Phrase-based Compressive Cross-Language Summarization

Jin-ge Yao **Xiaojun Wan** **Jianguo Xiao**
Institute of Computer Science and Technology, Peking University, Beijing 100871, China
Key Laboratory of Computational Linguistic (Peking University), MOE, China
{yaojinge, wanxiaojun, xiaojianguo}@pku.edu.cn

Abstract

The task of cross-language document summarization is to create a summary in a target language from documents in a different source language. Previous methods only involve direct extraction of automatically translated sentences from the original documents. Inspired by phrase-based machine translation, we propose a phrase-based model to simultaneously perform sentence scoring, extraction and compression. We design a greedy algorithm to approximately optimize the score function. Experimental results show that our methods outperform the state-of-the-art extractive systems while maintaining similar grammatical quality.

1 Introduction

The task of cross-language summarization is to produce a summary in a target language from documents written in a different source language. This task is particularly useful for readers to quickly get the main idea of documents written in a source language that they are not familiar with. Following Wan (2011), we focus on English-to-Chinese summarization in this work.

The simplest and the most straightforward way to perform cross-language summarization is pipelining general summarization and machine translation. Such systems either translate all the documents before running generic summarization algorithms on the translated documents, or summarize from the original documents and then only translate the produced summary into the target language. Wan (2011) show that such pipelining approaches are inferior to methods that utilize information from both sides. In that work, the author proposes graph-based models and achieves fair amount of improvement. However, to the best of our knowledge, no previous work of this task tries to focus on summarization beyond pure sentence extraction.

On the other hand, cross-language summarization can be seen as a special kind of machine translation: translating the original documents into a brief summary in a different language. Inspired by phrase-based machine translation models (Koehn et al., 2003), we propose a phrase-based scoring scheme for cross-language summarization in this work.

Since our framework is based on phrases, we are not limited to produce extractive summaries. We can use the scoring scheme to perform joint sentence selection and compression. Unlike typical sentence compression methods, our proposed algorithm does not require additional syntactic preprocessing such as part-of-speech tagging or syntactic parsing. We only utilize information from translated texts with phrase alignments. The scoring function consists of a submodular term of compressed sentences and a bounded distortion penalty term. We design a greedy procedure to efficiently get approximate solutions.

For experimental evaluation, we use the DUC2001 dataset with manually translated reference Chinese summaries. Results based on the ROUGE metrics show the effectiveness of our proposed methods. We also conduct manual evaluation and the results suggest that the linguistic quality of produced summaries is not decreased by too much, compared with extractive counterparts. In some cases, the grammatical smoothness can even be improved by compression.

The contributions of this paper include:

- Utilizing the phrase alignment information, we design a scoring scheme for the cross-language document summarization task.
- We design an efficient greedy algorithm to generate summaries. The greedy algorithm is

Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pages 118–127,
Lisbon, Portugal, 17-21 September 2015.

partially submodular and has a provable constant approximation factor to the optimal solution up to a small constant.

- We achieve state-of-the-art results using the extractive counterpart of our compressive summarization framework. Performance in terms of ROUGE metrics can be significantly improved when simultaneously performing extraction and compression.

2 Background

Document summarization can be treated as a special kind of translation process: translating from a bunch of related source documents to a short target summary. This analogy also holds for cross-language document summarization, with the only difference that the languages of source documents and the target summary are different.

Our design of sentence scoring function for cross-language document summarization purpose is inspired by phrase-based machine translation models. Here we briefly describe the general idea of phrase-based translation. One may refer to Koehn (2009) for more detailed description.

2.1 Phrase-based Machine Translation

Phrase-based machine translation models are currently giving state-of-the-art translations for many pairs of languages and dominating modern statistical machine translation. Classical word-based IBM models cannot capture local contextual information and local reordering very well. Phrase-based translation models operate on lexical entries with more than one word on the source language and the target language. The allowance of multi-word expressions is believed to be the main reason for the improvements that phrase-based models give. Note that these multi-word expressions, typically addressed as phrases in machine translation literature, are essentially continuous n-grams and do not need to be linguistically integrate and meaningful constituents.

Define y as a phrase-based derivation, or more precisely a finite sequence of phrases $p_1, p_2, \ldots, p_L$. For any derivation y we use $e(y)$ to refer to the target-side translation text defined by y. This translation is derived by concatenating the strings $e(p_1), e(p_2), \ldots, e(p_L)$. The scoring scheme for a phrase-based derivation y from the source sentence to the target sentence $e(y)$ is:

$$\begin{aligned} f(y) &= \sum_{k=1}^{L} g(p_k) + LM(e(y)) \\ &\quad + \sum_{k=1}^{L-1} \eta |start(p_{k+1}) - 1 - end(p_k)| \end{aligned}$$

where $LM(\cdot)$ is the target-side language model score, $g(\cdot)$ is the score function of phrases, $\eta < 0$ is the distortion parameter for penalizing the distance between neighboring phrases in the derivation. Note that the phrases addressed here are typically continuous n-grams and need not to be grammatical linguistic phrasal units. Later we will directly use phrases provided by modern machine translation systems.

