Global Inference

- Let us define document D with t_n textual units

$$D = t_1, t_2, \ldots, t_{n-1}, t_n$$

- Let $Rel(i)$ be the relevance of t_i to be in the summary
- Let $Red(i,j)$ be the redundancy between t_i and t_j
- Let $l(i)$ be the length of t_i

Inference Problem

- The inference problem is to select a subset S of textual units from D such that summary score of S, i.e., $s(S)$, is maximized.
- $S = \arg\max_{S \subseteq D} \left[\sum_{t_i \in S} Rel(i) - \sum_{t_i, t_j \in S, i<j} Red(i,j) \right]$ such that $\sum_{t_i \in S} l(i) \leq K$, where k denotes the maximum length of the summary

A Greedy Solution

1. Sort D so that $Rel(i) > Rel(i+1) \forall i$
2. $S = \{t_1\}$
3. while $\sum_{t_i \in S} l(i) < K$
4. $\quad t_j = \arg\max_{t_j \in D-S} s(S \cup \{t_j\})$
5. $\quad S = S \cup \{t_j\}$
6. return S

Integer Linear Programming (ILP)

- Greedy algorithm is an approximate solution
- Use exact solution algorithms with ILP
- ILP is a constrained optimization problem
- Many solvers on the web
- Define the constraints based on relevance and redundancy for summarization

Sentence Level ILP Formulation

Optimization Function

maximize $\sum_i \alpha_i Rel(i) - \sum_{i<j} \alpha_{ij} Red(i,j)$

Sentence Level ILP Formulation

Optimization Function

maximize $\sum_i \alpha_i Rel(i) - \sum_{i<j} \alpha_{ij} Red(i,j)$

Constraints

such that $\forall i,j$:

- $\alpha_i, \alpha_{ij} \in \{0,1\}$
- $\sum_i \alpha_i l(i) \leq K$
- $\alpha_{ij} - \alpha_i \leq 0$
- $\alpha_{ij} - \alpha_j \leq 0$
- $\alpha_i + \alpha_j - \alpha_{ij} \leq 1$

Sentence Level ILP Formulation

Optimization Function

maximize $\sum_i \alpha_i Rel(i) - \sum_{i<j} \alpha_{ij} Red(i,j)$

Constraints

such that $\forall i,j$:

- $\alpha_i, \alpha_{ij} \in \{0,1\}$
- $\sum_i \alpha_i l(i) \leq K$
- $\alpha_{ij} - \alpha_i \leq 0$
- $\alpha_{ij} - \alpha_j \leq 0$
- $\alpha_i + \alpha_j - \alpha_{ij} \leq 1$

Is generic enough

Depending on your task, you can define your own optimization function and constrains.

The next steps: Sentence Ordering

The next steps: Sentence Ordering

Chronological ordering: the simplest method

List the sentences in the order, they appear in the document

The next steps: Sentence Ordering

Chronological ordering: the simplest method

List the sentences in the order, they appear in the document

Coherence

- Choose orderings that make neighboring sentences similar (by cosine)
- Choose orderings in which neighboring sentences discuss the same entity

The next steps: Sentence Ordering

Chronological ordering: the simplest method

List the sentences in the order, they appear in the document

Coherence

- Choose orderings that make neighboring sentences similar (by cosine)
- Choose orderings in which neighboring sentences discuss the same entity

Topical ordering

Learn the ordering of topics in the source documents

The next steps: Simplifying Sentences

Parse sentences, use rules to decide which modifiers to prune

The next steps: Simplifying Sentences

Parse sentences, use rules to decide which modifiers to prune

- **Initial adverbials:** *For example, on the other hand, as a matter of fact, at this point, ...*

The next steps: Simplifying Sentences

Parse sentences, use rules to decide which modifiers to prune

- **Initial adverbials:** *For example, on the other hand, as a matter of fact, at this point, ...*
- **PPs without named entities:** The commercial fishing restrictions in Washington will not be lifted unless the salmon population increases [*PP to a sustainable number*]

The next steps: Simplifying Sentences

Parse sentences, use rules to decide which modifiers to prune

- **Initial adverbials:** *For example, on the other hand, as a matter of fact, at this point, ...*
- **PPs without named entities:** The commercial fishing restrictions in Washington will not be lifted unless the salmon population increases [*PP to a sustainable number*]
- **Attribution clauses:** Rebels agreed to talks with government officials, *international observers said Tuesday*

The next steps: Simplifying Sentences

Parse sentences, use rules to decide which modifiers to prune

- **Initial adverbials:** *For example, on the other hand, as a matter of fact, at this point, ...*
- **PPs without named entities:** The commercial fishing restrictions in Washington will not be lifted unless the salmon population increases [*PP to a sustainable number*]
- **Attribution clauses:** Rebels agreed to talks with government officials, *international observers said Tuesday*
- **Appositives:** Rajan*, 28, an artist who was living at the time in Philadelphia,* found the inspiration in the back of city magazines

A Scalable Global Model for Summarization

Dan Gillick[1,2], **Benoit Favre**[2]
[1] Computer Science Division, University of California Berkeley, USA
[2] International Computer Science Institute, Berkeley, USA
{dgillick,favre}@icsi.berkeley.edu

Abstract

We present an Integer Linear Program for exact inference under a maximum coverage model for automatic summarization. We compare our model, which operates at the subsentence or "concept"-level, to a sentence-level model, previously solved with an ILP. Our model scales more efficiently to larger problems because it does not require a quadratic number of variables to address redundancy in pairs of selected sentences. We also show how to include sentence compression in the ILP formulation, which has the desirable property of performing compression and sentence selection simultaneously. The resulting system performs at least as well as the best systems participating in the recent Text Analysis Conference, as judged by a variety of automatic and manual content-based metrics.

