Joint Optimization of User-desired Content in Multi-document Summaries by Learning from User Feedback

Avinesh P.V.S and **Christian M. Meyer**
Research Training Group AIPHES and UKP Lab
Computer Science Department, Technische Universität Darmstadt
www.aiphes.tu-darmstadt.de, www.ukp.tu-darmstadt.de

Abstract

In this paper, we propose an extractive multi-document summarization (MDS) system using joint optimization and active learning for content selection grounded in user feedback. Our method interactively obtains user feedback to gradually improve the results of a state-of-the-art integer linear programming (ILP) framework for MDS. Our methods complement fully automatic methods in producing high-quality summaries with a minimum number of iterations and feedbacks. We conduct multiple simulation-based experiments and analyze the effect of feedback-based concept selection in the ILP setup in order to maximize the user-desired content in the summary.

1 Introduction

The task of producing summaries from a cluster of multiple topic-related documents has gained much attention during the Document Understanding Conference[1] (DUC) and the Text Analysis Conference[2] (TAC) series. Despite a lot of research in this area, it is still a major challenge to automatically produce summaries that are on par with human-written ones. To a large extent, this is due to the complexity of the task: a good summary must include the most relevant information, omit redundancy and irrelevant information, satisfy a length constraint, and be cohesive and grammatical. But an even bigger challenge is the high degree of subjectivity in content selection, as it can be seen in the small overlap of what is considered important by different users. Optimizing a system towards one single best summary that fits all users, as it is assumed by current state-of-the-art systems, is highly impractical and diminishes the usefulness of a system for real-world use cases.

In this paper, we propose an interactive concept-based model to assist users in creating a personalized summary based on their feedback. Our model employs integer linear programming (ILP) to maximize user-desired content selection while using a minimum amount of user feedback and iterations. In addition to the joint optimization framework using ILP, we explore pool-based active learning to further reduce the required feedback. Although there have been previous attempts to assist users in single-document summarization, no existing work tackles the problem of multi-document summaries using optimization techniques for user feedback. Additionally, most existing systems produce only a single, globally optimal solution. Instead, we put the human in the loop and create a personalized summary that better captures the users' needs and their different notions of importance.

Need for personalization. Table 1 shows the ROUGE scores (Lin, 2004) of multiple existing summarization systems, namely TF*IDF (Luhn, 1958), LexRank (Erkan and Radev, 2004), TextRank (Mihalcea and Tarau, 2004), LSA (Gong and Liu, 2001), KL-Greedy (Haghighi and Vanderwende, 2009), provided by the sumy package[3] and ICSI[4] (Gillick and Favre, 2009; Boudin et al., 2015), a strong state-of-the-art approach (Hong et al., 2014) in comparison to the extractive upper bound on DUC'04 and DBS. DUC'04 is an English dataset of abstractive summaries from ho-

[1] http://duc.nist.gov/
[2] http://www.nist.gov/tac/
[3] https://github.com/miso-belica/sumy
[4] https://github.com/boudinfl/sume

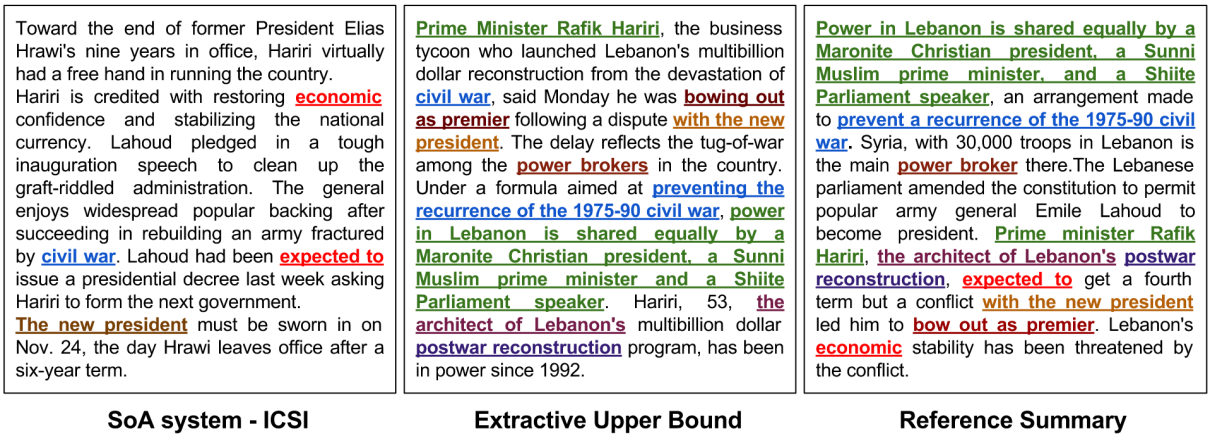

Figure 1: Lexical overlap of a reference summary (cluster D31043t in DUC 2004) with the summary produced by ICSI's state-of-the-art system (Boudin et al., 2015) and the extractive upper bound

	DUC'04			DBS		
Systems	R1	R2	SU4	R1	R2	SU4
TF*IDF	.292	.055	.086	.377	.144	.144
LexRank	.345	.070	.108	.434	.161	.180
TextRank	.306	.057	.096	.400	.167	.167
LSA	.294	.045	.081	.394	.154	.147
KL-Greedy	.336	.072	.104	.369	.133	.134
ICSI	.374	.090	.118	.452	.183	.190
UB	**.472**	**.210**	**.182**	**.848**	**.750**	**.532**

Table 1: ROUGE-1 (R1), ROUGE-2 (R2), and ROUGE-SU4 (SU4) scores of multiple systems compared to the extractive upper bound (UB)

mogenous news texts, whereas DBS (Benikova et al., 2016) is a German dataset of cohesive extracts from heterogeneous sources from the educational domain (see details in section 4.1). For each dataset, we compute an extractive upper bound (UB) by optimizing the sentence selection which maximizes ROUGE-2, i.e., the occurrence of bigrams as in the reference summary (Cao et al., 2016). Although some systems achieve state-of-the-art performance, their scores are still far from the extractive upper bound of individual reference summaries as shown in Figure 1. This is due to low inter-annotator agreement for concept selection: Zechner (2002) reports, for example, $\kappa = .13$ and Benikova et al. (2016) $\kappa = .23$. Most systems try to optimize for *all* reference summaries instead of personalizing, which we consider essential to capture user-desired content.

