**.134**	20.5%

Table 5.1: Summarization performance on student response (Eng). The last two rows are our proposed approaches. Cluster+LexRank uses LexRank to score phrases in the last step. Cluster+Medoid selects the medoid phrase instead of using LexRank to rank the phrases in a cluster to form the summary. The highest score for each column is shown in **bold**. † indicates that the improvement over the MEAD+MMR baseline is statistically significant. * indicates that the improvement over LexRank+MMR is statistically significant.

5.2.1 Results

For student response (Eng) (§3.1), 4 lectures are randomly selected as a development set and the remaining data used as a test set, yielding 12 sets of development data and 24 sets of testing data, each with a prompt, the students' responses and the gold-standard summary.

The performance on the test set is shown in Table 5.1 with the length limit L as 4 phrases (the average phrase number in the human summaries, Table 3.1). For the human evaluation, we select 4 competitive system-system pairs: Cluster+Medoid vs. MEAD+MMR, Cluster+Medoid vs. LexRank+MMR, Cluster+LexRank vs. MEAD+MMR, and Cluster+LexRank vs. LexRank+MMR.

First, our proposed method Cluster+LexRank, which clusters the extracted phrases and uses LexRank to score them, can outperform all the baselines over both R-1 and R-2 in terms of F-measure. In addition, the proposed model performs better than the clustering and LexRank alone. Through a paired t-test, our model outperforms LexRank statistically in terms of precision for both ROUGE scores and significantly improves Cluster+Medoid on all R-2 scores (except the precision with 0.06 p-value). We believe that the semantic similarity based clustering complements LexRank in two ways: 1) LexRank depends on the cosine similarity of TF-IDF vectors to build the graph while the clustering takes semantic similarity into account. 2) The clustering performed a global selection to form a summary by grouping similar phrases and ranking them by the number of covered students (similar to what the human did). Compared to LexRank, our approach captures the student coverage explicitly. While modifying LexRank by using semantic similarity is possible, estimating the student coverage is not straightforward.

Second, OriMEAD tends to select long sentences, resulting in a high recall but a low precision. The phrase version (MEAD) improves both the P and F scores by removing unnecessary parts in the original sentences.

Thirdly, Cluster+LexRank outperforms the MMR based baselines on the precision and F-measure of all two ROUGE scores. We observed that the MMR baselines suffer from the issue of diverse expressions used by the students (e.g., "graphs" and "charts").

Lastly, turkers prefer MEAD+MMR over our proposed methods. As shown in Table 5.1, MEAD+MMR does have a higher recall of ROUGE than other methods since it tends to select long phrases.

We also evaluate the proposed method on the student response Stat2015, Stat2016, and CS2016 (§3.2), using the same experimental setting as student response (Eng), except setting the summary length limit as 5 phrases (the same length limit given to human annotators). Note, there are two important improvements. First, there are more lectures. Second, the summaries are double-coded, taking the content selection variation in human summaries into account and thus making ROUGE evaluation more reliable (Van Halteren and Teufel, 2003; Teufel and van Halteren, 2004). When multiple human summaries are available, ROUGE scores are computed by averaging the scores using different human reference summaries with a Jackknifing procedure (Lin, 2004).

The results are summarized in Table 5.2, using the phrase summaries as the gold-standard summaries.

		R-1			R-2			Human
		R	P	F	R	P	F	Pre.
Stat2015	Keyphrase	.264	.297	.273	.083	.094	.087	-
	OriMEAD	.475	.138	.209	.173	.048	.073	-
	MEAD	**.485**	.228	.300	.208	.090	.121	-
	MEAD+MMR	.417	.205	.266	.158	.074	.098	24.0%
	LexRank	.475	.292	.349	.210	.123	.149	-
	LexRank+MMR	.436	.293	.340	.195	.121	.143	18.4%
	Cluster+Medoid	**.485***†	**.427***†	**.441***†	**.211**	**.177***†	**.187***†	24.0%*
	Cluster+LexRank	.480*†	.336*†	.382*†	.199	.129†	.152†	**26.0%***†
Stat2016	Keyphrase	.352	.517	.411	.159	.232	.185	-
	OriMEAD	.556	.256	.341	.260	.117	.157	-
	MEAD	**.584**	.356	.436	**.314**	.185	.229	-
	MEAD+MMR	.459	.290	.349	.236	.147	.177	19.2%
	LexRank	.553	.397	.455	.284	.198	.230	-
	LexRank+MMR	.494	.383	.423	.249	.188	.210	20.3%
	Cluster+Medoid	.534*	**.576***†	**.544***†	.264	**.285***†	**.268***†	**27.8%***†
	Cluster+LexRank	.546*†	.485*†	.501*†	.274	.246*†	.251*†	27.2%*†
CS2016	Keyphrase	.241	**.516**	.319	.072	**.160**	.096	-
	OriMEAD	**.535**	.124	.197	.169	.038	.061	-
	MEAD	.494	.236	.313	.167	.077	.103	-
	MEAD+MMR	.491	.225	.301	**.172**	.078	.104	**28.4%**
	LexRank	.428	.250	.305	.126	.072	.088	-
	LexRank+MMR	.430	.284	.332	.136	.089	.103	24.4%
	Cluster+Medoid	.352	.357*†	**.337**	.111	.116*†	**.107**	20.1%
	Cluster+LexRank	.398	.291	.319	.122	.090	.097	21.0%

Table 5.2: Summarization performance on student response Stat2015, Stat2016, and CS2016. The highest score for each column is shown in **bold**. † indicates that the improvement over the MEAD+MMR baseline is statistically significant. * indicates that the improvement over LexRank+MMR is statistically significant.

In general, most observations based on student response (Eng) still hold. The phrase-level MEAD is better than sentence-level (OriMEAD) for all ROUGE scores on P and F. Our proposed phrase summarization algorithms outperform the MEAD and LexRank baselines over ROUGE scores in terms of precision. Surprisingly, **Cluster+Medoid** achieves best F, different from Eng. Here are three possible reasons. First, the student responses for the Eng courses are more diverse than the other three courses (the Shannon's diversity index H of

Eng is significantly bigger than the other three courses, $p < 0.05$, as shown in Table 4.9), so that the clustering algorithm may not work as well. Second, there is only one gold standard for Eng, making ROUGE evaluation less reliable. Third, the student responses from Eng were collected on papers by writing while the other three are collected online by typing. Interestingly, our methods now not only have better ROUGE scores for Stat2015 and Stat2016, but also they are preferred more by humans.

5.2.2 Clustering output

To delve into how the clustering helps summarization, the clustering results are shown in Table 5.3 for the example in Appendix A. There are 66 candidate phrases extracted from the student feedback, grouping into 8 clusters. Generally speaking, the output of the clustering is reasonable. The first cluster is a little noisy, but all the other clusters have a good quality. For example, "the graphs", "size of print and colors" and "the different graphs" are grouped into the 3^{rd} cluster; all phrases related to "bond strength" and "thermal expansion" are in the 2^{nd} one.

Note duplicate phrases are not removed in the clustering because they are from different students. At the same time, the majority of phrases appear just once. That's why term-frequency based methods do not perform well on this task.

5.2.3 Summary output

To demonstrate the effectiveness of our proposed method, Table 5.4 shows the summaries generated from different methods for the example in Appendix A.

First, the keyphrase extraction does not work well. One reason is because this is a supervised method training on a different corpus. However, another important reason is that it relies on term frequency and ignores the semantic similarity. Moreover, it classifies each candidate phrase whether as a keyphrase independently, without considering the redundancy.

Second, LexRank does not capture the phrase "graphs of attraction/repulsive" even though it is the top-ranked phrase in the TA's summary. One reason is because the students used many different expressions (e.g., "graphs" and "charts").

cluster	student #	phrases
1	17	**the class**, this class, most of the lecture, the lecture, metal, each metal, a laser pointer,15 % of the class, *part iii on worksheet in class*, comparing metals ., the answers to part iii, hooke 's law, all information, a group member, a much faster rate, the values, the text, resilience, that calculation, the pictures, specific detail, any trouble with anything, the projector, printout
2	10	**equations with bond strength and hooke 's law**, *the coefficient of thermal expansion relationship to bond strength*, a little confusing properties related to bond strength, 4 : axes on coefficient of thermal expansion graph ., higher coefficient of thermal expansion, property related to bond strength, the bond strength, the coeff of thermal expansion, equations with stress, the concept of thermal expansion
3	8	**graphs and equations**, the graphs, graphs, the different graphs that look the same, several slides with complicated graphs and undefined variables, *the graph*, energy vs. distance between atoms graph and what it tells us, size of print and colors
4	6	***graphs of attractive + repulsive forces***, graphs of attraction / repulsive& interatomic separation, the graphs of attraction and repulsion, the attractive and repulsive force graphs from the third slide, the repulsive / attraction charts, stress + strain, atomic structure
5	5	**the activity (part iii)**, *the activity*, the activity, the activity, more than activities
6	5	***elastic modulus***, elastic modulus, elastic modulus, the elastic modulus, the working definition of elasticity
7	5	***not the least bit confusing***, nothing confusing, a little bit, the white board, van der waals
8	3	**the terms and equations**, the trends, *the concepts*

Table 5.3: Clustering output for the example in Appendix A. Phrases within a cluster are ranked in order by LexRank scores. **Bold** one is top ranked. *Italics* one is the medoid.

Third, the MMR does not change the LexRank output. All 4 phrases given by LexRank have high scores. Hence, even with a redundancy penalty, they still do not get removed even though they are redundant with each other (e.g. "bond strength").

Lastly, by clustering, phrases in the summary are from different clusters and the phrases are ranked by the number of covered students, which makes the phrase "graphs of attractive + repulsive forces" rank higher than "the activity (part iii)". At the same time, we notice Clustering+LexRank introduces noise (e.g. "the class") with a large cluster. This phenomenon is also observed by Basu et al. (2013), called “collapse”.

5.3 SUMMARY

In this chapter, we presented a novel algorithm to summarize student feedback to reflection prompts by a combination of phrase extraction, phrase clustering, and phrase ranking. It makes use of metric clustering to rank the phrases by their student coverage, taking the information source into account. Experimental results demonstrate the good effectiveness of our models on student response data with respect to automatic evaluation via ROUGE and some human evaluation results. Therefore, H2 is supported.

Human Summary	1) Graphs of attraction/ repulsive & atomic separation [10] 2) Properties and equations with bond strength [7] 3) Coefficient of thermal expansion [6] 4) Activity part III [4]
Keyphrase	1) coefficient of thermal 2) elastic modulus 3) thermal expansion 4) thermal expansion graph
LexRank	1) graphs and equations [1.0] 2) equations with bond strength and Hooke 's law [0.97] 3) the coefficient of thermal expansion relationship to bond strength [0.95] 4) the activity (part iii) [0.91]
LexRank+MMR	1) graphs and equations 2) equations with bond strength and Hooke 's law 3) the coefficient of thermal expansion relationship to bond strength 4) the activity (part iii)
Clustering+Medoid	1) part iii on worksheet in class [17] 2) the coefficientof thermal expansion relationship to bond strength [10] 3) the graph [8] 4) graphs of attractive + repulsive forces [6]
Clustering+LexRank	1) the class [17] 2) equations with bond strength and hooke 's law [10] 3) graphs and equations [8] 4) graphs of attractive + repulsive forces [6]

Table 5.4: Summary outputs for the example in Appendix A. The numbers shown in square brackets for our models are computed by the method introduced in §5.1.3, which indicates the number of students for the corresponding phrase. The numbers for LexRank are the LexRank scores assigned to the phrases, ranging from 0 to 1.0.

6.0 QUANTITATIVE PHRASE SUMMARIZATION

In this chapter, we design new approaches to address the quantity challenge since the quantitative information is valuable to instructors. Results evaluating the proposed method on Stat2015 and Stat2016 were previously published (Luo et al., 2016a). In this chapter, we also evaluate our approach on a third course (CS2016) and add cross-course evaluation.

Recall that the proposed phrase summarization (Luo and Litman, 2015) (henceforth **L&L**) introduced in §5 consists of three stages: phrase extraction, phrase clustering, and phrase ranking. The approach extracts noun phrases from student responses, groups the phrases using a greedy clustering algorithm, and finally selects representative phrases from the clusters. Although the *phrase summarization framework* partially addresses the quantity challenge, it has four limitations.

First, noun phrases do not suffice. Other types of phrases such as "how confidence intervals linked with previous topics" are useful and should be allowed. Second, clustering is based on similarity, but the similarity of phrases that do not appear in a background corpus (i.e., the corpus used to learn the similarities) cannot be captured in the previous setting. Third, a greedy clustering algorithm k-medoids (Kaufman and Rousseeuw, 1987) was previously used to group candidate phrases. It ignores global information and may suffer from a "collapsing" effect, which leads to the generation of a large cluster with unrelated items (Basu et al., 2013). Last, there is no evaluation of the estimated student number at all. ROUGE measures how well a system summary overlaps human summaries, however, it is limited at least in two ways. 1) All sentences/phrases in the summary are assumed to be equally important during the evaluation. It is against our assumption that the concepts (represented as phrases) mentioned by more students should get more attention from the instructor and are thus more important. 2) ROUGE only considers the overlap of words that are exactly the same and ignores the lexical variety problem[1].