Searching for the best translation under this score definition is difficult in general. Thus approximate decoding algorithms such as beam search should be applied. Meanwhile, several constraints should be satisfied during the decoding process. The most important one is to set a constant limit of the distortion term $|start(p_{k+1}) - 1 - end(p_k)| \leq \delta$ to exhibit derivations with distant phrase translations.

3 Phrase-based Cross-Language Summarization

Inspired by the general idea of phrase-based machine translation, we describe our proposed phrase-based model for cross-language summarization in this section.

3.1 Phrase-based Sentence Scoring

In the context of cross-language summarization, here we assume that we can also have phrases in both source and target languages along with phrase alignments between the two sides. For summarization purposes, we may wish to select sentences containing more important phrases. Then it is plausible to measure the scores of these aligned phrases via importance weighing.

Inspired by phrase-based translation models, we can assign phrase-based scores to sentences from the translated documents for summarization purposes. We define our scoring function for each sentence s as:

$$\begin{aligned} F(s) &= \sum_{p \in s} d_0 g(p) + bg(s) \\ &\quad + \eta \, dist(y(s)) \end{aligned}$$

Here in the first term $g(\cdot)$ is the score of phrase p, which can be simply set to document frequency. The phrase score is penalized with a constant damping factor d_0 to decay scores for repeated phrases. The second term $bg(s)$ is the bigram score of sentence s. It is used here to simulate the effect of language models in phrase-based translation models. Denoting $y(s)$ as the phrase-based derivation (as mentioned earlier in the previous section) of sentence s, the last distortion term $dist(y(s)) = \sum_{k=1}^{L} |start(p_{k+1}) - 1 - end(p_k)|$ is exactly the same as the distortion penalty term in phrase-based translation models. This term can be used as a reflection of complexity of the translation. All the above terms can be derived from bilingual sentence pairs with phrase alignments.

Meanwhile, we may also wish to exclude unimportant phrases and badly translated phrases. Our definition can also be used to guide sentence compression by trying to remove redundant phrase.

Based on the definition over sentences, we define our summary scoring measure over a summary S:

$$F(S) = \sum_{p \in S} \sum_{i=1}^{count(p,S)} d^{i-1} g(p) + \sum_{s \in S} bg(s) + \eta \sum_{s \in S} dist(y(s))$$

where d is a predefined constant damping factor to penalize repeated occurrences of the same phrases, $count(p, S)$ is the number of occurrences in the summary S for phrase p. All other terms are inherited from the sentence score definition.

In the next section we describe our framework to efficiently utilize this scoring function for cross-language summarization.

3.2 A Greedy Algorithm for Compressed Sentence Selection

Utilizing the phrase-based score definition of sentences, we can use greedy algorithms to simultaneously perform sentence selection and sentence compression. Assuming that we have a predefined budget B (e.g. total number of Chinese characters allowed) to restrict the total length of a generated summary. We use $C(S)$ to denote the cost of a summary S, measured by the number of Chinese characters contained in total. The greedy algorithm we will use for our compressive summarization is listed in Algorithm 1.

Algorithm 1 A greedy algorithm for phrase-based summarization

1: $S \leftarrow \emptyset$
2: $i \leftarrow 1$
3: $single\_best = \text{argmax}_{s \in U, C(\{s\}) \leq B} F(\{s\})$
4: **while** $U \neq \emptyset$ **do**
5: $\quad s_i = \text{argmax}_{s \in U} \frac{F(S_{i-1} \cup \{s\}) - F(S_{i-1})}{C(\{s\})^r}$
6: $\quad$ **if** $C(S_{i-1} \cup \{s\}) \leq B$ **then**
7: $\quad\quad S_i \leftarrow S_{i-1} \cup \{s\}$
8: $\quad\quad i \leftarrow i + 1$
9: $\quad$ **end if**
10: $\quad U \leftarrow U \setminus \{s_i\}$
11: **end while**
12: **return** $S^* = \text{argmax}_{S \in \{single\_best, S_i\}} F(S)$

The space U denotes the set of all possible compressed sentences. In each iteration, the algorithm tries to find the compressed sentence with maximum gain-cost ratio (Line 5, where we will follow previous work to set $r = 1$), and merge it to the summary set at the current iteration (denoted as S_i). The target is to find the compression with maximum gain-cost ratio. This will be discussed in the next section. Note that the algorithm is also naturally applicable to extractive summarization. For extractive summarization, Line 5 corresponds to direct calculations of sentence scores based on our proposed phrase-based function and U will denote all full sentences from the original translated documents.