Need for user feedback. The goal of concept selection is finding the important information within a given set of source documents. Although existing summarization algorithms come up with a generic notion of importance, it is still far from the user-specific importance as shown in Figure 1. In contrast, humans can easily assess importance given a topic or a query. One way to achieve personalized summarization is thus by combining the advantages of both human feedback and the generic notion of importance built in a system. This allows users to interactively steer the summarization process and integrate their user-specific notion of importance.

Contributions. In this work, (1) we propose a novel ILP-based model using an interactive loop to create multi-document user-desired summaries, and (2) we develop models using pool-based active learning and joint optimization techniques to collect user feedback on identifying important concepts of a topic. In order to encourage the community to advance research and replicate our results, we provide our interactive summarizer implementation as open-source software.[5].

Our proposed method and our new interactive summarization framework can be used in multiple application scenarios: as an interactive annotation tool, which highlights important sentences for the annotators, as a journalistic writing aid that suggests important, user-adapted content from multiple source feeds (e.g., live blogs), and as a medical data analysis tool that suggests key information assisting a patient's personalized medical diagnosis.

The rest of the paper is structured as follows: In section 2, we discuss related work. Section 3

[5] https://github.com/UKPLab/acl2017-interactive_summarizer

introduces our computer-assisted summarization framework using the concept-based optimization. Section 4 describes our experiment data and setup. In section 5, we then discuss our results and analyze the performance of our models across different datasets. Finally, we conclude the paper in section 6 and discuss future work.

2 Related Work

Previous works related to our research address extractive summarization as a budgeted subset selection problem, computer-assisted approaches, and personalized summarization models.

Bugeted subset selection. Extractive summarization systems that compose a summary from a number of important sentences from the source documents are by far the most popular solution for MDS. This task can be modeled as a budgeted maximum coverage problem. Given a set of sentences in the document collection, the task is to maximize the coverage of the subset of sentences under a length constraint. The scoring function estimates the importance of the content units for a summary. Most previous works consider sentences as content units and try different scoring functions to optimize the summary.

One of the earliest systems by McDonald (2007) models a scoring function by simultaneously maximizing the relevance scores of the selected content units and minimizing their pairwise redundancy scores. They solve the global optimization problem using an ILP framework. Later, several state-of-the-art results employed an ILP to maximize the number of relevant concepts in the created summary: Gillick and Favre (2009) use an ILP with bigrams as concepts and hand-coded deletion rules for compression. Berg-Kirkpatrick et al. (2011) combine grammatical features relating to the parse tree and use a maximum-margin SVM trained on annotated gold-standard compressions. Woodsend and Lapata (2012) jointly optimize content selection and surface realization, Li et al. (2013) estimate the weights of the concepts using supervised methods, and Boudin et al. (2015) propose an approximation algorithm to achieve the optimal solution. Although these approaches achieve state-of-the-art performance, they produce only one globally optimal summary which is impractical for various users due to the subjectivity of the task. Therefore, we research interactive computer-assisted approaches in order to produce personalized summaries.

Computer-assisted summarization. The majority of the existing computer-assisted summarization tools (Craven, 2000; Narita et al., 2002; Orăsan et al., 2003; Orăsan and Hasler, 2006) present important elements of a document to the user. Creating a summary then requires the human to cut, paste, and reorganize the important elements in order to formulate a final text. The work by Orăsan and Hasler (2006) is closely related to ours, since they assist users in creating summaries for a source document based on the output of a given automatic summarization system. However, their system is neither interactive nor does it consider the user's feedback in any way. Instead, they suggest the output of the state-of-the-art (single-document) summarization method as a summary draft and ask the user to construct the summary without further interaction.

Personalized summarization. While most previous work focuses on generic summaries, there have been a few attempts to take a user's preferences into account. The study by Berkovsky et al. (2008) shows that users prefer personalized summaries that precisely reflect their interests. These interests are typically modeled with the help of a query (Park and An, 2010) or keyword annotations reflecting the user's opinions (Zhang et al., 2003).

In another strand of research, Díaz and Gervás (2007) create user models based on social tagging and Hu et al. (2012) rank sentences by combining informativeness scores with a user's interests based on fuzzy clustering of social tags. Extending the use of social content, another recent work showed how personalized review summaries (Poussevin et al., 2015) can be useful in recommender systems beyond rating predictions. Although these approaches show that personalized summaries are more useful than generic summaries, they do not attempt to iteratively refine a summary in an interactive user–system dialog.

3 Approach

The goal of our work is maximizing the user-desired content in a summary within a minimum number of iterations. To this end, we propose an interactive loop that alternates the automatic creation of a summary and the acquisition of user feedback to refine the next iteration's summary.

3.1 Summary Creation

Our starting point is the concept-based ILP summarization framework by Boudin et al. (2015). Let C be the set of concepts in a given set of source documents D, c_i the presence of the concept i in the resulting summary, w_i a concept's weight, ℓ_j the length of sentence j, s_j the presence of sentence j in the summary, and Occ_{ij} the occurrence of concept i in sentence j. Based on these definitions, we formulate the following ILP:

$$\max \quad \sum_i w_i c_i \tag{1}$$

$$\forall j. \quad \sum_j \ell_j s_j \leq L \tag{2}$$

$$\forall i, j. \quad \sum_j s_j Occ_{ij} \geq c_i \tag{3}$$

$$\forall i, j. \quad s_j Occ_{ij} \leq c_i \tag{4}$$

$$\forall i. \quad c_i \in \{0, 1\} \tag{5}$$

$$\forall j. \quad s_j \in \{0, 1\} \tag{6}$$

The objective function (1) maximizes the occurrence of concepts c_i in the summary based on their weights w_i. The constraint formalized in (2) ensures that the summary length is restricted to a maximum length L, (3) ensures the selection of all concepts in a sentence s_j if s_j has been selected for the summary. Constraint (4) ensures that a concept is only selected if it is present in at least one of the selected sentences.