The goal of this work is to explore a phrase-based highlighting scheme, which is new to the summarization task. We aim to improve the phrase summarization framework by exploiting new capabilities that are enabled by the highlighting scheme. In the new scheme, human annotators are instructed to 1) create summary phrases from the student responses, 2) associate a number with each summary phrase which indicates the number of students who raise the issue (henceforth **student supporters**), and 3) highlight the corresponding phrases in both the human summary and student responses. Table 6.1 illustrates the highlighting scheme and more details are presented in §6.1. The new highlighting scheme makes it possible to develop a supervised candidate phrase extraction model (§6.2.1) and estimate pairwise phrase similarity with supervision (§6.2.2). To solve the third limitation, we explore a community detection algorithm OSLOM (Lancichinetti et al., 2011) that optimizes the statistical significance of clusters with respect to a global null model (§6.2.3). Experimental results show that the newly developed phrase extraction model is better than noun phrases only, in terms of both intrinsic and extrinsic measures; phrase similarity learning appears to produce marginal improvement; and the community detection approach yields better phrase summaries with more accurate estimation of the number of student supporters.

In summary, the contribution of this work is threefold.

- We introduce a new phrase-based highlighting scheme for automatic summarization, a departure from prior work. It highlights the phrases in the human summary and also the semantically similar phrases in student responses.
- We push the boundary of a phrase-based summarization framework by using our highlighting scheme to enable identification of candidate phrases as well as estimation of phrase similarities with supervision, and by using community detection to group phrases into clusters.
- We conduct comprehensive evaluations in terms of both summary text quality, measured by ROUGE (Lin, 2004) and human evaluation, and by how well phrase summaries capture the most pressing student needs, measured by a new evaluation metric based on

[1] ROUGE solves the lexical variety problem by using multiple annotators. However, even with multiple annotators, the human summaries do not list all semantically-equivalent expressions.

color matching.

Reflective Prompt Describe what was confusing or needed more detail.	
Student Responses S1: In the age of distributions example, application of qq plot ^{g} was confusing S2: Last problem about normalization ^{m} S3: central limit teorem ^{y} and A And B events example formulas were different. I did not understand that part well	**Human Summary 1** - central limit theorem ^{y} [12] - q-q plot ^{g} [9] - sampling distribution ^{r} [6] - normal approximation ^{b} [5] - normalization (last example) ^{m} [3]
S4: Sampling distribution ^{r} was a little bit abstract S5: Q-q plot ^{g} S6: Central Limit Thm ^{y} S7: CLT ^{y} S8: Normal approximation to binomial ^{b} S9: bernaulli random variables S10: The central limit ^{y} and normal approximations ^{b} ...	**Human Summary 2** - central limit theorem [13] - q-q plots [9] - general more explanations/details, better handwriting, move slower [9] - sampling distributions [6] - nothing [6]

Table 6.1: Example prompt, student responses, and two human summaries. 'S1'–'S10' are student IDs. The summary phrases are each tagged with the number of students who raise the issue (i.e., student supporters). The summary and phrase highlights are manually created by annotators. Phrases that bear the same color belong to the same issue. Each annotator is free to choose his/her color palette. We have only demonstrated the highlights of **Human Summary 1** to avoid overlaying of two sets of colors on student responses. The superscripts of the phrase highlights are imposed by the author to differentiate colors when printed in grayscale (y: yellow, g: green, r: red, b: blue, and m: magenta).

6.1 DATA SETS

We will use student response Stat2015, Stat2016 and CS2016 (§3.2) to develop as well as to evaluate the proposed quantitative phrase summarization because of the unique highlighting scheme of the three data sets.

We argue that the new highlighting scheme can provide many unique benefits. First, it allows us to track the "source phrases" that humans use to create the summary phrase. For example, the first summary phrase in Human Summary 1 of Table 6.1 is "central limit theorem." It is created from a collection of phrases in the student responses, including "The central limit", "central limit teorem" (a typo by the student), "CLT" (its abbreviation), and "Central Limit Thm" (another abbreviation). Naturally the highlighted source phrases lend themselves to a supervised approach to candidate phrase extraction. Second, the highlights inform us about the similarity and dissimilarity of phrases. For example, the source phrases that bear the same color are semantically similar to each other, whereas those with different colors are semantically dissimilar. In a similar vein, we develop a supervised approach that learns to predict the phrase similarity using highlights as guidance. Third, we are now able to accurately match the phrases in a system summary to those in a human summary, allowing the development of a novel summarization evaluation metric. For instance, assuming the system summary contains the phrase "Last problem about normalization" from S2 (Table 6.1), using the color highlights, we know that this phrase matches the human summary phrase "normalization (last example)." Such semantic matching between system and human summaries remains an elusive challenge for traditional summarization evaluation, but highlights make it an easy decision. Finally, the highlights on source texts indicate to what extent the information has been retained in the human summary. Specific to our task, we are interested to know the percentage of students whose responses are covered by the human summary. We define a student coverage score where a student is covered if and only if part of his/her response is highlighted. For example, in Table 6.1, S9 is considered not covered by Human Summary 1.

Basic statistics of the dataset are presented in Table 6.2.[2] The student coverage scores (75.9% for Stat2015, 82.4% for Stat2016, and 76.9% for CS2016) highlight the effectiveness of the current annotation scheme, with a majority of students covered by the human summaries.

[2] While there are 22 lectures in total for Stat2015, unfortunately, only 11 of them have phrase highlighting.

Course	Students	Lectures	Averaged by Lecture/Prompt				
			Responses	Words	Words/Res.	Highlights	Coverage
Stat2015	66	11	34.1	156.5	4.5	27.8	75.9%
Stat2016	74	24	41.9	161.8	3.7	37.2	82.4%
CS2016	38	23	22.4	217.1	9.5	20.0	76.9%

Table 6.2: Basic statistics of the dataset. Because the student responses and human summaries are created for each lecture and prompt, we take the average of the corresponding statistics.

6.2 PROPOSED APPROACH

We describe three improvements to the phrase-based summarization framework. Our first improvement involves a supervised approach to candidate phrase extraction (§6.2.1). Next, we learn to predict the pairwise phrase similarity (§6.2.2). Further, we explore a community detection algorithm to group the phrases into clusters (§6.2.3). We use the cluster size as an approximation to the number of student supporters for all the phrases within the cluster. L&L (Luo and Litman, 2015) adopt LexRank (Erkan and Radev, 2004) to finally choose one representative phrase from each cluster. We follow the convention in this study. Note that our focus of this chapter is not on developing new algorithms but to explore new capabilities that are enabled by the highlighting scheme. We thus perform direct comparisons with approaches described in L&L and leave comparisons to other approaches to future work. We present an intrinsic evaluation of each improvement in this section, followed by a comprehensive extrinsic evaluation in §6.3.

6.2.1 Candidate phrase extraction

The phrase-based highlighting scheme lends itself to a supervised phrase extraction approach. In contrast, L&L used heuristics to extract noun phrases (NPs) only. This limitation has meant that informative non-NP phrases such as “how confidence intervals linked with previous topics” will be excluded from the summary, whereas uninformative NP phrases such as "the most interesting point" may be included.

We attempt to resolve this issue by formulating candidate phrase extraction as a word-level sequence labeling task. Concretely, we aim to assign a label to each word in the student responses. We choose to use the 'BIO' labeling scheme, where 'B' stands for the beginning of a phrase, 'I' for continuation of a phrase, 'O' for outside of a phrase. For example, " The (B) central (I) limit (I) and (O) normal (B) approximations (I) " illustrates the tagging of individual words, where the "The central limit" and "normal approximations" are two phrases highlighted by our annotators.

Local Features	• Word trigram within a 5-word window • Part-of-Speech tag trigram within a 5-word window • Chunk tag trigram within a 5-word window • Whether the word is in the prompt • Whether the word is a stopword • Label bigrams.
Global Features	• Total number of word occurrences (stemmed) • Rank of the word's term frequency

Table 6.3: Local and global features for supervised phrase extraction. Local features are extracted within one student's response. Global features are extracted using all student responses to a prompt in one lecture.

We choose to use the Conditional Random Fields (CRF) (Lafferty et al., 2001) as our sequence labeler[3] and develop a number of features (Table 6.3) based on sentence syntactic structure and word importance to signal the likelihood of a word being included in the candidate phrase. During training, we merge the phrase highlights produced by two annotators in order to form a large pool of training instances. When two highlights overlap completely, e.g., "normal approximations" are marked by both annotators using different colors, we keep only one instance of the phrase. When the highlights partially overlap, we use each phrase highlight as a separate training instance, resulting in 1,115, 2,682 and 1,189 instances for Stat2015, Stat2016 and CS2016 respectively. In this and all the following experiments, we perform leave-one-lecture-out cross-validation on all the lectures and report results averaged across folds. Table 6.4 presents the intrinsic evaluation results on the phrase extraction task.

[3] We use the implementation of Wapiti (Lavergne et al., 2010) with default parameters.

We calculate Precision (P), Recall (R) and F-measure (F) scores based on the exact match of system phrases to gold-standard phrases. While the sequence labeling approach and the features presented here are straightforward, they do produce a collection of candidate phrases with higher precision. It removes noun phrases that are commonly used by students but uninformative (e.g., “a little bit abstract”, “a problem with today’s topic”) as they were not highlighted by annotators. Phrase well-formedness is highly important to the summary quality, as evaluated in §6.3.

	Stat2015			Stat2016			CS2016		
Extraction	P	R	F	P	R	F	P	R	F
L&L (NPs only)	.426	**.633**	.503	.538	.714	.609	.199	.387	.256
Sequence Labeling	**.692***	.569*	**.618***	**.771***	**.743**	**.753***	**.577***	**.402**	**.468***

Table 6.4: Results of phrase extraction, intrinsically evaluated by comparing the system phrases to gold-standard phrases using exact match. The highest score in each column is shown in bold. * means the difference is significant with $p < 0.05$.

6.2.2 Ensemble similarity learning

Accurately estimating pairwise phrase similarity plays an essential role in phrase-based summarization. Better similarity learning helps produce better phrase clusters, which in turn leads to more accurate estimation of the number of student supporters for each summary phrase. While a human annotator could distinguish the semantic similarity or dissimilarity of the phrase highlights, it remains unclear if a single similarity metric could fulfill this goal or if we may need an ensemble of different metrics.

L&L calculate the pairwise phrase similarity using SEMILAR (Rus et al., 2013) with the latent semantic analysis (LSA) trained on the Touchstone corpus (Ştefănescu et al., 2014). One drawback of this approach is that the similarity of phrases that do not appear in a background corpus cannot be captured. In this work, we develop an ensemble of similarity metrics by feeding them into a supervised classification framework. We use the phrase highlights as supervision, where phrases of the same color are positive examples and those of different colors are negative examples. We experiment with a range of metrics for measuring lexical similarity, including lexical overlap (Rus et al., 2013), cosine similarity, LIN similarity (Miller, 1995), BLEU (Papineni et al., 2002), SimSum (Lin, 2004), Word Embedding (Goldberg and Levy, 2014), and LSA (Deerwester et al., 1990). LIN similarity is based on WordNet definitions. Lexical overlap, cosine similarity, BLEU, and SimSum are related to how many words the two phrases have in common, while Word Embedding and LSA both capture the phrase similarity in a low dimensional semantic space. Therefore, we use an ensemble of the above similarity metrics by feeding them as features in a SVM classification model, assuming it will be better suited for this task than the LSA alone. Table 6.5 presents the intrinsic evaluation results. LSA has a poor degree of coverage (low recall) with many phrase similarities not being picked up by the metric.

	Stat2015			Stat2016			CS2016		
Similarity	P	R	F	P	R	F	P	R	F
L&L (LSA)	**.904**	.665	.730	.878	.506	.584	.856	.840	.820
Ensemble learning	.895	**.801****	**.833***	**.943***	**.768***	**.836***	**.867***	**.852***	**.836***

Table 6.5: Results of predicting pairwise phrase similarity, measured using classification P/R/F.

6.2.3 Phrase clustering

L&L use k-medoids for phrase clustering. It is a greedy iterative clustering algorithm (Kaufman and Rousseeuw, 1987), which may suffer from local minima. We instead treat phrase clustering as a community detection problem. We define a **community** as a set of phrases that are semantically similar to each other, as compared to the rest of the phrases in student responses (Malliaros and Vazirgiannis, 2013). In our formulation, we consider each candidate phrase as a node in the network graph. We create an edge between two nodes if the two phrases are considered semantically similar to each other using the above similarity learning approach. Our goal is to identify tightly connected phrase communities in the network structure. The community size is used as a proxy for the number of students who semantically mention the phrase. Community detection has seen considerable success in tasks such as word sense disambiguation (Jurgens, 2011), medical query analysis (Campbell et al., 2014), and automatic summarization (Qazvinian and Radev, 2011; Mehdad et al., 2013).