The outline of this algorithm is very similar to the greedy algorithm used by Morita et al. (2013) for subtree extraction, except that in our context the increase of cost function when adding a sentence is exactly the cost of that sentence.

When the distortion term is ignored ($\eta = 0$), the scoring function is clearly submodular [1] (Lin and Bilmes, 2010) in terms of the set of compressed sentences, since the score now only consists of functional gains of phrases along with bigrams of a compressed sentence. Morita et al. (2013) have proved that when $r = 1$, this greedy algorithm will achieve a constant approximation factor $\frac{1}{2}(1 - e^{-1})$ to the optimal solution. Note that this only gives us the worst case guarantee. What we can achieve in practice is usually far better.

On the other hand, setting $\eta < 0$ will not affect

[1] A set function $F : 2^U \rightarrow \mathbb{R}$ defined over subsets of a universe set U is said to be *submodular* iff it satisfies the *diminishing returns* property: $\forall S \subseteq T \subseteq U \setminus u$, we have $F(S \cup \{u\}) - F(S) \geq F(T \cup \{u\}) - F(T)$.

the performance guarantee too much. Intuitively this is because in most phrase-based translation models a distortion limit constraint $|start(p_{k+1}) - 1 - end(p_k)| \leq \delta$ will be applied on distortion terms, while performing sentence compression can never increase distortion. The main conclusion is formulated as:

Theorem 1. *If Algorithm 1 outputs S^{greedy} while the optimal solution is OPT, we have*

$$F(S^{greedy}) \geq \frac{1}{2}(1 - e^{-1})F(OPT) + \frac{1}{2}\eta\gamma.$$

Here $\gamma > 0$ is a constant controlled by distortion difference between sentences, which is relatively small in practice compared with phrase scores. $\eta < 0$ is the distortion parameter. Note that when η is set to be 0, the scoring function is submodular and then we recover the $\frac{1}{2}(1 - e^{-1})$ approximation factor as studied by Morita et al. (2013). We leave the proof of Theorem 1 to supplementary materials due to space limit. The submodularity term in the score plays an important role in the proof.

3.3 Finding the Maximum Density Compression

In Algorithm 1, the most important part is the greedy selection process (Line 5). The greedy selection criteria here is to maximize the gain-cost ratio. For compressive summarization, we are trying to compress each unselected sentence s to $\tilde{s}$, aiming at maximizing the gain-cost ratio, where the gain corresponds to

$$\begin{aligned} & F(S_{i-1} \cup \{s\}) - F(S_{i-1}) \\ = & \sum_{p \in s} \sum_{i=1}^{count(p,S)} d^{i-1} g(p) + bg(s) + \eta dist(s), \end{aligned}$$

and then add the compressed sentence $\tilde{s}$ with maximum gain-cost ratio to the summary. We will also address the compression process for each sentence as finding the maximum density compression. The whole framework forms a joint selection and compression process.

In our phrase-based scoring for sentences, although there exist no apparent optimal substructure available for exact dynamic programming due to nonlocal distortion penalty, we can have a tractable approximate procedure since the search space is only defined by local decisions on whether a phrase should be kept or dropped.

Our compression process for each sentence s is displayed in Algorithm 2. It gradually expands the set of phrases to be kept in the final compression, from the initial set of large density phrases (Line 4, assuming that phrases with large scores and small costs will always be kept), we can recover the compression with maximum density. The function $dist(\cdot, \cdot)$ is the unit distortion penalty defined as $dist(a, b) = |start(b) - 1 - end(a)|$. We define $p.score$ to be the sum of damped phrase score for phrase p, i.e. $p.score = \sum_{i=1}^{count(p,S_{i-1})} d^{i-1} g(p)$, when the current partial summary is S_{i-1}. Therefore during each iteration of the greedy selection process, the compression procedure will also be affected by sentences that have already been included. Define $p.cost$ as the number of words p contains.

Algorithm 2 A growing algorithm for finding the maximum density compressed sentence

```
1: function GET_MAX_DENSITY_COMPRESSION(s, S_{i-1})
2:     queue Q ← ∅, kept ← ∅
3:     for each phrase p in s.phrases do
4:         if p.score/p.cost > 1 then
5:             kept ← kept ∪{p}
6:             Q.enqueue(p)
7:         end if
8:     end for
9:     while Q ≠ ∅ do
10:        p ← Q.deque()
11:        ppv ← p.previous_phrase, pnx ← p.next_phrase
12:        if (ppv.score+bg(ppv,p)+ηdist(ppv,p)) / (ppv.cost+p.cost) > 1 then
13:            Q.enqueue(ppv), kept ← kept ∪{ppv}
14:        end if
15:        if (pnx.score+bg(pnx,p)+ηdist(p,pnx)) / (p.cost+pnx.cost) > 1 then
16:            Q.enqueue(pnx), kept ← kept ∪{pnx}
17:        end if
18:    end while
19: return s̃ = kept, ratio = (F(S_{i-1}∪{s̃}) − F(S_{i-1})) / s̃.cost
20: end function
```