The two key factors for the performance of this ILP are defining the concept set C and a method to estimate the weights $w_i \in W$. Previous works have used word bigrams as concepts (Gillick and Favre, 2009; Li et al., 2013; Boudin et al., 2015) and either use document frequency (i.e. the number of source documents containing the concept) as weights (Woodsend and Lapata, 2012; Gillick and Favre, 2009) or estimate them using a supervised regression model (Li et al., 2013). For our implementation, we likewise use bigrams as concepts and document frequency as weights, as Boudin et al. (2015) report good results with this simple strategy. Our approach is, however, not limited to this setup, as our interactive approach allows for any definition of C and W, including potentially more sophisticated weight estimation methods, e.g., based on deep neural networks. In section 5.2, we additionally analyze how other notions of concepts can be integrated into our approach.

3.2 Interactive Summarization Loop

Algorithm 1 provides an overview of our interactive summarization approach. The system takes the set of source documents D as input, derives the set of concepts C, and initializes their weights W. In line 5, we start the interactive feedback loop iterating over $t = 0, \ldots, T$. We first create a summary S_t (line 6) by solving the ILP and then extract a set of concepts Q_t (line 7), for which we query the user in line 11 As the user feedback in the current time step, we use the concepts $I_t \subseteq Q_t$ that have been considered important by the user. For updating the weights W in line 12, we may use all feedback collected until the current time step t, i.e., $I_0^t = \bigcup_{j=0}^{t} I_j$ and the set of concepts $Q_0^t = \bigcup_{j=0}^{t} Q_j$ seen by the user (with $Q_0^{-1} = \emptyset$). If there are no more concepts to query (i.e., $Q_t = \emptyset$), we stop the iteration and return the personalized summary S_t.

Algorithm 1 Interactive summarizer

```
1:  procedure INTERACTIVESUMMARIZER()
2:      input: Documents D
3:      C ← extractConcepts(D)
4:      W ← conceptWeights(C)
5:      for t = 0...T do
6:          S_t ← getSummary(C, W)
7:          Q_t ← extractConcepts(S_t) − Q_0^{t−1}
8:          if Q_t = ∅ then
9:              return S_t
10:         else
11:             I_t ← obtainFeedback(S_t, Q_t)
12:             W ← updateWeights(W, I_0^t, Q_0^t)
13:         end if
14:     end for
15: end procedure
```

3.3 User Feedback Optimization

To optimize the summary creation based on user feedback, we iteratively change the concept weights in the objective function of the ILP setup. We define the following models:

Accept model (ACCEPT). This model presents the current summary S_t with highlighted concepts Q_t to a user and asks him/her to select all important concepts I_t. We assign the maximum weight MAX to all concepts in I_t and consider the remaining $Q_t - I_t$ as unimportant by setting their weight to 0 (see equation 7 and 8). The intuition

behind this baseline is that the modified scores cause the ILP to prefer the user-desired concepts while avoiding unimportant ones.

$$\forall i \in I_0^t. \quad w_i = MAX \tag{7}$$
$$\forall i \in Q_0^t - I_0^t. \quad w_i = 0 \tag{8}$$

Joint ILP with User Feedback (JOINT). The ACCEPT model fails in cases where the user could not accept concepts that never appear in one of the S_t summaries. To tackle this, in our JOINT model, we change the objective function of the ILP in order to create S_t by jointly optimizing importance and user feedback. We thus replace the equation (1) with:

$$\max \begin{cases} \sum_{i \notin Q_0^t} w_i c_i - \sum_{i \in Q_0^t} w_i c_i & \text{if } t \leq \tau \\ \sum_i w_i c_i & \text{if } t > \tau \end{cases} \tag{9}$$

Equation (9) maximizes the use of concepts for which we yet lack feedback ($i \notin Q_0^t$) and minimizes the use of concepts for which we already have feedback ($i \in Q_0^t$). In this JOINT model, we use an exploration phase $t = 0 \ldots \tau$ to collect the feedback, which terminates when the user does not return any important concepts (i.e., $I_t = \emptyset$). In the exploratory phase, the minus term in the equation 9 helps to reduce the score of the sentences whose concepts have received feedback already. In other words, it causes higher scores for sentences consisting of concepts which yet lack feedback. After the exploration step, we fall back to the original importance-based optimization function from equation (1).

Active learning with uncertainty sampling (AL). Our JOINT model explores well in terms of prioritizing the concepts which yet lack user feedback. However, it gives equal probabilities to all the unseen concepts. The AL model employs pool-based active learning (Kremer et al., 2014) during the exploration phase in order to prioritize concepts for which the model is most uncertain. We distinguish the unlabeled concept pool $C_u = \{\Phi(\tilde{x}_1), \Phi(\tilde{x}_2), ..., \Phi(\tilde{x}_N)\}$ and the labeled concept pool $C_\ell = \{(\Phi(x_1), y_1), (\Phi(x_2), y_2), \ldots, (\Phi(x_N), y_N)\}$, where each concept x_i is represented as a d-dimensional feature vector $\Phi(x_i) \in \mathbb{R}^d$. The labels $y_i \in \{-1, 1\}$ are 1 for all important concepts in I_0^t and -1 for all unimportant concepts in $Q_0^t - I_0^t$. Initially, the labeled concept pool C_ℓ is small or empty, whereas the unlabeled concept pool C_u is relatively large.