Phrase Clustering	Stat2015	Stat2016	CS2016
L&L (k-medoids)	82.2%	84.0%	76.5%
Community Detection with OSLOM	**85.2%***	**88.8%***	**85.9%***

Table 6.6: Results of phrase clustering measured by purity: ratio of number of phrases agreeing with the majority color in clusters.

We use OSLOM (Order Statistics Local Optimization Method) (Lancichinetti et al., 2011) in this work. It is a widely used community detection algorithm that detects community structures (i.e., clusters of vertices) from a weighted, directed network. It optimizes locally the statistical significance of clusters with respect to a global null model during community expansion. We use an undirected version of OSLOM and set the p-value as 1.0 to encourage more communities to be identified[4] since the number of vertices in the constructed graph is relatively small compared to large complex networks. The key feature of OSLOM is that it supports finding overlapped community structures and orphaned vertices, offering more flexibility in the clustering process than k-medoids. We want to investigate if the unique characteristics of OSLOM allow it to produce better phrase clusters, hence more accurate estimation of the number of student supporters. We conduct an intrinsic evaluation using purity, corresponding to the percentage of phrases in the cluster that agree with the majority color. Results are presented in Table 6.6. While this metric by itself is not thorough enough, it does highlight the strength of the community detection approach in generating cohesive clusters. One advantage of OSLOM we found is that it will treat a phrase different from any other phrase as a singleton, while this phrase must be assigned to one of the clusters in k-medoids, resulting in a noisy cluster.

[4] L&L set the number of clusters is to be the square root of the number of extracted phrases.

6.3 SUMMARY EVALUATION

The previous section described three improvements to the phrase summarization framework. Next, we evaluate them on the end task of summarizing student course responses. The phrase summaries are evaluated along two dimensions: we expect ROUGE (Lin, 2004) to measure the informativeness of the summary text content (§6.3.1); we further propose a new metric to quantify to what extent the most pressing student needs have been captured in the summary (§6.3.2).

6.3.1 ROUGE and human evaluation

ROUGE measures the n-gram overlap between system and human summaries. In this work, we report R-1, and R-2 scores, which respectively measure the overlap of unigrams and bigrams. We also perform human evaluation, similar to what we did in §4.3.2 and we select two system-system pairs: CDSum vs. PhraseSum and SequenceSum vs. PhraseSum. We name the phrase summarization framework described in Luo and Litman (2015) as **PhraseSum**. The summary is limited to 5 phrases or less in all experiments, corresponding to the length limit given to human annotators. Note that, the summary length is set independently of the number of clusters. If the number of clusters produced in §6.2.3 is less than 5, the phrase number is equal to the cluster number.

The summarization performance is shown in Table 6.7 (the caption explains the system names). For our enhancements of PhraseSum, the proposed supervised phrase extraction (SequenceSum) significantly improves P and thus improves (mostly significantly) F as well. SimSum is slightly better than SequenceSum for R and F, however, it is not significant using a two-tailed paired t-test. It suggests that a supervised method is not necessarily better than an unsupervised model in terms of the end-task performance, and its improvement over the PhraseSum baseline is mainly due to the supervised phrase extraction step. In fact, the predicted similarity scores using the similarity learning model and the LSA model are highly correlated to each other ($r = 0.852$, $p < 0.01$) although it has a better classification performance (Table 6.5). Although CDSum is not significantly different from SequenceSum for the

Course	System	R-1			R-2			Human Preference
		P	R	F	P	R	F	
Stat2015	PhraseSum	.402	**.466**	.415	.170	.208	.178	**64.3%**
	SequenceSum	.600*	.448	.493*	.307*	.231	.249*	11.4%*
	SimSum	.597*	.460	**.504***	.302*	**.241**	.260*	-
	CDSum	**.634***	.435	.499*	**.335***	.229	**.262***	10.0%*
Stat2016	PhraseSum	.492	**.545**	.508	.231	.258	.239	**50.3%**
	SequenceSum	.618*	.485*	.531	.347*	.267	.294*	21.1%*
	SimSum	.618*	.500*	.543	.353*	**.284**	.309*	-
	CDSum	**.702***†	.480*	**.550***	**.433***†	.279	**.324***	19.9%*
CS2016	PhraseSum	.276	**.344**	.283	.080	.088	.077	**68.8%**
	SequenceSum	.470*	.253*	.287	.142*	.069	.083	12.2%*
	SimSum	.575*†	.300	**.375***†	.236*†	**.119***†	**.148***†	-
	CDSum	**.652***†	.274*	.351*†	**.249***†	.101†	.130*†	12.0%*

Table 6.7: Summarization Performance. **SequenceSum** means replacing the syntax phrase extraction in the PhraseSum baseline with the supervised sequence labeling phrase extraction. **SimSum** means replacing not only the phrase extraction but also the similarity scores using the supervised models. **CDSum** means using all three proposed techniques including the community detection. * indicates that the difference is statistically significant compared to PhraseSum with $p < 0.05$. † means that the difference over SequenceSum is statistically significant with $p < 0.05$.

Stat2015, it does improve P significantly for all ROUGE metrics for Stat2016 and CS2016. One possible explanation is that the latter courses have higher student response ratios, and thus benefit more from the community detection as the graph is larger. Unfortunately, for human evaluation, turkers prefer significantly more to PhraseSum, which has a higher recall of R-1 than other methods.

6.3.2 A new metric based on color matching

Our goal is to create a comprehensive evaluation metric that takes into account the following two factors.

- **Phrase matching.** While ROUGE is a classic summarization evaluation metric, it trivially compares the system vs. human summaries based on surface text form. In contrast, the phrase highlights allow us to accurately match the phrases in the system summary to those in the human summary based on color matching. This is due to two facts: first, our methods are extractive-based and all candidate phrases are extracted from the student responses; second, in the new highlighting scheme, the annotators are asked to highlight both the human summary phrase and any phrases in the student responses that are semantically the same with the summary phrase using the same color. It thus becomes easy to track the colors of the extracted phrases and verify if they match any of those in the human summary.
- **Student supporters.** Each summary phrase is tagged with the number of students who raise the issue. For human summary, this number is created by human annotators. For system summary, we approximate this number using the size of the cluster, from which the summary phrase is extracted.

Our proposed new metric resembles precision, recall, and F-measure. We define the true positive (TP) as the number of *shared colors* between system and human summaries. Each color is weighted by the number of student supporters, taken as the smaller value between system and human estimates. The *precision* is defined as TP over the total number of colors in the *system* summary, each weighted by system estimates; while *recall* is defined as TP over the total number of colors in the *human* summary, each weighted by human estimates. For example, assuming the phrases in the human summary are colored and tagged with estimates on student support: yellow/12, green/9, red/6, blue/5, magenta/3; similarly the phrases in the system summary are colored and tagged: yellow/11+3, green/17, red/7, blue/7. There are two phrases in the system summary that bear the same color, we thus add up the system estimates into yellow/11+3 (see Human Summary 1 in Table 6.1 and SequenceSum in Table 6.9). There are 4 shared colors between system and human summaries. The true positive is calculated as: 12+9+6+5 = 32. The precision is $32/((11+3)+17+7+7) = 0.711$, and recall is $32/(12 + 9 + 6 + 5 + 3) = 0.914$. The F-measure is calculated as the harmonic mean of precision and recall scores.

The performance is shown in Table 6.8. Similar to the ROUGE evaluation, SequenceSum improves the P and F significantly. Now, CDSum not only significantly improves P, but also F for Stat2016. Note that, the P improves 156.1% and the F improves 68.7% relatively from PhraseSum and SequenceSum for CS2016. As we have mentioned, L&L calculate the pairwise phrase similarity using SEMILAR (Rus et al., 2013) with the latent semantic analysis (LSA) trained on the Touchstone corpus (Ştefănescu et al., 2014), collected in the discipline of Applied Science. However, domain technology words like "quicksort" (a sorting algorithm), "shellsort" (another sorting algorithm), "adt" (short for "abstract data type") often appear in student responses but not in the Touchstone background corpus and thus cannot be captured by LSA. Therefore, it benefits a lot from learning phrase extraction within the corpus itself, evidenced by a big jump of P/R/F in phrase extraction evaluation (Table 6.4).

	Stat2015			Stat2016			CS2016		
	P	R	F	P	R	F	P	R	F
PhraseSum	.349	.615	.437	.485	.747	.576	.228	**.574**	.316
SequenceSum	$.626^*$	**.642**	$\mathbf{.614}^*$	$.698^*$	.757	$.717^*$	$.584^*$	.520	$\mathbf{.533}^*$
SimSum	$.602^*$	.636	$.595^*$	$.711^*$	.753	$.723^*$	$.569^*$	.517	$.524^*$
CDSum	$\mathbf{.643}^*$	.634	$.613^*$	$\mathbf{.777}^{*\dagger}$	**.762**	$\mathbf{.759}^{*\dagger}$	$\mathbf{.753}^{*\dagger}$	$.408^{*\dagger}$	$.496^*$

Table 6.8: Evaluation based on the new metric of color matching. P, R, and F are averaged by the annotators.

6.3.3 Example summaries

The automatic summaries generated by different systems for the same example in Table 6.1 are shown in Table 6.9. The PhraseSum baseline extracts unnecessary content, which could be eliminated by the supervised phrase extraction model. For example, including "the example after" before "central limit theorem" makes it too specific. The "collapse" effect with a large cluster with unrelated items (Basu et al., 2013) can also be illustrated (e.g., the quantitative numbers for the phrase "i" in PhraseSum and "q-q plot" in "SequenceSum" are much larger than the gold standard). This is solved by the community detection algorithm where such bigger clusters will not be considered as a single community.

PhraseSum
- i [40]
- the example after central limit theorem ^{y} [12]
- q q plot ^{g} [9]
- the fact that we can sample as many as we want [9]
- last problem about normalization ^{m} [6]
SequenceSum
- q-q plot ^{g} [17]
- central limit theorem ^{y} [11]
- normal approximation to binomial ^{b} [7]
- sampling distributions ^{r} [7]
- clt ^{y} [3]
CDSum
- central limit theorem ^{y} [11]
- q-q plot ^{g} [10]
- sampling distributions ^{r} [7]
- normal approximation to binomial ^{b} [5]
- nothing [4]

Table 6.9: Example system summaries for the example in Table 6.1. Note, the highlights in these summaries are NOT annotated by human after they are generated. Instead, they are automatically extracted from the dataset (§6.3.2).

6.3.4 Cross-course evaluation

In previous experiments, we perform a leave-one-lecture-out cross-validation for each course. That is, one lecture was used for testing and all the other lectures were used to train the models. However, this setting might favor our supervised models with the lexical features because students may use the same words across lectures. For example, lecture 20, 21, 22 are about "hypothesis testing" in Stat2016 and the term "p value" often appears in students' responses in all three lectures. In the future, we will apply our approach to different courses, and thus it is better to evaluate it on different courses beyond lectures. Therefore, we train the supervised phrase extraction and similarity learning models with Stat2015 and test on Stat2016 and CS2016. Testing on Stat2016 simulates the situation that we develop a supervised model with a course and apply it to the same course but taught in the future. Testing on CS2016 simulates the case that we train a model on one course and test on another, which is more practical in real application.

The results are shown in Tables 6.10 and 6.11. As we can see, for both ROUGE scores and the new color-based metric, SequenceSum is no longer dominant over PhraseSum. However, SimSum and CDSum still achieve significantly better P, which shows that the latter two improvements are necessary. In addition, both SimSum and CDSum transfer very well from Stat2015 to CS2016, a relatively different course.

In sum, these results are encouraging, especially when we train the models with only 11 lectures in Stat2015 and test on all lectures in Stat2016 and CS2016.

6.4 SUMMARY

In this work, we introduced a new phrase-based highlighting scheme for automatic summarization. It highlights the phrases in the human summary and also the corresponding phrases in student responses. Enabled by the highlighting scheme, we improved the phrase-based summarization framework proposed by Luo and Litman (2015) by developing a supervised candidate phrase extraction, learning to estimate the phrase similarities, and experimenting with different clustering algorithms to group phrases into clusters. We further introduced a

Course	System	R-1			R-2		
		P	R	F	P	R	F
Stat2016	PhraseSum	.492	**.545**	.508	.231	**.258**	.239
	SequenceSum	.526	.452*	.476	.271	.228	.241
	SimSum	.589*†	.480*	**.520**	.304*	.236	**.260**
	CDSum	**.644***†	.455*	.516	**.348***†	.224	**.260**
CS2016	PhraseSum	.276	**.344**	.283	.080	**.088**	.077
	SequenceSum	.514*	.255*	.320	.163*	.072	.093
	SimSum	.504*	.254*	.320	.163*	.074	.096
	CDSum	**.549***	.271*	**.345***	**.197***	**.088**	**.114***

Table 6.10: Summarization Performance when training the supervised phrase extraction and similarity learning models with Stat2015.