Empirically we find this procedure gives almost the same results with exhaustive search while maintaining efficiency. Assuming that sentence length is no more than L, then the asymptotic complexity of Algorithm 2 will be $O(L)$ since the algorithm requires two passes of all phrases. Therefore the whole framework requires $O(kNL)$ time for a document cluster containing N sentences in total to generate a summary with k sentences.

In the final compressed sentence we just leave the selected phrases continuously as they are, relying on bigram scores to ensure local smoothness. The task is after all a summarization task, where bigram scores play a role of not only controlling grammaticality but keeping main information of

the original documents.

Later we will see that this compression process will not hurt grammatical fluency of translated sentences in general. In many cases it may even improve fluency by deleting redundant parentheses or removing incorrectly reordered (unimportant) phrases.

4 Experiments

4.1 Data

Currently there are not so many available datasets for our particular setting of the cross-language summarization task. Hence we only evaluate our method on the same dataset used by Wan (2011). The dataset is created by manually translating the reference summaries into Chinese from the original DUC 2001 dataset in English. We will refer to this dataset as the DUC 2001 dataset in this paper. There are 30 English document sets in the DUC 2001 dataset for multi-document summarization. Each set contains several documents related to the same topic. Three generic reference English summaries are provided by NIST annotators for each document set. All these English summaries have been translated to Chinese by native Chinese annotators.

All the English sentences in the original documents have been automatically translated into Chinese using Google Translate. We also collect the phrase alignment information from the responses of Google Translate (stored in JSON format) along with the translated texts. We use the Stanford Chinese Word Segmenter [2] for Chinese word segmentation.

The parameters in the algorithms are simply set to be $r = 1, d = 0.5, \eta = -0.5$.

4.2 Evaluation

We will report the performance of our compressive solution, denoted as PBCS (for Phrase-Based Compressive Summarization), with comparisons of the following systems:

- **PBES:** The acronym comes from Phrase-Based Extractive Summarization. It is the extractive counterpart of our solution without calling Algorithm 2.
- **Baseline (EN):** This baseline relies on merely the English-side information for English sentence ranking in the original documents. The scoring function is designed to be document frequencies of English bigrams, which is similar to the second term in our proposed sentence scoring function in Section 3.1 and is submodular. [3] The extracted English summary is finally automatically translated into the corresponding Chinese summary. This is also known as the summary translation scheme.
- **Baseline (CN):** This baseline relies on merely the Chinese-side information for Chinese sentence ranking. The scoring function is similarly defined by document frequency of Chinese bigrams. The Chinese summary sentences are then directly extracted from the translated Chinese documents. This is also known as the document translation scheme.
- **CoRank:** We reimplement the graph-based CoRank algorithm, which gives the state-of-the-art performance on the same DUC 2001 dataset for comparison.
- **Baseline (ENcomp):** This is a compressive baseline where the extracted English sentences in Baseline (EN) will be compressed before being translated to Chinese. The compression process follows from an integer linear program as described by Clarke and Lapata (2008). This baseline gives strong performance as we have found on English DUC 2001 dataset as well as other monolingual datasets.

We experiment with two kinds of summary budgets for comparative study. The first one is limiting the summary length to be no more than five sentences. The second one is limiting the total number of Chinese characters of each produced summary to be no more than 300. They will be addressed as Sentence Budgeting and Character Budgeting in the experimental results respectively.

Similar to traditional summarization tasks, we use the ROUGE metrics for automatic evaluation of all systems in comparison. The ROUGE metrics measure summary quality by counting overlapping word units (e.g. n-grams) between the candidate summary and the reference summary. Following previous work in the same

[2] http://nlp.stanford.edu/software/segmenter.shtml

[3] In our experiments this method gives similar performance compared with graph-based pipelining baselines implemented in previous work.

task, we report the following ROUGE F-measure scores: ROUGE-1 (unigrams), ROUGE-2 (bigrams), ROUGE-W (weighted longest common subsequence; weight=1.2), ROUGE-L (longest common subsequences), and ROUGE-SU4 (skip bigrams with a maximum distance of 4). Here we investigate two kinds of ROUGE metrics for Chinese: ROUGE metrics based on words (after Chinese word segmentation) and ROUGE metrics based on singleton Chinese characters. The latter metrics will not suffer from the problem of word segmentation inconsistency.