The learning algorithm is presented with a $C = C_\ell \cup C_u$ and is first called to learn a decision function $f^{(0)}\colon \mathbb{R}^d \rightarrow \mathbb{R}$, where the function $f^{(0)}(\Phi(\tilde{x}))$ is taken to predict the label of the input vector $\Phi(\tilde{x})$. Then, in each t^{th} iteration, where $t = 1, 2, \ldots, \tau$, the querying algorithm selects an instance of $\tilde{x}_t \in C_u$ for which the learning algorithm is least certain. Thus, our learning goal of active learning is to minimize the expected loss $\mathcal{L}$ (i.e., hinge loss) with limited querying opportunities to obtain a decision function $f^{(1)}, f^{(2)}, \ldots, f^{(\tau)}$ that can achieve low error rates:

$$\min \mathbb{E}_{(\Phi(x),y) \in C_\ell} \Big[\mathcal{L}(f^{(t)}(\Phi(x)), y) \Big] \tag{10}$$

As the learning algorithm, we use a support vector machine (SVM) with a linear kernel. To obtain the probability distribution over classes we use Platt's calibration (Platt, 1999), an effective approach for transforming classification models into a probability distribution. Equation (11) shows the probability estimates for $f^{(t)}$, where $f^{(t)}$ is the uncalibrated output of the SVM in the t^{th} iteration and A, B are scalar parameters that are learned by the calibration algorithm. The uncertainty scores are calculated as described in the equation (12) for all the concepts which lack feedback (C_u).

$$p(y \mid f^{(t)}) = \frac{1}{1 + \exp(A f^{(t)} + B)} \tag{11}$$
$$u_i = 1 - \max_{y \in \{-1,1\}} p(y \mid f^{(t)}) \tag{12}$$

For our AL model, we now change the objective function in order to create S_t by multiplying uncertainty scores u_i to the weights w_i. We thus replace the objective function from (9) with

$$\max \begin{cases} \sum_{i \notin Q_0^t} u_i w_i c_i & \text{if } t \leq \tau \\ \sum_i w_i c_i & \text{if } t > \tau \end{cases} \tag{13}$$

Active learning with positive sampling (AL+). One way to sample the unseen concepts is using uncertainty as in AL, but another way is to actively choose samples for which the learning algorithm predicts as a possible important concept. In AL+, we introduce the notion of certainty $(1 - u_i)$ for the positively predicted samples ($f^{(t)}(\Phi(\tilde{x}_i)) = 1$) in

Dataset	Lang	Topics	Summary type	Length
DBS	de	10	Coherent extracts	≈500 words
DUC'01	en	30	Abstracts	100 words
DUC'02	en	59	Abstracts	100 words
DUC'04	en	50	Abstracts	100 words

Table 2: Statistics of the MDS datasets used

the objective function (1) for producing S_t

$$\max \begin{cases} \sum_{i \notin Q_0^t} (1 - u_i) \ell_i w_i c_i & \text{if } t \leq \tau \\ \sum_i w_i c_i & \text{if } t > \tau \end{cases} \quad (14)$$

$$\text{where } \ell_i = \begin{cases} 0 & \text{if } f^{(t)}(\Phi(\tilde{x}_i)) = -1 \\ 1 & \text{if } f^{(t)}(\Phi(\tilde{x}_i)) = 1 \end{cases} \quad (15)$$

4 Experimental Setup

4.1 Data

For our experiments, we mainly focus on the DBS corpus, which is an MDS dataset of coherent extracts created from heterogeneous sources about multiple educational topics (Benikova et al., 2016). This corpus is well-suited for our evaluation setup, since we are able to easily simulate a user's feedback based on the overlap between generated and reference summary.

Additionally, we carry out experiments on the most commonly used evaluation corpora published by DUC/NIST from the generic multi-document summarization task carried out in DUC'01, DUC'02 and DUC'04. The documents are all from the news domain and are grouped into various topic clusters. Table 2 shows the properties of these corpora.

For evaluating the summaries against the reference summary we use ROUGE (Lin, 2004) with the parameters suggested by (Owczarzak et al., 2012) yielding high correlation with human judgments (i.e., with stemming and without stopword removal).[6] Since DBS summaries do not have a fixed length, we use a variable length parameter L for evaluation, where L denotes the length of the reference summary. All results are averaged across all topics and reference summaries.

4.2 Data Pre-processing and Features

To pre-process the datasets, we perform tokenization and stemming with NLTK (Loper and Bird, 2002) and constituency parsing with the Stanford parser (Klein and Manning, 2003) for English and German. The parse trees will be used in section 5.2 below to experiment with a syntactically motivated concept notion.

As a concept's feature representation Φ for our active learning setups AL and AL+, we use pre-trained word embeddings. We use the Google News embeddings with 300 dimensions by Mikolov et al. (2013) for English and the 100-dimensional news- and Wikipedia-based embeddings by Reimers et al. (2014) for German. Additionally, we add TF*IDF, number of stop words, presence of named entities, and word capitalization as features. Discrete features, such as part-of-speech tags, are mapped into the word representation via lookup tables.

4.3 Oracle-Based Simulation of User Feedback

The presence of a human in the loop typically demands for a user study based evaluation, but to collect sufficient data for various settings of our models would be too expensive. Therefore, we resort to an oracle-based approach, where the oracle is a system simulating the user by generating the feedback based on reference outputs. This idea has been widely used in the development of interactive systems (González-Rubio et al., 2012; Knowles and Koehn, 2016) for studying the problem and exhibiting solutions in a theoretical and controlled environment.

To simulate user feedback in our setting, we consider all concepts $I_t \subseteq Q_t$ from the system-suggested summary S_t as important if they are present in the reference summary. Let Ref be the set of concepts in the reference summary. In the t^{th} iteration, we return $I_t = Q_t \cap Ref$ as the simulated user feedback. Thus, the goal of our system is to reach the upper bound for a user's reference summary within a minimal number of iterations.

We limit our experiments to ten iterations, since it appears unrealistic that users are willing to participate in more feedback cycles. Petrie and Bevan (2009) even report only three to five iterations.