	Stat2016			CS2016		
	P	R	F	P	R	F
PhraseSum	.485	.747	.576	.228	**.574**	.316
SequenceSum	**.624***	**.753**	**.671***	.540*	.455*	.475*
SimSum	.607*	.732	.649*	.547*	.451*	**.478***
CDSum	.608*†	.747	.656*†	**.629***†	.390*†	.449*

Table 6.11: Evaluation based on the new metric of color matching when training the supervised phrase extraction and similarity learning models with Stat2015.

new metric that offers a promising direction for making progress on developing automatic summarization evaluation metrics. Experimental results showed that our proposed methods not only yield better summarization performance evaluated using ROUGE, but also produce summaries that capture the pressing student needs. Therefore, H3.1, H3.2, and H3.3 are supported.

7.0 FUTURE DIRECTIONS

There are several remaining research questions that deserve consideration in the future.

7.1 SUMMARIZATION EVALUATION METRICS

ROUGE is often adopted in research papers to evaluate the quality of summarization because it is fast and is correlated well to human evaluation (Lin, 2004; Graham, 2015). At the same time, it is also criticized that ROUGE cannot thoroughly capture the semantic similarity between system and reference summaries. Therefore, many researchers supplement ROUGE with a manual evaluation. This is why we conduct evaluations using both ROUGE and human evaluation in this work. In addition, to supplement ROUGE, we proposed a new metric to evaluate summarization based on color matching, as introduced in §6.3.2. It considered semantic similarity between system and reference summaries by leveraging the phrase highlighting human annotation.

However, we found that a system with better ROUGE scores does not necessarily mean it is more preferred by humans (§4.3.2). For example, ILP is preferred more on all three review data sets even if it got lower ROUGE scores than the other systems. It coincides with the fact that the ILP generated shorter summaries in terms of number of sentences than the other two systems (Table 4.6). For phrase summarization, we noticed another pattern that people seem to like systems with high recalls (Table 5.1, Table 5.2, and Table 6.7).

This leaves some open questions to be explored:

- Which metric best describes the summarization quality?
- How automatic metrics correlate to human judges?

- Do people like systems with high recalls?
- Do people prefer shorter system summaries when the summaries to be compared are long?
- Similarly, do people prefer longer summaries when the summaries to be compared are short?
- How long are the summaries to make a difference?

A related question is that which system to choose when we want to deploy a summarization system in a real application?

7.2 MULTI-DOMAIN ISSUE

In this work, we evaluate our proposed methods across different genres for the sentence summarization, and across different courses for the phrase summarization and quantitative phrase summarization. In general, there is no winner for all data sets. For sentence summarization evaluated by ROUGE (Table 4.4), MEAD is the best one on camera; SumBasic is best on Stat2016 and mostly on Stat2015; ILP is best on DUC04; our method ILP+MC is best on peer review and mostly on Eng and CS2016. For phrase summarization, Cluster+LexRank achieved best ROUGE F for Eng (Table 5.1), while Cluster+Medoid achieved best ROUGE F for the other courses (Table 5.2). For quantitative phrase summarization (Table 6.8), CDSum won on Stat2016, but SequenseSum won on Stat2015 and CS2016 when evaluating by the color-matching F-measure.

We hypothesize that different methods favor different corpora with certain properties. We have explored the impact of $\alpha_{b=1}$ to the ILP-based approaches, the ratio of bigrams in human summaries that are in the input only once. However, we do not have a conclusive answer when our method will work better. In the future, we would like to consider more attributes, such as new metrics for diversity. For example, we do observe different distributions of number of student supporters (the number of students who raise the issue) in different student response data sets, as shown in Fig. 7.1. For CS2016, 78.2% of the human summary phrases associated a student supporter number less or equal than 5, compared to 61.0% for

Stat2015 and 53.5% for Stat2016. If the number of student supporter is high, the diversity is low since a large number of students have similar or exactly the same responses.

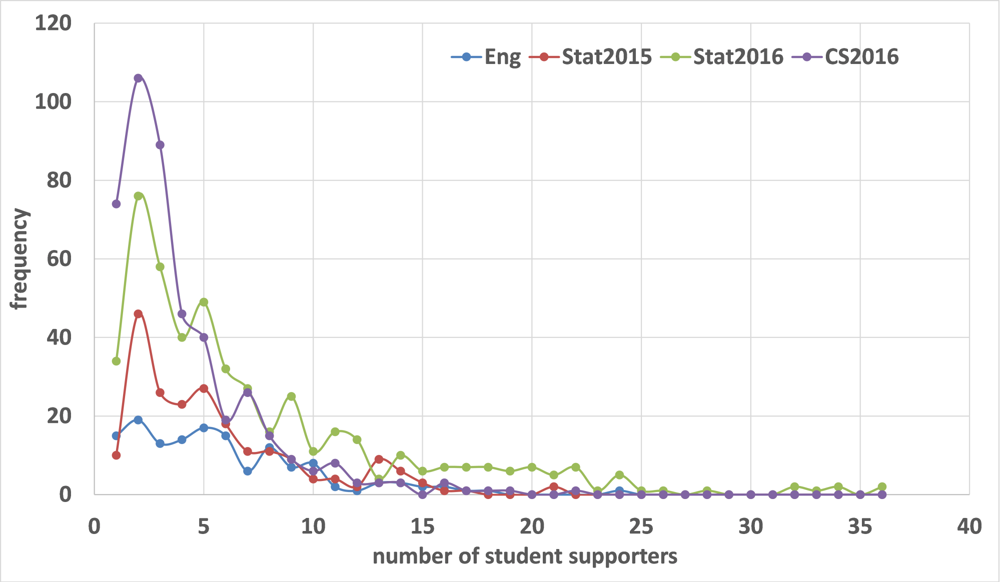


Figure 7.1: Distribution of number of student supporters annotated by humans. X axis is the actual number of student supporters. Y axis is the frequency that the number of student supporters occurs in human summaries. For example, there are 106 human summary phrases in CS2016 that are associated with 2 student supporters; while there are 19, 46, and 76 for Eng, Stat2015 and Stat2016 respectively.

7.3 BETTER SUMMARIZATION BY USING MORE RESOURCES

One limitation of our proposed methods is that they only take the student responses and human summaries into account and ignore other useful resources. First, additional external resources such as lecture slides and textbook can be used to develop a better candidate phrase extraction model. For example, domain-specific concepts can be extracted from the lecture slides. Second, domain knowledge from instructors may be utilized and a list of key concepts can be provided by instructors. Last, we may take advantage of the high quality student responses (Luo and Litman, 2016) to improve the summarization performance.

7.4 PHRASE VS. SENTENCE

As far as we know, our work is the first one to summarize student response at a phrase level. The motivation is to apply the proposed summarization method into mobile devices which have limited screen size. Another reason we chose summarization at the phrase level is due to the fact the first student summarization data set (student response Eng as introduced in §3.1) was created at the phrase level. However, whether summarization at a phrase level is better than summarization at a sentence level is not answered. This can be explored from two perspectives. First, for the usability, we can examine whether a phrase summary is easier to read but at the same time maintains similar information compared to a sentence summary. Second, for the learning gain, whether an instructor teaches better and students learn more with phrase summarization? Our ultimate goal of this work is to enhance student-instructor feedback and thus improve learning and teaching. Therefore, we can ask students and instructors to evaluate different types of summarization (sentence vs. phrase). We can also measure the learning gain with a controlled experiment by deploying one summarization system to half of the students in a course and another system to the other half of students. In this way, we can directly compare the learning effect by using different summarization systems.

7.5 LARGE-SCALE INTRINSIC EVALUATION FOR MATRIX COMPLETION

In §4.3.1, we noticed some interesting examples (shown in Table 4.1) that some bigrams are associated in a sentence but they do not appear in the sentence. We therefore performed an intrinsic evaluation about whether the low-rank matrix approximation captures similar bigrams or not. We confirmed that a bigram does receive a bigger score in a sentence that contains similar bigram(s) to it than a sentence that does not after the low-rank approximation. However, we only observe that a sentence gives significantly more partial scores to bigrams that are similar to its own bigrams than bigrams that are different from its own on one of the data sets. We hypothesize if we have a large-scale gold-standard data set that contains pairs of similar bigrams and pairs of different bigrams, we are able to observe significant results.

7.6 COMPARISON WITH OTHER APPROACHES

In §4.4, we investigated the impact of the ratio $\alpha_{b=1}$ on the original and extended ILP framework. However, seeing its impact to other baselines is also interesting.

In §6, we compared the proposed methods only with the phrase summarization (PhraseSum), because it supports the ability to compute the number of student supporters. However, it is possible to extend existing summarization algorithms such as SumBasic, LexRank to estimate the number of student supporters. One straightforward method is to post-processes the generated summaries by counting similar phrases to the summaries. However, it needs additional parameters such as a similarity threshold.

7.7 BEYOND STUDENT RESPONSES

For the phrase summarization, we are eager to apply it to other types of data sets beyond student responses, such as product or peer reviews. The current challenge is that we have not found such a data set annotated with phrase summaries. This can be solved by annotating existing data sets with phrase summaries or collecting new data sets. In this way, we can test the generalizability of the proposed phrase summarization beyond student responses.

7.8 NUMBER OF CLUSTERS

For the community detection algorithm, OSLOM, we set the p-value as 1.0 to encourage more communities to be identified while the phrase summarization proposed by Luo and

Litman (2015) set the number of clusters is to be the square root of the number of extracted phrases. However, we found that the community detection algorithm identified less number of clusters than k-medoids. Therefore, it might be interesting to investigate how the number of clusters affects the summarization performance.

7.9 UPPER BOUND OF THE PHRASE SUMMARIZATION FRAMEWORK

In §6, we improved the phrase-based summarization framework by developing a supervised candidate phrase extraction, learning to estimate the phrase similarities, and experimenting with different clustering algorithms to group phrases into clusters.

What is the upper bound by improving each step in this framework? Recall that the proposed phrase summarization involves three stages: *candidate phrase extraction*, *phrase clustering*, and *phrase ranking*.

To determine the upper bound, we replace the phrase extraction, clustering and phrase ranking steps using the human annotations. In specific, for candidate phrase extraction, instead of using a syntax parser or a sequence labeling model to extract candidate phrases, we use the human highlighted phrases. For the phrase clustering, we group phrases in the same color instead of using an automatic clustering algorithm. For phrase ranking, we can choose the phrase that maximizes the ROUGE scores instead of using LexRank to select the representative phrase in each cluster.

The results are presented in Table 7.1 using the two annotators[1]. Using human annotations improves the ROUGE scores for each step. The oracle candidate phrase extraction improves R-2 F scores of the baseline by 8.5% for Stat2015, 8.1% for Stat2016 and 4.4% for CS2016 (absolute values, averaged by two annotators). The oracle clustering yields an averaged improvement of 5.3%, 7.5%, and 9.6% compared to the '+clustering' for the three courses respectively. Note that the oracle clustering yields better performance gain than

[1] Different annotators may have different phrase selections and phrase highlights. The ROUGE scores are obtained using both annotators.

			R-1			R-2		
			P	R	F	P	R	F
Stat2015	PhraseSum		.466	.402	.415	.208	.170	.178
	A1	+extraction	.472	.612*	.523*	.262	.361*	.298*
		+clustering	.570*†	.664*	.603*†	.309*†	.383*	.337*
		+ranking	.642$^{*\dagger\circ}$	**.769**$^{*\dagger\circ}$	**.691**$^{*\dagger\circ}$	.398$^{*\dagger\circ}$	**.478**$^{*\dagger\circ}$	**.429**$^{*\dagger\circ}$
	A2	+extraction	.465	.538*	.481	.221	.252	.227
		+clustering	.601*†	.555*	.563*†	.306*†	.297*	.294*†
		+ranking	**.671**$^{*\dagger\circ}$	.668$^{*\dagger\circ}$	.657$^{*\dagger\circ}$	**.414**$^{*\dagger\circ}$	.420$^{*\dagger\circ}$	.409$^{*\dagger\circ}$
Stat2016	PhraseSum		.545	.492	.508	.258	.231	.239
	A1	+extraction	.498*	.673*	.559*	.272	.384*	.311*
		+clustering	.647*†	.722*	.674*†	.382*†	.428*	.398*†
		+ranking	.709$^{*\dagger\circ}$	.781$^{*\dagger\circ}$	.736$^{*\dagger\circ}$	.473$^{*\dagger\circ}$	.524$^{*\dagger\circ}$	.491$^{*\dagger\circ}$
	A2	+extraction	.520	.649*	.564*	.302*	.383*	.329*
		+clustering	.647*†	.705*†	.668*†	.382*†	.412*	.392*†
		+ranking	**.716**$^{*\dagger\circ}$	**.791**$^{*\dagger\circ}$	**.747**$^{*\dagger\circ}$	**.510**$^{*\dagger\circ}$	**.561**$^{*\dagger\circ}$	**.531**$^{*\dagger\circ}$
CS2016	PhraseSum		.344	.276	.283	.088	.080	.077
	A1	+extraction	.298	.454*	.343*	.110	.173*	.126*
		+clustering	.525*†	.542*†	.523*†	.245*†	.259*†	.246*†
		+ranking	.565$^{*\dagger\circ}$	**.631**$^{*\dagger\circ}$	**.583**$^{*\dagger\circ}$	**.321**$^{*\dagger\circ}$	**.363**$^{*\dagger\circ}$	**.333**$^{*\dagger\circ}$
	A2	+extraction	.351	.396*	.358*	.113	.129*	.115*
		+clustering	.518*†	.462*†	.475*†	.201*†	.186*†	.187*†
		+ranking	**.572**$^{*\dagger\circ}$	.538$^{*\dagger\circ}$	.540$^{*\dagger\circ}$	.255$^{*\dagger\circ}$	.247$^{*\dagger\circ}$	.244$^{*\dagger\circ}$

Table 7.1: ROUGE scores using human annotations in the phrase summarization. 'A1' uses the 1st annotator's highlights. 'A2' uses the 2nd annotator's highlights. '+extraction' uses human highlighted phrases as candidate phrases. '+clustering' groups phrases in the same highlighted color. '+ranking' selects the phrase that maximizes R-1. *, †, and $^{\circ}$ mean significantly better than PhraseSum, '+extraction' and '+ranking' respectively.

oracle phrase extraction for CS2016. This correlates our findings that SimSum and CDSum achieved significant better ROUGE scores for CS2016 but not SequenceSum (Table 6.7). However, the most important step is the last phrase selection step, which we ignored in the proposed quantitative phrase summarization. It improves the R-2 by 16.2%, 19.3% and 12.7% respectively compared to '+clustering'. Based on this observation, it is desirable to design new methods to select the most important phrase in a cluster. For example, we can learn a supervised model to score phrases.