To compare our method with extractive baselines in terms of information loss and grammatical quality, we also ask three native Chinese students as annotators to carry out manual evaluation. The aspects considered during evaluation include Grammaticality (GR), Non-Redundancy (NR), Referential Clarity (RC), Topical Focus (TF) and Structural Coherence (SC). Each aspect is rated with scores from 1 (poor) to 5 (good) [4]. This evaluation is performed on the same random sample of 10 document sets from the DUC 2001 dataset. One group of the gold-standard summaries is left out for evaluation of human-level performance. The other two groups are shown to the annotators, giving them a sense of topics talked about in the document sets.

4.3 Results and Discussion

Table 1 and Table 2 display the ROUGE results for our proposed methods and the baseline methods, including both word-based and character-based evaluation. We also conduct pairwise t-test and find that almost all the differences between PBCS and other systems are statistically significant with $p \ll 0.01$ [5] except for the ROUGE-W metric.

We have the same observations with previous work on the inferiority of using information from only one-side, while using Chinese-side information only is more beneficial than English-side only. The CoRank algorithm utilizes both sides of information together and achieves significantly better performance over Baseline(EN) and Baseline(CN). Our compressive system outperforms the CoRank algorithm [6] in all metrics.

Also our system overperforms the compressive pipelining system (Baseline(ENcomp)) as well. Note that the latter only considers information from the source language side. Meanwhile sentence compression may sometimes causes worse translations compared with translating the full original sentence.

For manual evaluation, the average score and standard deviation for each metric is displayed in Table 3. From the comparison between compressive summarization and the extractive version, there exist slight improvements of non-redundancy. This exactly matches what we can expect from sentence compression that keeps only important part and drop redundancy. We also observe certain amount of improvements on referential clarity. This may be a result of deletions of some phrases containing pronouns, such as *he said*. Most of such phrases are semantically unimportant and will be dropped during the process of finding the maximum density compression.

Despite not directly using syntactic information, our compressive summaries do not suffer too much loss of grammaticality. This suggest that bigrams can be treated as good indicators of local grammatical smoothness. We reckon that sentences describing the same events may partially share descriptive bigram patterns, thus sentences selected by the algorithm will consist of mostly important patterns that appear repeatedly in the original document cluster. Only those words that are neither semantically important nor syntactically pivotal will be deleted.

Figure 1 lists the summaries for the first document set D04 in the DUC 2001 dataset produced by the proposed compressive system. The Chinese side sentences have been split with spaces according to phrase alignment results. Phrases that have been compressed are grayed out. We also include original English sentences for reference, with deletions according to word alignments from the Chinese sentences. We can observe that our compressive system tries to compress sentences by removing relatively unimportant phrases. The effect of translation errors (e.g. the word *watch* in *on storm watch* has been incorrectly translated in the example) can also be reduced since those incorrectly translated words will be dropped for having low information gains. In some cases the gram-

[4] Fractional numbers are allowed for cases where the annotators feel uncertain about.

[5] The significance level holds after Bonferroni adjustment, for the purpose of multiple testing.

[6] There exists ignorable difference between the results of our reimplemented version of CoRank and those reported by Wan (2011). We believe that this comes from different machine translation results output by Google Translate.

Sentence Budgeting	ROUGE-1	ROUGE-2	ROUGE-W	ROUGE-L	ROUGE-SU4
Baseline(EN)	0.23655	0.03550	0.05324	0.12559	0.06410
Baseline(CN)	0.23454	0.03858	0.05753	0.13120	0.06962
PBES	0.25313	0.04073	0.06103	0.13583	0.06970
CoRank (reported)	N/A	0.04282	0.06158	0.14521	0.07805
CoRank (reimplemented)	0.24257	0.04115	0.06076	0.13717	0.07453
Baseline(ENcomp)	0.24879	0.04441	0.05865	0.13233	0.07543
PBCS	**0.26872**	**0.04815**	**0.06425**	**0.14607**	**0.08065**

Character Budgeting	ROUGE-1	ROUGE-2	ROUGE-W	ROUGE-L	ROUGE-SU4
Baseline(EN)	0.21460	0.03494	0.05150	0.12343	0.06278
Baseline(CN)	0.21589	0.03732	0.05420	0.12867	0.06405
PBES	0.22825	0.04037	0.05527	0.12856	0.06894
CoRank (reimplemented)	0.22593	0.04069	0.05887	0.12818	0.07241
Baseline(ENcomp)	0.23663	0.04245	0.06134	0.13070	0.07365
PBCS	**0.24917**	**0.04632**	**0.06252**	**0.13591**	**0.07953**