5 Results and Analysis

5.1 Methods

Table 3 shows the evaluation results of our four models. When evaluating a summarization system, it is common to report the mean ROUGE scores across clusters using all the reference summaries. However, since we aim at personalizing

[6] -n 4 -m -a -x -c 95 -r 1000 -f A -p 0.5 -t 0 -2 -4 -u

Datasets	ICSI			UB			ACCEPT			JOINT			AL			AL+		
	R1	R2	SU4	R1	R2	SU4	R1	R2	SU4	R1	R2	SU4	R1	R2	SU4	R1	R2	SU4
Concept Notion: Bigrams																		
DBS	.451	.183	.190	.848	.750	.532	.778	.654	.453	.815	.707	.484	**.833**	**.729**	.498	.828	.721	**.500**
DUC'04	.374	.090	.118	.470	.212	.185	.442	.176	.165	**.444**	**.180**	**.166**	.440	.178	.160	.427	.166	.154
DUC'02	.350	.085	.110	.474	.216	.187	.439	.178	.161	.444	.182	.165	**.448**	**.188**	.165	**.448**	.184	**.170**
DUC'01	.333	.073	.105	.450	.213	.181	.414	.171	.156	.418	.167	.149	**.435**	**.186**	**.163**	.426	.181	.158
Concept Notion: Content Phrases																		
DBS	.403	.135	.154	.848	.750	.532	.691	.531	.430	.742	.597	.419	**.776**	**.652**	**.448**	.767	.629	.440
DUC'04	.374	.090	.118	.470	.212	.185	.441	.176	.160	.441	.179	.162	**.444**	**.180**	**.162**	.422	.164	.150
DUC'02	.350	.085	.110	.474	.216	.187	.436	.181	.162	.444	.183	.165	**.446**	**.185**	**.168**	.442	.182	.162
DUC'01	.333	.073	.105	.450	.213	.181	.410	.165	.153	.417	.170	.156	**.433**	**.182**	**.161**	.420	.179	.154

Table 3: ROUGE-1 (R1), ROUGE-2 (R2) and ROUGE SU-4 (SU4) achieved by our models after the tenth iteration of the interactive loop in comparison to the upper bound and the basic ILP setup

Datasets	ACCEPT #F	JOINT #F	AL #F	AL+ #F
Concept Notion: Bigrams				
DBS	313	296	348	342
DUC'04	15	14	16	14
DUC'02	14	13	15	15
DUC'01	13	11	13	13
Concept Notion: Content Phrases				
DBS	110	114	133	145
DUC'04	8	9	10	10
DUC'02	7	7	8	6
DUC'01	7	7	8	6

Table 4: Average amount of user feedback (#F) considered by our models at the end of the tenth iteration of the interactive summarization loop

the summary for an individual user, we evaluate our models based on the mean ROUGE scores across clusters per reference summary. In Table 4, we additionally evaluate the models based on the amount of feedback (#F $= |I_0^T|$) taken by the oracles to converge to the upper bound within ten iterations.

To examine the system performance based on user feedback, we analyze our models' performance on multiple datasets. The results in Table 3 show that our idea of interactive multi-document summarization allows users to steer a general summary towards a personalized summary consistently across all datasets. From the results, we can see that the AL model starts from the concept-based ILP summarization and nearly reaches the upper bound for all the datasets within ten iterations. AL+ performs similar to AL in terms of ROUGE, but requires less feedback (compare Table 4). Furthermore, the ACCEPT and JOINT models get stuck in a local optimum due to the less exploratory nature of the models.

5.2 Concept Notion

Our interactive summarization approach is based on the scalable global concept-based model which uses bigrams as concepts. Thus, it is intuitive to use bigrams for collecting user feedback as well.[7] Although our models reach the upper bound when using bigram-based feedback, they require a significantly large number of iterations and much feedback to converge, as shown in Table 4.

To reduce the amount of feedback, we also consider content phrases to collect feedback. That is, syntactic chunks from the constituency parse trees consisting of non-function words (i.e., nouns, verbs, adjectives, and adverbs). For DBS being extractive dataset, we use bigrams and content phrases as concepts, both for the objective function in equation (1) and as feedback items, whereas for the DUC datasets, the concepts are always bigrams for both the feedback types (bigrams/content phrases). For DUC being abstractive, in the case of feedback given on content phrases, they are projected back to the bigrams to change the concept weights in order to have more overlap of simulated feedback. Table 4 shows feedbacks based on the content phrases reduces the number of feedbacks by a factor of 2. Furthermore, when content phrases are used as concepts for DBS, the performance of the models is lower compared to bigrams, as seen in Table 3.

5.3 Datasets

Figure 2 compares the ROUGE-2 scores and the amount of feedback used over time when applied to the DBS and the DUC'04 corpus. We can see from the figure that all models show an improvement of +.45 ROUGE-2 after merely 4 iterations

[7] We prune bigrams consisting of only functional words.