8.0 CONCLUSIONS

Effective teachers use student feedback to adjust their teaching strategies. Nowadays, in large classes, there is far too much feedback for a single teacher to manage and attend to. If different perspectives in the student feedback could be summarized and pressing issues identified, it would greatly enhance the teachers' ability to make informed choices. Our emphasis is on the textual feedback submitted by students after each lecture in response to the following reflective prompts: 1) "Describe what you found most interesting in today's class", 2) "Describe what was confusing or needed more detail." and 3) "Describe what you learned about how you learn." Education researchers have demonstrated that asking students to respond to reflection prompts can improve both teaching and learning. However, summarizing these responses for large classes (e.g., introductory STEM, MOOCs) remains costly, time-consuming, and an onerous task for humans. In this thesis, we seek to automatically summarize the student course feedback, which is challenging from both the input perspective and output perspective. First, there is a high **lexical variety** issue, because students tend to use different word expressions to communicate the same or similar meanings (e.g., "bike elements" vs. "bicycle parts"). Second, there is also a high **length variety** issue, as the student responses range from a single word to multiple sentences. Third, there is a **redundancy** issue since some content among student responses are not useful (e.g., including phrases such as "the most interesting point" in the summary is a waste of space given that the prompt is asking "Describe what you found most interesting in today's class".). Fourth, our human summaries consist of a list of important phrases (**phrase scale**) instead of sentences, which is very different from existing summarization corpora. Last, from an instructor's perspective, the quantitative number of students (**quantity**) who have a particular problem or are interested in a particular topic is valuable.

To address such challenges, we developed different techniques to summarize student responses to reflective prompts at multiple levels of granularity.

Following the line of existing summarization research work, we first proposed a novel summarization algorithm at the sentence level, by extending an ILP-based framework with a low-rank matrix approximation in order to address the challenge of lexical variety. The low-rank matrix approximation process makes two notable changes to the existing ILP framework. First, it extends the domain of the co-occurrence matrix from binary to a continuous scale, which offers a better sentence-level semantic representation. Second, the binary concept variables are also relaxed to a continuous domain, which allows the concepts to be "partially" included in the summary. To evaluate the proposed method, we construct gold-standard pairs of similar bigrams and pairs of different bigrams from our student response data sets, with the goal to test whether the low-rank matrix approximation helps to capture similar concepts. It confirmed that a bigram does receive more partial score in a sentence that contains similar bigram(s) to it than a sentence that does not. We also evaluate the proposed approach automatically based on ROUGE scores and manually based on Amazon Mechanical Turk. Our method shows promising results against a range of baselines on the two student responses. We also apply the method to other data sets including product and peer reviews, news articles. To understand when and why our proposed method works, we investigated a variety of attributes that might impact the performance. Unfortunately, we do not have a conclusive answer yet.

With the goal to aggregate and display summaries into mobile devices which have limited screen size, we proposed a phrase summarization algorithm in order to address the **phrase scale**. To address length variety and redundancy challenges, we extracted phrases rather than sentences to form summaries. To address the lexical variety and quantity challenges, we adopted a metric clustering paradigm with a semantic distance to group extracted phrases. Experimental results showed the effectiveness on all student response data sets.

Also at the phrase level, we proposed a quantitative phrase summarization algorithm in order to estimate the number of students who semantically mention the phrases in a summary, addressing the quantity challenge. We first introduced a new phrase-based highlighting scheme for automatic summarization. It highlights the phrases in the human summary and also the corresponding phrases in student responses. Enabled by the highlighting scheme, we improved the phrase-based summarization framework proposed by Luo and Litman (2015) by developing a supervised candidate phrase extraction, learning to estimate the phrase similarities, and experimenting with different clustering algorithms to group phrases into clusters. We further introduced a new metric that offers a promising direction for making progress on developing automatic summarization evaluation metrics. Experimental results show that our proposed methods not only yield better summarization performance evaluated using ROUGE, but also produce summaries that capture the pressing student needs on three student response courses.

BIBLIOGRAPHY

Aleven, V. A. and Koedinger, K. R. (2002). An effective metacognitive strategy: Learning by doing and explaining with a computer-based cognitive tutor. *Cognitive science*, 26(2):147–179.

Almeida, M. and Martins, A. (2013). Fast and robust compressive summarization with dual decomposition and multi-task learning. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 196–206, Sofia, Bulgaria. Association for Computational Linguistics.

Baird, J. R., Fensham, P. J., Gunstone, R. F., and White, R. T. (1991). The importance of reflection in improving science teaching and learning. *Journal of research in Science Teaching*, 28(2):163–182.

Barker, E., Paramita, M. L., Aker, A., Kurtic, E., Hepple, M., and Gaizauskas, R. (2016). The sensei annotated corpus: Human summaries of reader comment conversations in online news. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 42–52, Los Angeles. Association for Computational Linguistics.

Basu, S., Jacobs, C., and Vanderwende, L. (2013). Powergrading: a clustering approach to amplify human effort for short answer grading. *Transactions of the Association for Computational Linguistics*, 1:391–402.

Berg-Kirkpatrick, T., Gillick, D., and Klein, D. (2011). Jointly learning to extract and compress. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 481–490, Portland, Oregon, USA. Association for Computational Linguistics.

Boud, D., Keogh, R., Walker, D., et al. (2013). *Reflection: Turning experience into learning.* Routledge.

Brooks, B. J., Gilbuena, D. M., Krause, S., and Koretsky, M. D. (2014). Using word clouds for fast, formative assessment of students' short written responses. *Chemical Engineering Education*, 48(4):190–198.

Buyukkokten, O., Garcia-Molina, H., and Paepcke, A. (2001). Seeing the whole in parts: Text summarization for web browsing on handheld devices. In *Proceedings of the 10th International Conference on World Wide Web*, WWW '01, pages 652–662, New York, NY, USA. ACM.

Callister, W. D. and Rethwisch, D. G. (2010). *Materials science and engineering: An introduction*, volume 8. Wiley New York.

Campbell, W., Baseman, E., and Greenfield, K. (2014). Content+context=classification: Examining the roles of social interactions and linguist content in Twitter user classification. In *Proceedings of the Second Workshop on Natural Language Processing for Social Media*, pages 59–65, Dublin, Ireland.

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

Cho, K. (2008). Machine classification of peer comments in physics. In *Educational Data Mining 2008*, pages 192–196.

Chopra, S., Auli, M., and Rush, A. M. (2016). Abstractive sentence summarization with attentive recurrent neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 93–98, San Diego, California. Association for Computational Linguistics.

Collobert, R. (2011). Deep learning for efficient discriminative parsing. In Gordon, G. J. and Dunson, D. B., editors, *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics (AISTATS-11)*, volume 15, pages 224–232. Journal of Machine Learning Research - Workshop and Conference Proceedings.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391.

Durrett, G., Berg-Kirkpatrick, T., and Klein, D. (2016). Learning-based single-document summarization with compression and anaphoricity constraints. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1998–2008, Berlin, Germany. Association for Computational Linguistics.

Erkan, G. and Radev, D. R. (2004). LexRank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 22(1):457–479.

Fan, X., Luo, W., Menekse, M., Litman, D., and Wang, J. (2015). CourseMIRROR: Enhancing large classroom instructor-student interactions via mobile interfaces and natural language processing. In *Works-In-Progress of ACM Conference on Human Factors in Computing Systems*. ACM.

Fan, X., Luo, W., Menekse, M., Litman, D., and Wang, J. (2017). Scaling reflection prompts in large classrooms via mobile interfaces and natural language processing. In *Proceedings of 22nd ACM Conference on Intelligent User Interfaces (IUI 2017)*.

Galanis, D., Lampouras, G., and Androutsopoulos, I. (2012). Extractive multi-document summarization with integer linear programming and support vector regression. In *Proceedings of COLING*.

Gillick, D. and Favre, B. (2009). A scalable global model for summarization. In *Proceedings of the Workshop on Integer Linear Programming for Natural Langauge Processing*, pages 10–18. Association for Computational Linguistics.

Gillick, D., Favre, B., and Hakkani-Tür, D. (2008). The ICSI summarization system at TAC 2008. In *Proceedings of TAC*.

Gillick, D., Favre, B., Hakkani-Tur, D., Bohnet, B., Liu, Y., and Xie, S. (2009). The ICSI/UTD summarization system at TAC 2009. In *Proceedings of TAC*.

Glassman, E. L., Kim, J., Monroy-Hernández, A., and Morris, M. R. (2015). Mudslide: A spatially anchored census of student confusion for online lecture videos. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 1555–1564. ACM.

Goldberg, Y. and Levy, O. (2014). word2vec explained: Deriving Mikolov et al.'s negative-sampling word-embedding method. *arXiv preprint arXiv:1402.3722*.

Gorinski, P. J. and Lapata, M. (2015). Movie script summarization as graph-based scene extraction. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1066–1076, Denver, Colorado. Association for Computational Linguistics.

Graham, Y. (2015). Re-evaluating automatic summarization with bleu and 192 shades of rouge. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 128–137, Lisbon, Portugal. Association for Computational Linguistics.

Gung, J. and Kalita, J. (2012). Summarization of historical articles using temporal event clustering. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL HLT '12, pages 631–635, Stroudsburg, PA, USA. Association for Computational Linguistics.

Harwood, W. S. (1996). The one-minute paper. *Journal of Chemical Education*, 73(3):229.

Hasan, K. S. and Ng, V. (2014). Automatic keyphrase extraction: A survey of the state of the art. In *Proceedings of the 52nd Annual Meeting of the Association for Computa-*

tional Linguistics, pages 1262–1273, Baltimore, Maryland. Association for Computational Linguistics.

He, Z., Chen, C., Bu, J., Wang, C., Zhang, L., Cai, D., and He, X. (2012). Document summarization based on data reconstruction. In *Proceedings of AAAI*.

Hong, K., Conroy, J., Favre, B., Kulesza, A., Lin, H., and Nenkova, A. (2014). A repository of state of the art and competitive baseline summaries for generic news summarization. In Calzolari, N., Choukri, K., Declerck, T., Loftsson, H., Maegaard, B., Mariani, J., Moreno, A., Odijk, J., and Piperidis, S., editors, *Proceedings of LREC*, pages 1608–1616, Reykjavik, Iceland. ACL Anthology Identifier: L14-1070.

Jindal, N. and Liu, B. (2008). Opinion spam and analysis. In *Proceedings of the 2008 International Conference on Web Search and Data Mining*, pages 219–230. ACM.

Jones, S., Jones, M., and Deo, S. (2004). Using keyphrases as search result surrogates on small screen devices. *Personal Ubiquitous Comput.*, 8(1):55–68.

Jurgens, D. (2011). Word sense induction by community detection. In *Proceedings of TextGraphs-6 Workshop*, pages 24–28, Portland, Oregon.

Kan, M.-Y. (2015). Keywords, phrases, clauses and sentences: topicality, indicativeness and informativeness at scales. In *Proceedings of the ACL 2015 Workshop on Novel Computational Approaches to Keyphrase Extraction*, page 1, Beijing, China. Association for Computational Linguistics.

Kaufman, L. and Rousseeuw, P. (1987). Clustering by means of medoids. *Statistical Data Analysis Based on the L1-Norm and Related Method*, pages 405–416.

Kiddon, C., Zettlemoyer, L., and Choi, Y. (2016). Globally coherent text generation with neural checklist models. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 329–339, Austin, Texas. Association for Computational Linguistics.

Lafferty, J. D., McCallum, A., and Pereira, F. C. N. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Lancichinetti, A., Radicchi, F., Ramasco, J. J., and Fortunato, S. (2011). Finding statistically significant communities in networks. *PloS one*, 6(4):e18961.