Table 1: Results of word-based ROUGE evaluation

Sentence Budgeting	ROUGE-1	ROUGE-2	ROUGE-W	ROUGE-L	ROUGE-SU4
Baseline(EN)	0.34842	0.11823	0.05505	0.15665	0.12320
Baseline(CN)	0.34901	0.12015	0.05664	0.15942	0.12625
PBES	0.36618	0.12281	0.05913	0.16018	0.11317
CoRank (reimplemented)	0.37601	0.12570	0.06088	0.17350	0.13352
Baseline(ENcomp)	0.36982	0.13001	0.06906	0.16233	0.13543
PBCS	**0.37890**	**0.13549**	**0.07102**	**0.17632**	**0.14098**

Character Budgeting	ROUGE-1	ROUGE-2	ROUGE-W	ROUGE-L	ROUGE-SU4
Baseline(EN)	0.33602	0.10546	0.05263	0.15437	0.12161
Baseline(CN)	0.34075	0.12012	0.05678	0.15736	0.11981
PBES	0.35483	0.11902	0.05642	0.15899	0.11205
CoRank (reimplemented)	0.36147	0.12305	0.05847	**0.16962**	0.13364
Baseline(ENcomp)	0.36654	0.12960	0.06503	0.15987	0.13421
PBCS	**0.37842**	**0.13441**	**0.07005**	**0.16928**	**0.13985**

Table 2: Results of character-based ROUGE evaluation

System	GR	NR	RC	TF	SC
CoRank	3.00±0.75	3.35±0.57	3.55±0.82	3.90±0.79	3.55±0.74
PBES	2.90±0.89	3.25±0.70	3.50±0.87	3.96±0.80	3.45±0.50
PBCS	2.90±0.83	3.60±0.49	3.75±0.82	3.93±0.68	3.40±0.58
Human	4.60±0.49	4.15±0.73	4.35±0.73	4.93±0.25	3.90±0.94

Table 3: Manual evaluation results

matical fluency can even be improved from sentence compression, as redundant parentheses may sometimes be removed. We leave the output summaries from all systems for the same document set to supplementary materials.

In our experiments, we also study the influence of relevant parameter settings. Figure 2a depicts the variation of ROUGE-2 F-measure when changing the damping factor d from different values in $\{1, 2^{-1}, 3^{-1}, 4^{-1}, 5^{-1}\}$, while $\eta = -0.5$ being fixed. We can see that under proper range the value of d does not effect the result for too much. No damping or too much damping will severely decrease the performance. Figure 2b shows the performance change under different settings of the distortion parameter η taking values from $\{0, -0.2, -0.5, -1, -3\}$, while fixing $d = 0.5$. The results suggest that, for our purposes of summarization, the difference of considering distortion penalty or not is obvious. At certain level, the effect brought by different values distortion parameter becomes stable.

We also empirically study the effect of approximation. The compressive summarization framework proposed in this paper can be trivially cast into an integer linear program (ILP), with the number of variables being too large to make the problem tractable [7]. In this experiment, we use

[7] By casting decisions on whether to select a certain phrase or bigram as binary variables, with additional linear constraints on phrase/bigram selection consistency, we get an ILP with essentially the same objective function and a linear budget constraint. This is conceptually equivalent to solving

凯特 女士 硬朗 ， 紧急服务 在佛罗里达州 的 戴德 县， 承担了 风暴 的冲击 主任 估计， 安德鲁 已经 造成 150亿 美元 到 200亿 美元 的损害 （ 75亿 英镑 ， 100亿 英镑 ） 。
Ms Kate Hale, director of emergency services in Florida's Dade County, which bore the brunt of the storm, estimated that Andrew had already caused Dollars 15bn to Dollars 20bn (Pounds 7.5bn-Pounds 10bn) of damage.

雨果飓风 ， 袭击 东海岸 在 1989年9月 ， 花费了 保险业 约 42亿 美元 。
Hurricane Hugo, which hit the east coast in September 1989, cost the insurance industry about Dollars 4.2bn.

美国城市 沿 墨西哥湾的 阿拉巴马州 到得克萨斯州 东部 是 在 风暴 手表 昨晚 安德鲁 飓风 向西 横跨 佛罗里达州南部 席卷 后 ，造成 至少 八人死亡 和严重的 财产损失 。
US CITIES along the Gulf of Mexico from Alabama to eastern Texas were on storm watch last night as Hurricane Andrew headed west after sweeping across southern Florida, causing at least eight deaths and severe property damage.

过去的 严重 飓风 美国 ，雨果 ， 袭击 南卡罗来纳州 于1989年 ， 耗资 从 保险 损失 行业 42亿 美元 ，但 造成的 总伤害 的 估计 60亿 美元 和 100亿 美元 之间 不等 。
The last serious US hurricane, Hugo, which struck South Carolina in 1989, cost the industry Dollars 4.2bn from insured losses, though estimates of the total damage caused ranged between Dollars 6bn and Dollars 10bn.