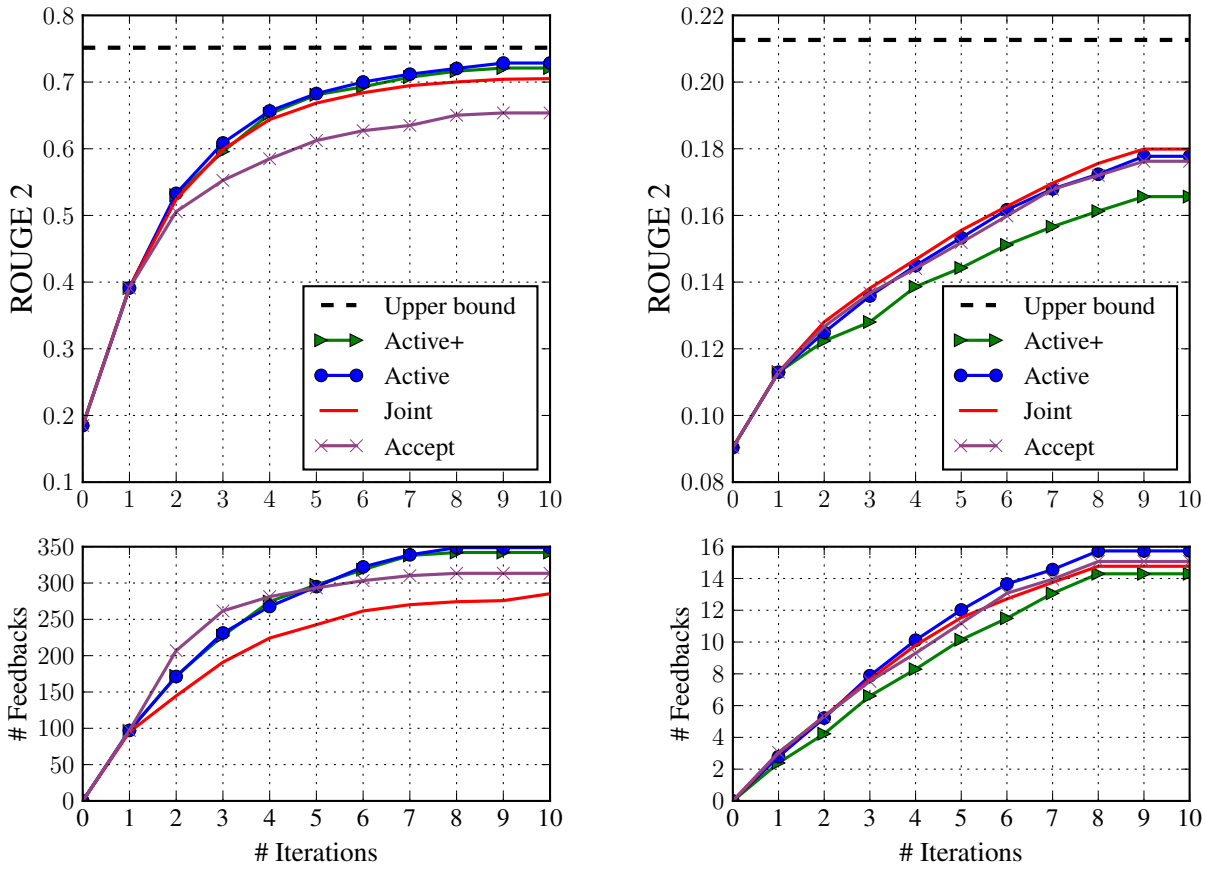

Figure 2: Analysis for the models over the DBS (left) and DUC'04 (right) datasets

on DBS. For DUC'04, the improvements are +.1 ROUGE-2 after ten iterations, which is relatively notable considering the lower upper bound of .21 ROUGE-2. This is primarily because DBS is a corpus of cohesive extracts, whereas DUC'04 consists of abstractive summaries. As a result, the oracles created using abstractive reference summaries have lower overlap of concepts as compared to that of the oracles created using extractive summaries.

For DBS, it becomes clear that the JOINT model converges faster with an optimum amount of feedback as compared to other models. ACCEPT takes relatively more feedbacks than JOINT, but performs low in terms of ROUGE scores. The best performing models are AL and AL+, which reach closest to the upper bound. This is clearly due to the exploratory nature of the models which use semantic representations of the concepts to predict uncertainty and importance of possible concepts for user feedback.

For DUC'04, the JOINT model reaches the closest to the upper bound, closely followed by AL. The JOINT model consistently stays above all other models and it gathers more important concepts due to optimizing feedbacks for concepts which lack feedback. Interestingly, AL+ performs rather worse in terms of both ROUGE scores and gathering important concepts. The primary reason for this is the fewer feedback collected from the simulation due to the abstractive property of reference summaries, which makes the AL+ model's prediction inconsistent.

5.4 Personalization

Figure 3 shows the performance of different models in comparison to two different oracles for the same document cluster. For DBS, the JOINT, AL, and AL+ models consistently converge to the upper bound in 4 iterations for different oracles, whereas ACCEPT takes longer for one oracle and does not reach the upper bound for the other.

For DUC'04, JOINT and AL show consistent performance across the oracles, whereas AL+ performs worse than the state-of-the-art system (iteration 0) for oracle created using abstractive summaries as shown in Figure 3 (right) for User:1.