Lavergne, T., Cappé, O., and Yvon, F. (2010). Practical very large scale CRFs. In *Proceedings the 48th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 504–513. Association for Computational Linguistics.

Li, C., Liu, F., Weng, F., and Liu, Y. (2013a). Document summarization via guided sentence compression. In *Proceedings of the 2013 Conference on Empirical Methods in Natural*

Language Processing, pages 490–500, Seattle, Washington, USA. Association for Computational Linguistics.

Li, C., Qian, X., and Liu, Y. (2013b). Using supervised bigram-based ILP for extractive summarization. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1004–1013, Sofia, Bulgaria. Association for Computational Linguistics.

Li, C., Wei, Z., Liu, Y., Jin, Y., and Huang, F. (2016a). Using relevant public posts to enhance news article summarization. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 557–566, Osaka, Japan. The COLING 2016 Organizing Committee.

Li, W., He, L., and Zhuge, H. (2016b). Abstractive news summarization based on event semantic link network. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 236–246, Osaka, Japan. The COLING 2016 Organizing Committee.

Li, Y. and Li, S. (2014). Query-focused multi-document summarization: Combining a topic model with graph-based semi-supervised learning. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1197–1207, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Lin, C.-Y. (2004). ROUGE: a package for automatic evaluation of summaries. In *Proceedings of the Workshop on Text Summarization Branches Out*, volume 8. Barcelona, Spain.

Lin, H. and Bilmes, J. (2010). Multi-document summarization via budgeted maximization of submodular functions. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 912–920. Association for Computational Linguistics.

Liu, F., Flanigan, J., Thomson, S., Sadeh, N., and Smith, N. A. (2015). Toward abstractive summarization using semantic representations. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1077–1086, Denver, Colorado. Association for Computational Linguistics.

Liu, F. and Liu, Y. (2010). Exploring correlation between rouge and human evaluation on meeting summaries. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(1):187–196.

Liu, Z., Li, P., Zheng, Y., and Sun, M. (2009). Clustering to find exemplar terms for keyphrase extraction. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, EMNLP '09, pages 257–266, Stroudsburg, PA, USA. Association for Computational Linguistics.

Loza, V., Lahiri, S., Mihalcea, R., and Lai, P.-H. (2014). Building a dataset for summarization and keyword extraction from emails. In *Proceedings of LREC*, pages 2441–2446, Reykjavik, Iceland.

Luhn, H. P. (1958). The automatic creation of literature abstracts. *IBM Journal of research and development*, 2(2):159–165.

Luo, W., Fan, X., Menekse, M., Wang, J., and Litman, D. (2015). Enhancing instructor-student and student-student interactions with mobile interfaces and summarization. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 16–20, Denver, Colorado. Association for Computational Linguistics.

Luo, W. and Litman, D. (2015). Summarizing student responses to reflection prompts. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1955–1960, Lisbon, Portugal. Association for Computational Linguistics.

Luo, W. and Litman, D. (2016). Determining the quality of a student reflective response. In *Proceedings 29th International FLAIRS Conference*, Key Largo, FL.

Luo, W., Liu, F., and Litman, D. (2016a). An improved phrase-based approach to annotating and summarizing student course responses. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 53–63, Osaka, Japan. The COLING 2016 Organizing Committee.

Luo, W., Liu, F., Liu, Z., and Litman, D. (2016b). Automatic summarization of student course feedback. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 80–85, San Diego, California. Association for Computational Linguistics.

Malliaros, F. D. and Vazirgiannis, M. (2013). Clustering and community detection in directed networks: A survey. *CoRR*, abs/1308.0971.

Mani, I., Klein, G., House, D., Hirschman, L., Firmin, T., and Sundheim, B. (2002). SUMMAC: a text summarization evaluation. *Natural Language Engineering*, 8(01):43–68.

Martins, A. and Smith, N. A. (2009). Summarization with a joint model for sentence extraction and compression. In *Proceedings of the Workshop on Integer Linear Programming for NLP*, pages 1–9, Boulder, Colorado.

Mazumder, R., Hastie, T., and Tibshirani, R. (2010). Spectral regularization algorithms for learning large incomplete matrices. *Journal of Machine Learning Research.*

McKeown, K. R., Barzilay, R., Evans, D., Hatzivassiloglou, V., Klavans, J. L., Nenkova, A., Sable, C., Schiffman, B., and Sigelman, S. (2002). Tracking and summarizing news on a daily basis with Columbia's Newsblaster. In *Proceedings of the second international* *conference on Human Language Technology Research*, pages 280–285. Morgan Kaufmann Publishers Inc.

Medelyan, O., Frank, E., and Witten, I. H. (2009). Human-competitive tagging using automatic keyphrase extraction. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3*, EMNLP '09, pages 1318–1327, Stroudsburg, PA, USA. Association for Computational Linguistics.

Mehdad, Y., Carenini, G., Tompa, F., and T. NG, R. (2013). Abstractive meeting summarization with entailment and fusion. In *Proceedings of the 14th European Workshop on Natural Language Generation*, pages 136–146, Sofia, Bulgaria.

Menekse, M., Stump, G., Krause, S. J., and Chi, M. T. (2011). The effectiveness of students daily reflections on learning in engineering context. In *Proceedings of the American Society for Engineering Education (ASEE) Annual Conference*, Vancouver, Canada.

Miller, G. A. (1995). WordNet: a lexical database for English. *Communications of the ACM*, 38(11):39–41.

Mohammad, S., Dorr, B., Egan, M., Hassan, A., Muthukrishan, P., Qazvinian, V., Radev, D., and Zajic, D. (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.

Mosteller, F. (1989a). The 'muddiest point in the lecture' as a feedback device. *On Teaching and Learning: The Journal of the Harvard-Danforth Center*, 3:10–21.

Mosteller, F. (1989b). The 'muddiest point in the lecture' as a feedback device. *Teaching and Learning.*

Nallapati, R., Xiang, B., and Zhou, B. (2016). Sequence-to-sequence rnns for text summarization. *CoRR*, abs/1602.06023.

Nanba, H. and Okumura, M. (1999). Towards multi-paper summarization using reference information. In *IJCAI*, volume 99, pages 926–931.

Nenkova, A. and McKeown, K. (2011). Automatic summarization. *Foundations and Trends in Information Retrieval.*

Nenkova, A. and Passonneau, R. (2004). Evaluating content selection in summarization: The pyramid method. In Susan Dumais, D. M. and Roukos, S., editors, *HLT-NAACL 2004: Main Proceedings*, pages 145–152, Boston, Massachusetts, USA. Association for Computational Linguistics.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). BlEU: A method for automatic evaluation of machine translation. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA.

Qazvinian, V. and Radev, D. R. (2008). Scientific paper summarization using citation summary networks. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 689–696. Association for Computational Linguistics.

Qazvinian, V. and Radev, D. R. (2011). Learning from collective human behavior to introduce diversity in lexical choice. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1098–1108, Portland, Oregon, USA. Association for Computational Linguistics.

Radev, D., Otterbacher, J., Winkel, A., and Blair-Goldensohn, S. (2005). Newsinessence: summarizing online news topics. *Communications of the ACM*, 48(10):95–98.

Radev, D. R., Jing, H., Styś, M., and Tam, D. (2004). Centroid-based summarization of multiple documents. *Inf. Process. Manage.*, 40(6):919–938.

Ren, P., Wei, F., CHEN, Z., MA, J., and Zhou, M. (2016). A redundancy-aware sentence regression framework for extractive summarization. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 33–43, Osaka, Japan. The COLING 2016 Organizing Committee.

Rus, V., Lintean, M. C., Banjade, R., Niraula, N. B., and Stefanescu, D. (2013). SEMILAR: The semantic similarity toolkit. In *ACL (Conference System Demonstrations)*, pages 163–168.

Rush, A. M., Chopra, S., and Weston, J. (2015). A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, Portugal. Association for Computational Linguistics.

Schinas, M., Papadopoulos, S., Kompatsiaris, Y., and Mitkas, P. A. (2015). Visual event summarization on social media using topic modelling and graph-based ranking algorithms. In *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*, ICMR '15, pages 203–210, New York, NY, USA. ACM.

Shen, C., Liu, F., Weng, F., and Li, T. (2013). A participant-based approach for event summarization using Twitter streams. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1152–1162, Atlanta, Georgia. Association for Computational Linguistics.

Ştefănescu, D., Banjade, R., and Rus, V. (2014). Latent semantic analysis models on Wikipedia and TASA. In *The 9th Language Resources and Evaluation Conference (LREC 2014)*, pages 26–31, Reykjavik, Iceland.

Takase, S., Suzuki, J., Okazaki, N., Hirao, T., and Nagata, M. (2016). Neural headline generation on abstract meaning representation. In *Proceedings of the 2016 Conference* *on Empirical Methods in Natural Language Processing*, pages 1054–1059, Austin, Texas. Association for Computational Linguistics.

Teufel, S. and Moens, M. (2002). Summarizing scientific articles: experiments with relevance and rhetorical status. *Computational linguistics*, 28(4):409–445.

Teufel, S. and van Halteren, H. (2004). Evaluating information content by factoid analysis: Human annotation and stability. In Lin, D. and Wu, D., editors, *Proceedings of EMNLP 2004*, pages 419–426, Barcelona, Spain. Association for Computational Linguistics.

Thomaidou, S., Leymonis, K., and Vazirgiannis, M. (2013). GrammAds: Keyword and Ad creative generator for online advertising campaigns. In *Digital Enterprise Design and Management 2013*, pages 33–44. Springer.

Turpin, A., Tsegay, Y., Hawking, D., and Williams, H. E. (2007). Fast generation of result snippets in web search. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 127–134. ACM.

Ueda, Y., Oka, M., Koyama, T., and Miyauchi, T. (2000). Toward the "at-a-glance" summary: Phrase-representation summarization method. In *Proceedings of the 18th Conference on Computational Linguistics - Volume 2*, COLING '00, pages 878–884, Stroudsburg, PA, USA. Association for Computational Linguistics.

Van den Boom, G., Paas, F., Van Merrienboer, J. J., and Van Gog, T. (2004). Reflection prompts and tutor feedback in a web-based learning environment: effects on students' self-regulated learning competence. *Computers in Human Behavior*, 20(4):551 – 567.

Van Halteren, H. and Teufel, S. (2003). Examining the consensus between human summaries: initial experiments with factoid analysis. In *Proceedings of the HLT-NAACL 03 on Text summarization workshop-Volume 5*, pages 57–64. Association for Computational Linguistics.

Van Labeke, N., Whitelock, D., Field, D., Pulman, S., and Richardson, J. T. (2013). What is my essay really saying? Using extractive summarization to motivate reflection and redrafting. In *AIED Workshops*.

Vanderwende, L., Suzuki, H., Brockett, C., and Nenkova, A. (2007). Beyond SumBasic: Task-focused summarization with sentence simplification and lexical expansion. *Information Processing & Management*, 43(6):1606–1618.

Varadarajan, R. and Hristidis, V. (2006). A system for query-specific document summarization. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, pages 622–631. ACM.

Vuurens, J. B., de Vries, A. P., Blanco, R., and Mika, P. (2015). Online news tracking for ad-hoc information needs. In *Proceedings of the 2015 International Conference on The Theory of Information Retrieval*, ICTIR '15, pages 221–230, New York, NY, USA. ACM.

Wan, X. and Yang, J. (2008). Multi-document summarization using cluster-based link analysis. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '08, pages 299–306, New York, NY, USA. ACM.

Wang, L. and Ling, W. (2016). Neural network-based abstract generation for opinions and arguments. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 47–57, San Diego, California. Association for Computational Linguistics.

Wang, X., Nishino, M., Hirao, T., Sudoh, K., and Nagata, M. (2016). Exploring text links for coherent multi-document summarization. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 213–223, Osaka, Japan. The COLING 2016 Organizing Committee.

Woodsend, K. and Lapata, M. (2012). Multiple aspect summarization using integer linear programming. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 233–243, Jeju Island, Korea. Association for Computational Linguistics.

Wu, Y.-f. B., Li, Q., Bot, R. S., and Chen, X. (2005). Domain-specific keyphrase extraction. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, CIKM '05, pages 283–284, New York, NY, USA. ACM.

Xiong, W. (2015). *Helpfulness Guided Review Summarization*. PhD thesis, University of Pittsburgh.

Xiong, W. and Litman, D. (2014). Empirical analysis of exploiting review helpfulness for extractive summarization of online reviews. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1985–1995, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Yang, R., Bu, Z., and Xia, Z. (2012). Automatic summarization for chinese text using affinity propagation clustering and latent semantic analysis. In *Proceedings of the 2012 International Conference on Web Information Systems and Mining*, WISM'12, pages 543–550, Berlin, Heidelberg. Springer-Verlag.

Yatani, K., Novati, M., Trusty, A., and Truong, K. N. (2011). Review Spotlight: A user interface for summarizing user-generated reviews using adjective-noun word pairs. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 1541–1550, New York, NY, USA. ACM.