最初的 报道称， 至少有一人 已经 死亡 ， 75 人受伤 ，数千 取得 沿着 路易斯安那州海岸 无家可归 ， 14 证实 在佛罗里达州和 死亡 三 巴哈马群岛 后 。
Initial reports said at least one person had died, 75 been injured and thousands made homeless along the Louisiana coast, after 14 confirmed deaths in Florida and three in the Bahamas.

Figure 1: Example compressive summary

lp_solve package [8] as the ILP solver to obtain an exact solution on the first document cluster (D04) in DUC 2001 dataset. In Figure 2c, we depict the objective value achieved by ILP as exact solution, comparing with results from sentences which are gradually selected and compressed by our greedy algorithm. We can see that the approximation is close.

5 Related Work

The task focused in this paper is cross-language document summarization. Several pilot studies have investigated this task. Before Wan (2011)'s work that explicitly utilizes bilingual information in a graph-based framework, earlier methods often use information only from one language (de Chalendar et al., 2005; Pingali et al., 2007; Orasan and Chiorean, 2008; Litvak et al., 2010).

This work is closely related to greedy algorithms for budgeted submodular maximization. Many studies have formalized text summarization tasks as submodular maximization problems (Lin and Bilmes, 2010; Lin and Bilmes, 2011; Morita et al., 2013). A more recent work (Dasgupta et al., 2013) discussed the problem of maximizing a function with a submodular part and a non-submodular dispersion term, which may appear to be closer to our scoring functions.

In recent years, some research has made progress beyond extractive summarization, especially in the context of compressive summarization. Zajic et al. (2006) tries a pipeline strategy with heuristics to generate multiple candidate compressions and extract from this compressed sentences. Berg-Kirkpatrick et al. (2011) create linear models of weights learned by structural SVMs for different components and tried to jointly formulate sentence selection and syntax tree trimming in integer linear programs. Woodsend and Lapata (2012) propose quasi tree substitution grammars for multiple rewriting operations. All these methods involve integer linear programming solvers to generate compressed summaries, which is time-consuming for multi-document summarization tasks. Almeida and Martins (2013) form the compressive summarization problem in a more efficient dual decomposition framework. Models for sentence compression and extractive summarization are trained by multi-task learning techniques. Wang et al. (2013) explore different types of compression on constituent parse trees for query-focused summarization. Li et al. (2013) propose a guided sentence compression model with ILP-based summary sentence selection. Their following work (Li et al., 2014) incorporate various constraints on constituent parse trees to improve the linguistic quality of the compressed sentences. In these studies, the best-performing systems require supervised learning for different subtasks. More recent work tries to formulate document summarization tasks as optimization problems and use their solutions to guide sentence compression(Li et al., 2015; Yao et al., 2015). Bing et al. (2015) employ integer linear programming for conducting phrase selection and merging simultaneously to form compressed sentences after phrase extraction.

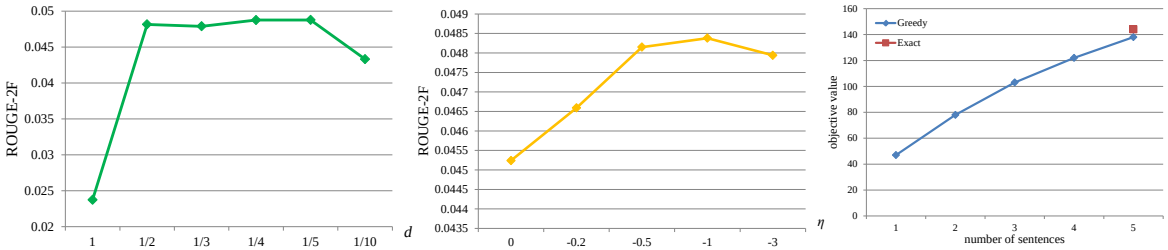
6 Conclusion and Future Work

In this paper we propose a phrase-based framework for the task of cross-language document summarization. The proposed scoring scheme can be naturally operated on compressive summarization. We use efficient greedy procedure to approximately optimize the scoring function. Experimental results show improvements of our compressive solution over state-of-the-art systems. Even though we do not explicitly use any syntactic information, the generated summaries of our system do not lose much grammaticality and fluency.

The scoring function in our framework is in-

the original maximization problem with pruned brute-force enumeration and therefore exactly optimal but too costly.

[8] http://lpsolve.sourceforge.net/



(a) Effect of damping factor d (b) Effect of distortion parameter η (c) Comparison with exact solution

Figure 2: Experimental analysis

spired by earlier phrase-based machine translation models. Our next step is to try more fine-grained scoring schemes using similar techniques from modern approaches of statistical machine translation. To further improve grammaticality of generated summaries, we may try to sacrifice the time efficiency for a little bit and use syntactic information provided by syntactic parsers.