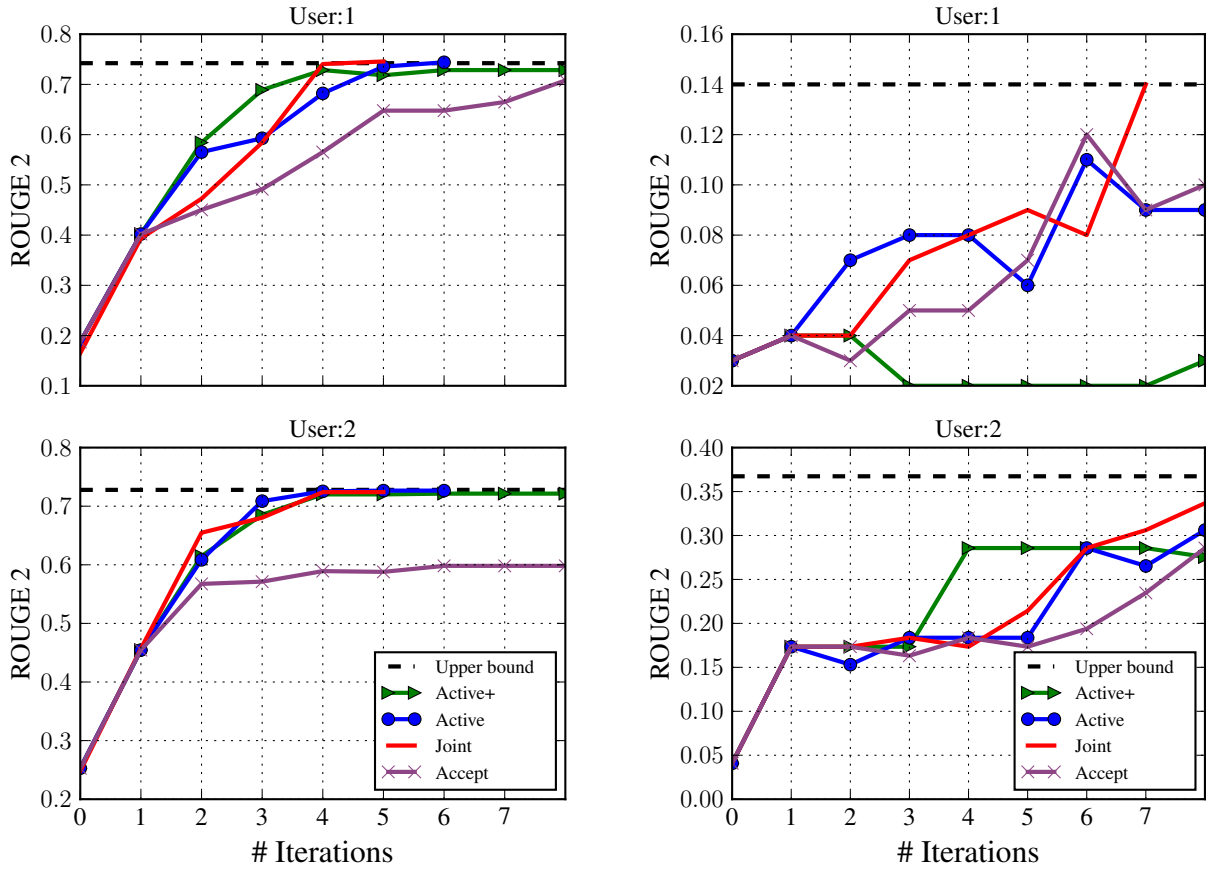

Figure 3: Analysis of models over cluster 7 from DBS (left) and cluster d30051t from DUC'04 (right) respectively for different oracles

However, for User:2, we observe a ROUGE-2 improvement of +.1 indicating that the predictions of the active learning system are better if there is more feedback. Nevertheless, we expect that in practical use, the human summarizers may give more feedback similar to DBS in comparison to DUC'04 simulation setting.

6 Conclusion and Future Work

We propose a novel ILP-based approach using interactive user feedback to create multi-document user-desired summaries. In this paper, we investigate pool-based active learning and joint optimization techniques to collect user feedback for identifying important concepts for a summary. Our models show that interactively collecting feedback consistently steers a general summary towards a user-desired personalized summary. We empirically checked the validity of our approach on standard datasets using simulated user feedback and observed that our framework shows promising results in terms of producing personalized multi-document summaries.

As future work, we plan to investigate more sophisticated sampling strategies based on active learning and concept graphs to incorporate lexical-semantic information for concept selection. We also plan to look into ways to propagate feedback to similar and related concepts with partial feedback, to reduce the total amount of feedback. This is a promising direction as we have shown that interactive methods help to create user-desired personalized summaries, and with minimum amount of feedbacks, it has propitious use in scenarios where user-adapted content is a requirement.

Acknowledgments

This work has been supported by the German Research Foundation as part of the Research Training Group Adaptive Preparation of Information from Heterogeneous Sources (AIPHES) under grant No. GRK 1994/1. We also acknowledge the useful comments and suggestions of the anonymous reviewers.

References

Darina Benikova, Margot Mieskes, Christian M. Meyer, and Iryna Gurevych. 2016. Bridging the gap between extractive and abstractive summaries: Creation and evaluation of coherent extracts from heterogeneous sources. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING)*. Osaka, Japan, pages 1039–1050. http://aclweb.org/anthology/C16-1099.

Taylor Berg-Kirkpatrick, Dan Gillick, and Dan Klein. 2011. Jointly learning to extract and compress. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL/HLT)*. Portland, OR, USA, pages 481–490. http://aclweb.org/anthology/P11-1049.

Shlomo Berkovsky, Timothy Baldwin, and Ingrid Zukerman. 2008. Aspect-based personalized text summarization. In *Adaptive Hypermedia and Adaptive Web-Based Systems. Proceedings of the 5th International Conference*, Springer, Berlin/Heidelberg, volume 5149 of *Lecture Notes in Computer Science*, pages 267–270. https://doi.org/10.1007/978-3-540-70987-9_31.

Florian Boudin, Hugo Mougard, and Benoit Favre. 2015. Concept-based summarization using integer linear programming: From concept pruning to multiple optimal solutions. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMLNP)*. Lisbon, Portugal, pages 1914–1918. http://aclweb.org/anthology/D15-1220.

Ziqiang Cao, Chengyao Chen, Wenjie Li, Sujian Li, Furu Wei, and Ming Zhou. 2016. TGSum: Build Tweet Guided Multi-Document Summarization Dataset. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI)*. Phoenix, AZ, USA, pages 2906–2912. http://www.aaai.org/ocs/index.php/AAAI/AAAI16.

T. C. Craven. 2000. Abstracts produced using computer assistance. *Journal of the American Society for Information Science* 51(8):745–756. https://doi.org/10.1002/(SICI)1097-4571(2000)51:8<745::AID-ASI70>3.0.CO;2-Z.

Alberto Díaz and Pablo Gervás. 2007. User-model based personalized summarization. *Information Process Management* 43(6):1715–1734. https://doi.org/10.1016/j.ipm.2007.01.009.

Günes Erkan and Dragomir R. Radev. 2004. LexRank: Graph-based Lexical Centrality As Salience in Text Summarization. *Journal of Artificial Intelligence Research* 22(1):457–479. https://www.jair.org/papers/paper1523.html.

Dan Gillick and Benoit Favre. 2009. A scalable global model for summarization. In *Proceedings of the Workshop on Integer Linear Programming for Natural Langauge Processing*. Boulder, CO, USA, pages 10–18. http://aclweb.org/anthology/W09-1802.

Yihong Gong and Xin Liu. 2001. Generic text summarization using relevance measure and latent semantic analysis. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. New Orleans, LA, USA, pages 19–25. https://doi.org/10.1145/383952.383955.