Zajic, D., Dorr, B. J., Lin, J., and Schwartz, R. (2007). Multi-candidate reduction: Sentence compression as a tool for document summarization tasks. *Information Processing and Management*.

Zhang, B., Li, H., Liu, Y., Ji, L., Xi, W., Fan, W., Chen, Z., and Ma, W.-Y. (2005). Improving web search results using affinity graph. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '05, pages 504–511, New York, NY, USA. ACM.

Zhang, Y., Zincir-Heywood, N., and Milios, E. (2004). World Wide Web site summarization. *Web Intelligence and Agent Systems*, 2(1):39–53.

Zhu, X., Goldberg, A., Van Gael, J., and Andrzejewski, D. (2007). Improving diversity in ranking using absorbing random walks. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics*, pages 97–104, Rochester, New York. Association for Computational Linguistics.

APPENDIX A

STUDENT RESPONSES TO MUDDIEST POINT

Prompt

Describe what was confusing or needed more detail.

Student Responses

S1: nothing
S2: Graphs of attraction/repulsive & interatomic separation
S3: Most of the lecture was muddy. I tried to follow along but I couldn't grasp the concepts. Plus it's hard to see what's written on the white board when the projector shines on it
S4: Energy vs. distance between atoms graph and what it tells us
S5: We jumped right into several slides with complicated graphs and undefined variables. I couldn't understand the trends
S6: I think graphs and equations are hard to understand because I'm unfamiliar with the terms and equations
S7: size of print and colors are hard to read on printout
S8: Equations with bond strength and Hooke's law
S9: 4: AXES on coefficient of thermal expansion graph
S10: 5:Hooke's law
S11: -You need a laser pointer,15S12: The activity (Part III)
S13: I didn't have any trouble with anything
S14: Stress + Strain
S15: What happens to atomic structure when heated
S16: I didn't fully understand the concept of thermal expansion
S17: Nothing
S18: The graphs of attraction and repulsion were confusing to me
S19: Property related to bond strength
S20: Elastic modulus
S21: Graphs are too small to look at specific detail
S22: van der waals
S23: Equations with stress

S24: Elastic modulus

S25: The activity was difficult to comprehend as the text fuzzing and difficult to read. The pictures are impossible to understand It's too small

S26: : I was unsure of how to determine the a values but a group member explained it more clearly.

S27: How to determine which metal has higher coefficient of thermal expansion

S28: The repulsive/ attraction charts

S29: The coefficient of thermal expansion relationship to bond strength

S30: Elastic modulus

S31: I found a little confusing properties related to bond strength

S32: The coefficient thermal expansion

S33: The worksheet we did in class

S34: What is the coeff of thermal expansion? The graphs were muddy but we better understood with the activity

S35: Graphs of attractive + repulsive forces

S36: The different graphs that look the same

S37: I struggled a little bit the elastic modulus

S38: Just thought the lecture helps out more than activities

S39: I didn't understand the attractive and repulsive force graphs from the third slide

S40: The working definition of elasticity is not very clear. I think I'm imagining resilience instead

S41: I would like to have learned more about how to calculate the bond strength analytically and how that calculation relates to the graph

S42: How to determine the answers to part III, in the activity

S43: Part III on worksheet in class, comparing metals. I was confused about why each metal was selected

S44: Not the least bit confusing. All information was understood as needed for the class

S45: Nothing confusing

S46: Nothing really. This class could perhaps move at a much faster rate

Human Summary

- Graphs of attraction/ repulsive & atomic separation [10]
- Properties and equations with bond strength [7]
- Coefficient of thermal expansion
- Activity part III [4]

Table A1: Example student responses to Muddiest Point.

APPENDIX B

PHRASE-BASED HIGHLIGHTING SUMMARIZATION ANNOTATION

Start Time: _________8:55am__________
In creating each summary you should keep in mind the following scenario for its use. Imagine you are a TA for this course, what do you want to present to the instructor after reading the students' responses?
Prompt1: "Describe what you found most interesting in today's class?"
Responses from students

student_id	sentence_id	responses
e0806	1	Guilt analogy
e7951	2	Error bounding is interesting and useful
e1520	3	the idea of c and finding that error looked great to me
e3572	4	nothing
e5865	5	the topic itself hypothesis testing
e1234	6	You stated that the concept of the error boundary is abstract however i got it very well
e1235	7	Examples
e4639	8	break for those who couldnt be able to be silent
e1352	9	deciding whether or not our guess is correct through probability calculations was interesting
e1107	10	The playing card example and the usage of the null and alternative hypothesis
e3141	11	critical value for hypothesis testing
e1100	12	determining the probability of the error while rejecting ho.
	13	because it was combining all the topics we have done
e3008	14	The process of hypothesis testing
e5658	15	Hypothesis testing
e1881	16	null and alternative hypotesis

e1494	17	Hypothesis testing
e1907	18	Hypothesis test
e6161	19	Examples made the subject clear
e1903	20	Determining the critical value for error
e6162	21	Good
e3451	22	h0 and h1
e8610	23	Defining h0 and h1
e3991	24	Error bound 'c' , which implicates our level of fail to reject.
e2909	25	hypothesis testing and the exam question with f distribution
e7677	26	the polio example is quite explanatory for the main idea
e2099	27	H1 and Ho conditionss
e4254	28	repeating everything
e0162	29	Rejecting Hzero
e1993	30	Your attitude is usually the most interesting part of the class:) i have never seen a that good teacher who watches the class and give a break when they need it .
e0387	31	guessing
e4916	32	hypothesis testing
e1958	33	multiple variable sampling
e1226	34	critical value for rejection
e3249	35	proven guilty analogy in hypothesis testing
e9731	36	decision mechanism and criteria of hypothesis testing
e2018	37	hypothesis testing, especially the phrase 'presumed innocent untip proven guilty'
e3345	38	if we cannot prove it is not true we cannot reject it is true
e2351	39	Baydogan finally check the students in the class.
	40	But i think it must be in every lecture even in the PS
e2509	41	Testing whether the information we have is true or not with hypothesis testing method was interesting
e1912	42	The analogy to innocent until proven guilty was really helpful.

Task1: Phrase Summarization. Create a summary using 5 phrases together with how many students semantically mentioned each phrase. You can use your own phrases.

Note, please also highlight the corresponding phrases in the student responses above which are semantically same to the summary phrases using the highlighted colors in the first row in the table below. The number of highlights for each phrase should match the number of students who semantically mentioned the phrase.

Rank	Phrases	# of students
1	Hypothesis testing (in general)	13
2	Error bounding	7
3	Guilt analogy helpful	5
4	Conditions for H1 and H0	5
5	Good use of examples	4

Finish Time: __________9:16am__________

Task2: Abstract Summarization. Given the students' responses, create a short summary using your own words (∼40 words, no specific format other than linear) of it.

%type your summary below

Most students found the hypothesis testing the most interesting example, along with error bounding and rejection of the null hypothesis. The students found the guilt analogy very helpful, as well as the examples used to introduce the main ideas.

Finish Time: _________9:17am__________

Task3: Extractive summary. Select five most representative sentences in order as the summary. (Use the sentence index number.)

Rank1: ____11_____

Rank2: ____2_____

Rank3: ____1_____

Rank4: ____22_____

Rank5: ____19_____

Finish Time: ________9:17am____________

You can take a break if you want.

Start Time: ___________9:17am_________

Prompt2: "Describe what was confusing or needed more detail?"
Responses from students

student_id	sentence_id	responses
e0806	1	Nothing
e7951	2	These topics are kind of abstract but it would be better if we solve a couple of examples in the ps
e1520	3	it was a clear lesson for me
e3572	4	nothing
e5865	5	type 1 error
e1234	6	I think the midterm questions solution was confusing
e1235	7	Nothing
e4639	8	setting what is h1 or h0
e1352	9	the reason why we compute the probability of getting a larger value than our observed value was a bit confusing (the suit of cards example)
e1107	10	At which probability do we say that an event is unlikely or likely (is a probability of 28% for X > 28 high or low?)?
	11	(Especially concerning the playing cards example)
e3141	12	nothing
e1100	13	the first thing of lecture f thing?
e3008	14	Everything is funny
e5658	15	None
e1881	16	hyposthesis testing applications
e1494	17	Type 1 error
e1907	18	M-8 m-25 problem
e6161	19	Nothing
e1903	20	The exam question, T square distribution
e6162	21	Nothibg
e3451	22	error in h0 and h1
e8610	23	Solution of the exam question
e3991	24	Today was complicated but I didnt have any muddiest points in the lecture.
e2909	25	probability of making mistake is a little confusing but i am sure i will understand it in the next lecture
e7677	26	I'm not sure about importance of this topic, but we discussed similar things again and again, even, it caused some confusion.
	27	One, but a good and detailed example may be more beneficial
e2099	28	Hiw we can decide h1 or H0
e4254	29	more examples about the testing

e0162	30	Why we have to reject hzero
e1993	31	This lesson was easy compared to the other ones but at the beginning we solved the question from the exam but i couldnt understand it
e0387	32	the critical value
e4916	33	type 1 error
e1958	34	how to compare two variances
e1226	35	i need more examples to truely understand hypothesis testing
e9731	37	solution of last example
e2018	38	probability of making a mistake, type1 error .
	39	the last example of the class
e3345	40	when exactly we accept the data as strong evidence, %5 rule ?
e2351	41	Baydogan quickly went over the course.
	42	I could not keep up with him
e2509	43	Choosing the critical probability is a little bit a relative subject
e1912	44	Need more clarification on the application of hypothesis testing

Task1: Phrase Summarization. Create a summary using 5 phrases together with how many students semantically mentioned each phrase. You can use your own phrases.

Note, please also highlight the corresponding phrases in the student responses above which are semantically same to the summary phrases using the highlighted colors in the first row in the table below. The number of highlights for each phrase should match the number of students who semantically mentioned the phrase.

Rank	Phrases	Student number
1	H0 vs H1	5
2	Type I error	5
3	Not enough hypothesis testing examples	4
4	Solution of exam question	3
5	Critical value	2

Finish Time: ________9:32am___________

Task2: Abstract Summarization. Given the students' responses, create a short summary using your own words (∼40 words, no specific format other than linear) of it.

%type your summary below

The majority of students that had trouble had confusing related to hypothesis testing, in particular H1 vs H0. A smaller proportion had trouble understanding type 1 error, critical value, and the solution to the exam question solved in class.

Finish Time: _______9:35am_____________

Task3: Extractive summary. Select five most representative sentences in order as the summary. (Use the sentence index number.)

Rank1: ____8_____
Rank2: ____17_____
Rank3: ____29_____
Rank4: ____31_____
Rank5: ____32_____
Finish Time: __________9:35am_________