Our framework currently uses only the single best translation. It will be more powerful to integrate machine translation and summarization, utilizing multiple possible translations.

Currently many successful statistical machine translation systems are phrase-based with alignment information provided and we utilize this fact in this work. It is interesting to explore how will the performance be affected if we are only provided with parallel sentences and then alignments can only be derived using an independent aligner.

Acknowledgments

We thank all the anonymous reviewers for helpful comments and suggestions. This work was supported by National Hi-Tech Research and Development Program (863 Program) of China (2015AA015403, 2014AA015102) and National Natural Science Foundation of China (61170166, 61331011). The contact author of this paper, according to the meaning given to this role by Peking University, is Xiaojun Wan.

References

Miguel Almeida and Andre Martins. 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, August. Association for Computational Linguistics.

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*, pages 481–490, Portland, Oregon, USA, June. Association for Computational Linguistics.

Lidong Bing, Piji Li, Yi Liao, Wai Lam, Weiwei Guo, and Rebecca Passonneau. 2015. Abstractive multi-document summarization via phrase selection and merging. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1587–1597, Beijing, China, July. Association for Computational Linguistics.

James Clarke and Mirella Lapata. 2008. Global inference for sentence compression: An integer linear programming approach. *Journal of Artificial Intelligence Research*, 31:273–381.

Anirban Dasgupta, Ravi Kumar, and Sujith Ravi. 2013. Summarization through submodularity and dispersion. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1014–1022, Sofia, Bulgaria, August. Association for Computational Linguistics.

Gaël de Chalendar, Romaric Besançon, Olivier Ferret, Gregory Grefenstette, and Olivier Mesnard. 2005. Crosslingual summarization with thematic extraction, syntactic sentence simplification, and bilingual generation. In *Workshop on Crossing Barriers in Text Summarization Research, 5th International Conference on Recent Advances in Natural Language Processing (RANLP2005)*.

Philipp Koehn, Franz Josef Och, and Marcu Daniel. 2003. Statistical phrase-based translation. In *Human Language Technologies: The 2003 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 48–54, Edmonton, May-June. Association for Computational Linguistics.

Philipp Koehn. 2009. *Statistical Machine Translation*. Cambridge University Press.

Chen Li, Fei Liu, Fuliang Weng, and Yang Liu. 2013. 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, October. Association for Computational Linguistics.

Chen Li, Yang Liu, Fei Liu, Lin Zhao, and Fuliang Weng. 2014. Improving multi-documents summarization by sentence compression based on expanded constituent parse trees. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 691–701, Doha, Qatar, October. Association for Computational Linguistics.

Piji Li, Lidong Bing, Wai Lam, Hang Li, and Yi Liao. 2015. Reader-aware multi-document summarization via sparse coding. In *IJCAI*.

Hui Lin and Jeff Bilmes. 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, Los Angeles, California, June. Association for Computational Linguistics.

Hui Lin and Jeff Bilmes. 2011. A class of submodular functions for document summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 510–520, Portland, Oregon, USA, June. Association for Computational Linguistics.

Marina Litvak, Mark Last, and Menahem Friedman. 2010. A new approach to improving multilingual summarization using a genetic algorithm. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 927–936, Uppsala, Sweden, July. Association for Computational Linguistics.

Hajime Morita, Ryohei Sasano, Hiroya Takamura, and Manabu Okumura. 2013. Subtree extractive summarization via submodular maximization. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1023–1032, Sofia, Bulgaria, August. Association for Computational Linguistics.

Constantin Orasan and Oana Andreea Chiorean. 2008. Evaluation of a cross-lingual romanian-english multi-document summariser. In *LREC*.

Prasad Pingali, Jagadeesh Jagarlamudi, and Vasudeva Varma. 2007. Experiments in cross language query focused multi-document summarization. In *Workshop on Cross Lingual Information Access Addressing the Information Need of Multilingual Societies in IJCAI2007*. Citeseer.

Xiaojun Wan. 2011. Using bilingual information for cross-language document summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1546–1555, Portland, Oregon, USA, June. Association for Computational Linguistics.

Lu Wang, Hema Raghavan, Vittorio Castelli, Radu Florian, and Claire Cardie. 2013. A sentence compression based framework to query-focused multi-document summarization. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1384–1394, Sofia, Bulgaria, August. Association for Computational Linguistics.

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*, pages 233–243. Association for Computational Linguistics.

Jin-ge Yao, Xiaojun Wan, and Jianguo Xiao. 2015. Compressive document summarization via sparse optimization. In *IJCAI*.

David M Zajic, Bonnie Dorr, Jimmy Lin, and Richard Schwartz. 2006. Sentence compression as a component of a multi-document summarization system. In *Proceedings of the 2006 Document Understanding Workshop, New York*.