Jesús González-Rubio, Daniel Ortiz-Martínez, and Francisco Casacuberta. 2012. Active learning for interactive machine translation. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*. Avignon, France, pages 245–254. http://aclweb.org/anthology/E12-1025.

Aria Haghighi and Lucy Vanderwende. 2009. Exploring content models for multi-document summarization. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*. Boulder, CO, USA, pages 362–370. http://aclweb.org/anthology/N09-1041.

Kai Hong, John M. Conroy, Benoît Favre, Alex Kulesza, Hui Lin, and Ani Nenkova. 2014. A repository of state of the art and competitive baseline summaries for generic news summarization. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC)*. Reykjavik, Iceland, pages 1608–1616. http://www.lrec-conf.org/proceedings/lrec2014/summaries/1093.html.

Po Hu, Donghong Ji, Chong Teng, and Yujing Guo. 2012. Context-enhanced personalized social summarization. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING)*. Mumbai, India, pages 1223–1238. http://www.aclweb.org/anthology/C12-1075.

Dan Klein and Christopher D. Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics (ACL)*. Sapporo, Japan, pages 423–430. https://doi.org/10.3115/1075096.1075150.

Rebecca Knowles and Philipp Koehn. 2016. Neural interactive translation prediction. In *Proceedings of the Conference of the Association for Machine Translation in the Americas (AMTA)*.

Jan Kremer, Kim Steenstrup Pedersen, and Christian Igel. 2014. Active learning with support vector machines. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 4(4):313–326. https://doi.org/10.1002/widm.1132.

Chen Li, Xian Qian, and Yang Liu. 2013. 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)*. Sofia, Bulgaria, pages 1004–1013. http://aclweb.org/anthology/P13-1099.

Chin-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*. Barcelona, Spain, pages 74–81. http://aclweb.org/anthology/W04-1013.

Edward Loper and Steven Bird. 2002. NLTK: The Natural Language Toolkit. In *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics*. pages 63–70. https://doi.org/10.3115/1118108.1118117.

H. P. Luhn. 1958. The automatic creation of literature abstracts. *IBM Journal of Research and Development* 2(2):159–165. https://doi.org/10.1147/rd.22.0159.

Ryan McDonald. 2007. A study of global inference algorithms in multi-document summarization. In *Advances in Information Retrieval. Proceedings of the 29th European Conference on IR Research (ECIR)*, Springer, Berlin/Heidelberg, volume 4425 of *Lecture Notes in Computer Science*, pages 557–564. https://doi.org/10.1007/978-3-540-71496-5_51.

Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing Order into Text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Barcelona, Spain, pages 404–411. http://aclweb.org/anthology/W04-3252.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *CoRR* abs/1301.3781. http://arxiv.org/abs/1301.3781.

Masumi Narita, Kazuya Kurokawa, and Takehito Utsuro. 2002. A Web-based English Abstract Writing Tool Using a Tagged E–J Parallel Corpus. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC)*. Las Palmas, Spain. http://www.lrec-conf.org/proceedings/lrec2002/sumarios/137.htm.

Constantin Orăsan and Laura Hasler. 2006. Computer-aided Summarisation: What the User Really Wants. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC)*. Genoa, Italy, pages 1548–1551. http://www.lrec-conf.org/proceedings/lrec2006/summaries/52.html.

Constantin Orăsan, Ruslan Mitkov, and Laura Hasler. 2003. CAST: a computer-aided summarisation tool. In *Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics (EACL)*. Budapest, Hungary, pages 135–138. http://aclweb.org/anthology/E03-1066.

Karolina Owczarzak, John M. Conroy, Hoa Trang Dang, and Ani Nenkova. 2012. An assessment of the accuracy of automatic evaluation in summarization. In *Proceedings of Workshop on Evaluation Metrics and System Comparison for Automatic Summarization*. Montréal, Canada, pages 1–9. http://aclweb.org/anthology/W12-2601.

Sun Park and Dong Un An. 2010. Automatic Query-based Personalized Summarization That Uses Pseudo Relevance Feedback with NMF. In *Proceedings of the 4th International Conference on Ubiquitous Information Management and Communication (ICUIMC)*. pages 61:1–61:7. https://doi.org/10.1145/2108616.2108690.

Helen Petrie and Nigel Bevan. 2009. The evaluation of accessibility, usability, and user experience. In Constantine Stephanidis, editor, *The Universal Access Handbook*, Boca Raton: CRC Press, Human Factors and Ergonomics, chapter 20, pages 1–16. https://doi.org/10.1201/9781420064995-c20.

John C. Platt. 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances In Large Margin Classifiers*. MIT Press, pages 61–74.

Mickaël Poussevin, Vincent Guigue, and Patrick Gallinari. 2015. Extended recommendation framework: Generating the text of a user review as a personalized summary. In *Proceedings of the 2nd Workshop on New Trends on Content-Based Recommender Systems co-located with 9th ACM Conference on Recommender Systems (RecSys 2015), Vienna, Austria, September 16-20, 2015*. pages 34–41. http://ceur-ws.org/Vol-1448/paper7.pdf.

Nils Reimers, Judith Eckle-Kohler, Carsten Schnober, Jungi Kim, and Iryna Gurevych. 2014. GermEval-2014: Nested Named Entity Recognition with Neural Networks. In *Workshop Proceedings of the 12th Edition of the KONVENS Conference*. Hildesheim, Germany, pages 117–120.

Kristian Woodsend and Mirella Lapata. 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 (EMNLP/CoNLL)*. Jeju Island, Korea, pages 233–243. http://aclweb.org/anthology/D12-1022.

Klaus Zechner. 2002. Automatic summarization of open-domain multiparty dialogues in diverse genres. *Journal of Computational Linguistics* 28(4):447–485. https://doi.org/10.1162/089120102762671945.

Haiqin Zhang, Zheng Chen Wei-ying Ma, and Qingsheng Cai. 2003. A study for documents summarization based on personal annotation. In *Proceedings of the HLT-NAACL 03 on Text Summarization Workshop*. pages 41–48. https://doi.org/10.3115/1119467.1119473.