APPENDIX C

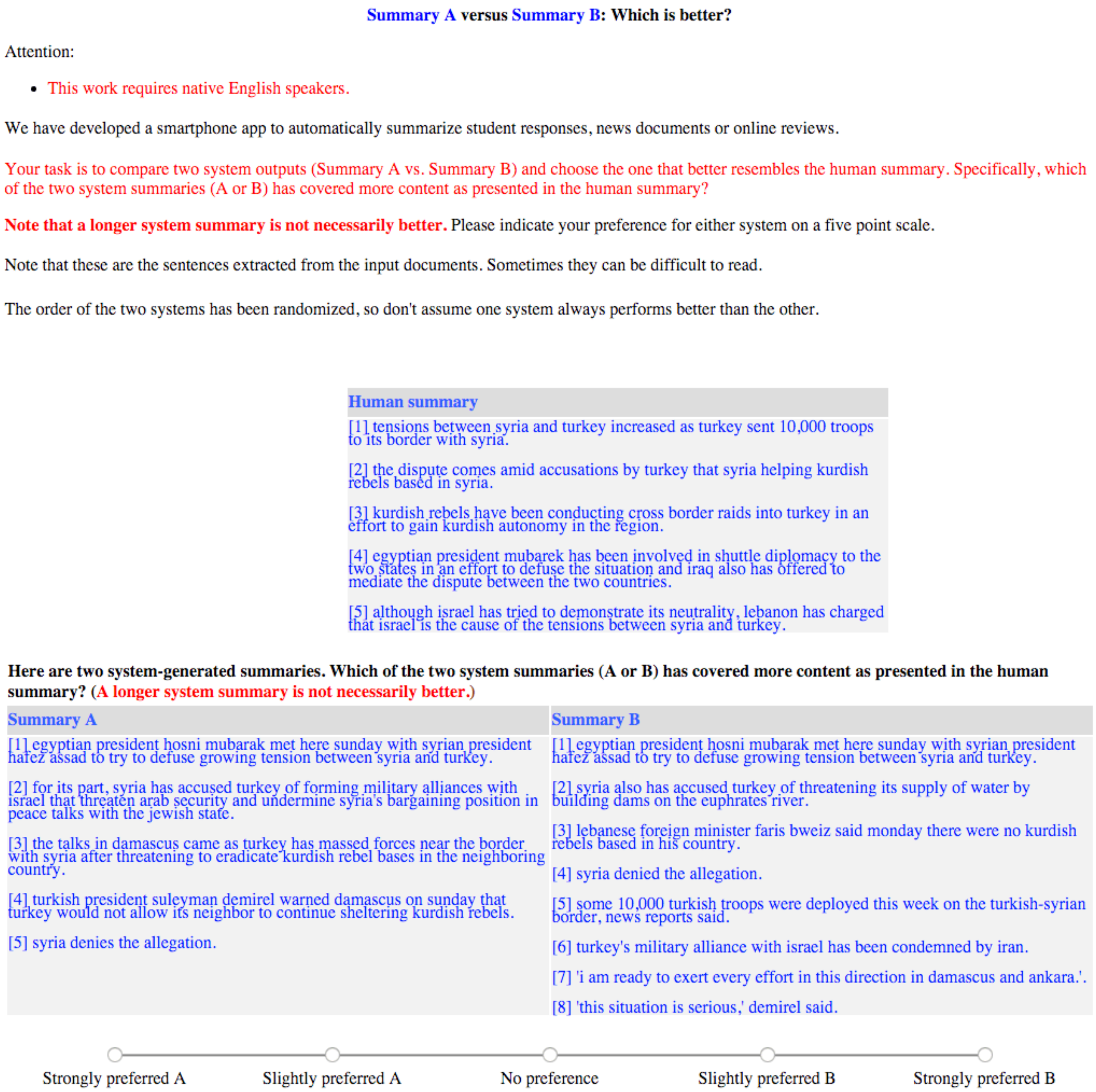
RESULTS WITHOUT REMOVING LOW-FREQUENCY BIGRAMS

		R-1			**R-2**		
	System	R	P	F	R	P	F
Eng	ILP	.351	.315	.295	.108	.106	**.098**
	ILP+MC	**.355**	**.322**	**.301**	**.111**	**.110**	**.098**
Stat2015	ILP	.401	.390	.386	.186	.173	.173
	ILP+MC	**.418**	.389	**.393**	**.210**$^+$	**.187**	**.191**
Stat2016	ILP	.470	.496	.479	.249	.265	.255
	ILP+MC	.423$^-$	.447$^-$	.432$^-$	.209$^-$	.222$^-$	.213$^-$
CS2016	ILP	.374	.394	.375	.138	.144	.138
	ILP+MC	**.380**	**.408**	**.383**	**.144**	**.149**	**.143**
camera	ILP	.456	.461	.458	.168	.168	.168
	ILP+MC	.440	.437	.438	.146	.145	.146
movie	ILP	.426	.422	.422	.109	.109	.109
	ILP+MC	**.430**	.414	.419	.102	.097	.099
peer	ILP	.470	.464	.465	.228	.217	.221
	ILP+MC	.452	.447	.448	.175	.172	.173
DUC04	ILP	.377	.381	.379	.092	.093	.092
	ILP+MC	.337$^-$	.349$^-$	.342$^-$	.071$^-$	.074$^-$	.072$^-$

Table C1: Summarization results without removing low-frequency bigrams. That is, all bigrams are used in the matrix approximation process. Compared to Table 4.4, by using the cutoff technique, both ILP and ILP+MC get better. In specific, 70 out of 96 ROUGE scores (8 data sets × 2 methods × (3 R-1 + 3 R-2)) are improved, 10 of them are even, and only 16 get worse.

APPENDIX D

EXAMPLE HITS FOR AMAZON MECHANICAL TURK



Summary A versus Summary B: Which is better?

Attention:

- This work requires native English speakers.

We have developed a smartphone app to automatically summarize student responses, news documents or online reviews.

Your task is to compare two system outputs (Summary A vs. Summary B) and choose the one that better resembles the human summary. Specifically, which of the two system summaries (A or B) has covered more content as presented in the human summary?

Note that a longer system summary is not necessarily better. Please indicate your preference for either system on a five point scale.

Note that these are the sentences extracted from the input documents. Sometimes they can be difficult to read.

The order of the two systems has been randomized, so don't assume one system always performs better than the other.

Human summary

[1] tensions between syria and turkey increased as turkey sent 10,000 troops to its border with syria.

[2] the dispute comes amid accusations by turkey that syria helping kurdish rebels based in syria.

[3] kurdish rebels have been conducting cross border raids into turkey in an effort to gain kurdish autonomy in the region.

[4] egyptian president mubarek has been involved in shuttle diplomacy to the two states in an effort to defuse the situation and iraq also has offered to mediate the dispute between the two countries.

[5] although israel has tried to demonstrate its neutrality, lebanon has charged that israel is the cause of the tensions between syria and turkey.

Here are two system-generated summaries. Which of the two system summaries (A or B) has covered more content as presented in the human summary? (A longer system summary is not necessarily better.)

Summary A

[1] egyptian president hosni mubarak met here sunday with syrian president hafez assad to try to defuse growing tension between syria and turkey.

[2] for its part, syria has accused turkey of forming military alliances with israel that threaten arab security and undermine syria's bargaining position in peace talks with the jewish state.

[3] the talks in damascus came as turkey has massed forces near the border with syria after threatening to eradicate kurdish rebel bases in the neighboring country.

[4] turkish president suleyman demirel warned damascus on sunday that turkey would not allow its neighbor to continue sheltering kurdish rebels.

[5] syria denies the allegation.

Summary B

[1] egyptian president hosni mubarak met here sunday with syrian president hafez assad to try to defuse growing tension between syria and turkey.

[2] syria also has accused turkey of threatening its supply of water by building dams on the euphrates river.

[3] lebanese foreign minister faris bweiz said monday there were no kurdish rebels based in his country.

[4] syria denied the allegation.

[5] some 10,000 turkish troops were deployed this week on the turkish-syrian border, news reports said.

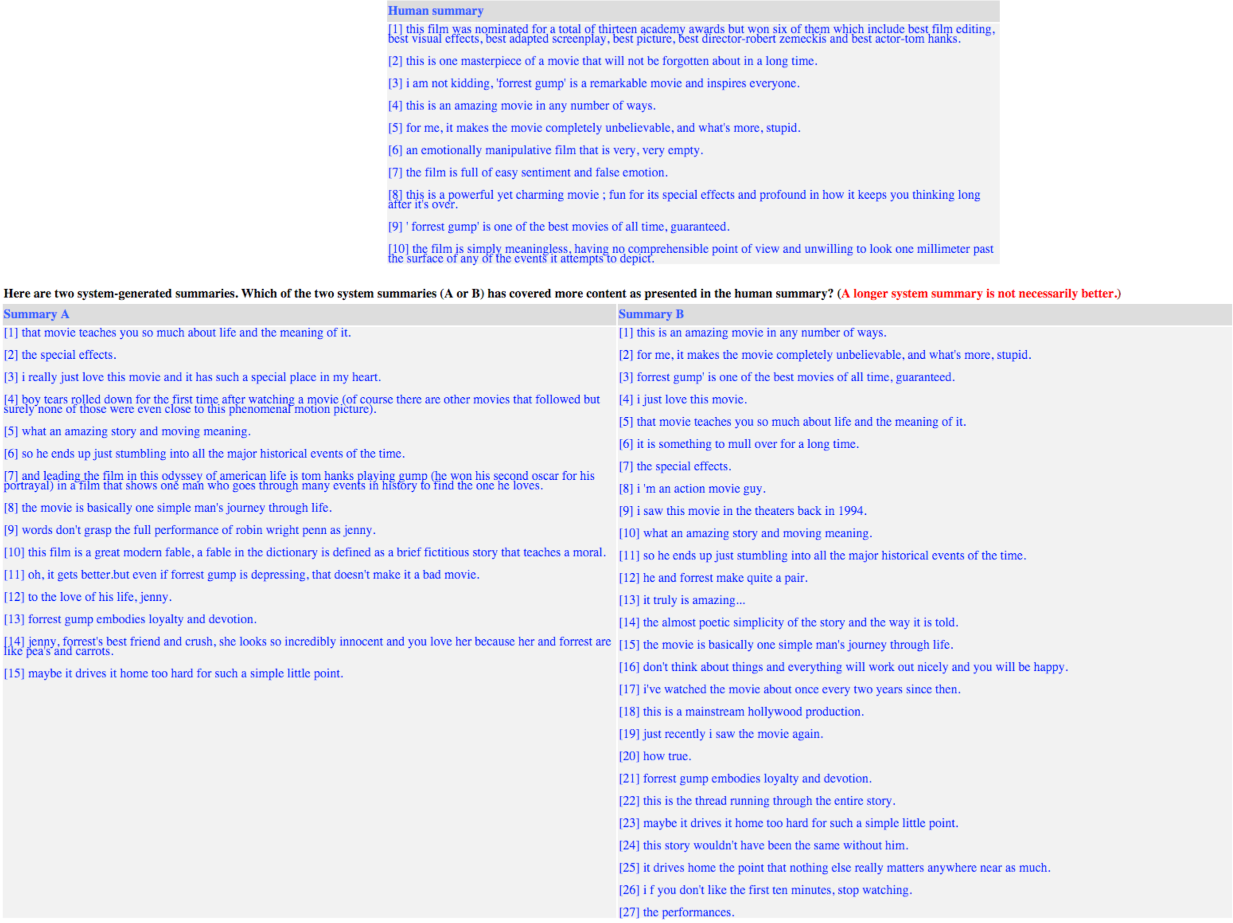
[6] turkey's military alliance with israel has been condemned by iran.

[7] 'i am ready to exert every effort in this direction in damascus and ankara.'.

[8] 'this situation is serious,' demirel said.

○ Strongly preferred A ○ Slightly preferred A ○ No preference ○ Slightly preferred B ○ Strongly preferred B

Figure D1: An example HIT from DUC04, System A is ILP and System B is SumBasic.



Human summary

[1] this film was nominated for a total of thirteen academy awards but won six of them which include best film editing, best visual effects, best adapted screenplay, best picture, best director-robert zemeckis and best actor-tom hanks.

[2] this is one masterpiece of a movie that will not be forgotten about in a long time.

[3] i am not kidding, 'forrest gump' is a remarkable movie and inspires everyone.

[4] this is an amazing movie in any number of ways.

[5] for me, it makes the movie completely unbelievable, and what's more, stupid.

[6] an emotionally manipulative film that is very, very empty.

[7] the film is full of easy sentiment and false emotion.

[8] this is a powerful yet charming movie ; fun for its special effects and profound in how it keeps you thinking long after it's over.

[9] ' forrest gump' is one of the best movies of all time, guaranteed.

[10] the film is simply meaningless, having no comprehensible point of view and unwilling to look one millimeter past the surface of any of the events it attempts to depict.

Here are two system-generated summaries. Which of the two system summaries (A or B) has covered more content as presented in the human summary? (A longer system summary is not necessarily better.)

Summary A

[1] that movie teaches you so much about life and the meaning of it.

[2] the special effects.

[3] i really just love this movie and it has such a special place in my heart.

[4] boy tears rolled down for the first time after watching a movie (of course there are other movies that followed but surely none of those were even close to this phenomenal motion picture).

[5] what an amazing story and moving meaning.

[6] so he ends up just stumbling into all the major historical events of the time.

[7] and leading the film in this odyssey of american life is tom hanks playing gump (he won his second oscar for his portrayal) in a film that shows one man who goes through many events in history to find the one he loves.

[8] the movie is basically one simple man's journey through life.

[9] words don't grasp the full performance of robin wright penn as jenny.

[10] this film is a great modern fable, a fable in the dictionary is defined as a brief fictitious story that teaches a moral.

[11] oh, it gets better.but even if forrest gump is depressing, that doesn't make it a bad movie.

[12] to the love of his life, jenny.

[13] forrest gump embodies loyalty and devotion.

[14] jenny, forrest's best friend and crush, she looks so incredibly innocent and you love her because her and forrest are like pea's and carrots.

[15] maybe it drives it home too hard for such a simple little point.

Summary B

[1] this is an amazing movie in any number of ways.

[2] for me, it makes the movie completely unbelievable, and what's more, stupid.

[3] forrest gump' is one of the best movies of all time, guaranteed.

[4] i just love this movie.

[5] that movie teaches you so much about life and the meaning of it.

[6] it is something to mull over for a long time.

[7] the special effects.

[8] i 'm an action movie guy.

[9] i saw this movie in the theaters back in 1994.

[10] what an amazing story and moving meaning.

[11] so he ends up just stumbling into all the major historical events of the time.

[12] he and forrest make quite a pair.

[13] it truly is amazing...

[14] the almost poetic simplicity of the story and the way it is told.

[15] the movie is basically one simple man's journey through life.

[16] don't think about things and everything will work out nicely and you will be happy.

[17] i've watched the movie about once every two years since then.

[18] this is a mainstream hollywood production.

[19] just recently i saw the movie again.

[20] how true.

[21] forrest gump embodies loyalty and devotion.

[22] this is the thread running through the entire story.

[23] maybe it drives it home too hard for such a simple little point.

[24] this story wouldn't have been the same without him.

[25] it drives home the point that nothing else really matters anywhere near as much.

[26] i f you don't like the first ten minutes, stop watching.

[27] the performances.

Figure D2: An example HIT from moive, System A is ILP and System B is ILP+MC.