1 Introduction

Automatic summarization systems are typically extractive or abstractive. Since abstraction is quite hard, the most successful systems tested at the Text Analysis Conference (TAC) and Document Understanding Conference (DUC)[1], for example, are extractive. In particular, sentence selection represents a reasonable trade-off between linguistic quality, guaranteed by longer textual units, and summary content, often improved with shorter units.

Whereas the majority of approaches employ a greedy search to find a set of sentences that is both relevant and non-redundant (Goldstein et al., 2000; Nenkova and Vanderwende, 2005), some recent work focuses on improved search (McDonald, 2007; Yih et al., 2007). Among them, McDonald is the first to consider a non-approximated maximization of an objective function through Integer Linear Programming (ILP), which improves on a greedy search by 4-12%. His formulation assumes that the quality of a summary is proportional to the sum of the relevance scores of the selected sentences, penalized by the sum of the redundancy scores of all pairs of selected sentences. Under a maximum summary length constraint, this problem can be expressed as a quadratic knapsack (Gallo et al., 1980) and many methods are available to solve it (Pisinger et al., 2005). However, McDonald reports that the method is not scalable above 100 input sentences and discusses more practical approximations. Still, an ILP formulation is appealing because it gives exact solutions and lends itself well to extensions through additional constraints.

Methods like McDonald's, including the well-known Maximal Marginal Relevance (MMR) algorithm (Goldstein et al., 2000), are subject to another problem: Summary-level redundancy is not always well modeled by pairwise sentence-level redundancy. Figure 1 shows an example where the combination of sentences (1) and (2) overlaps completely with sentence (3), a fact not captured by pairwise redundancy measures. Redundancy, like content selection, is a global problem.

Here, we discuss a model for sentence selection with a globally optimal solution that also addresses redundancy globally. We choose to represent infor-

[1] TAC is a continuation of DUC, which ran from 2001-2007.

Proceedings of the NAACL HLT Workshop on Integer Linear Programming for Natural Language Processing, pages 10–18,
Boulder, Colorado, June 2009.

(1) The cat is in the kitchen.
(2) The cat drinks the milk.
(3) The cat drinks the milk in the kitchen.

Figure 1: Example of sentences redundant as a group. Their redundancy is only partially captured by sentence-level pairwise measurement.

mation at a finer granularity than sentences, with concepts, and assume that the value of a summary is the sum of the values of the unique concepts it contains. While the concepts we use in experiments are word n-grams, we use the generic term to emphasize that this is just one possible definition. Only crediting each concept once serves as an implicit global constraint on redundancy. We show how the resulting optimization problem can be mapped to an ILP that can be solved efficiently with standard software.

We begin by comparing our model to McDonald's (section 2) and detail the differences between the resulting ILP formulations (section 3), showing that ours can give competitive results (section 4) and offer better scalability[2] (section 5). Next we demonstrate how our ILP formulation can be extended to include efficient parse-tree-based sentence compression (section 6). We review related work (section 7) and conclude with a discussion of potential improvements to the model (section 8).

2 Models

The model proposed by McDonald (2007) considers information and redundancy at the sentence level. The score of a summary is defined as the sum of the relevance scores of the sentences it contains minus the sum of the redundancy scores of each pair of these sentences. If s_i is an indicator for the presence of sentence i in the summary, Rel_i is its relevance, and Red_{ij} is its redundancy with sentence j, then a summary is scored according to:

$$\sum_i Rel_i s_i - \sum_{ij} Red_{ij} s_i s_j$$

Generating a summary under this model involves maximizing this objective function, subject to a length constraint. A variety of choices for Rel_i and Red_{ij} are possible, from simple word overlap metrics to the output of feature-based classifiers trained to perform information retrieval and textual entailment.

As an alternative, we consider information and redundancy at a sub-sentence, "concept" level, modeling the value of a summary as a function of the concepts it covers. While McDonald uses an explicit redundancy term, we model redundancy implicitly: a summary only benefits from including each concept once. With c_i an indicator for the presence of concept i in the summary, and its weight w_i, the objective function is:

$$\sum_i w_i c_i$$

We generate a summary by choosing a set of sentences that maximizes this objective function, subject to the usual length constraint.

In summing over concept weights, we assume that the value of including a concept is not effected by the presence of any other concept in the summary. That is, concepts are assumed to be independent. Choosing a suitable definition for concepts, and a mapping from the input documents to concept weights, is both important and difficult. Concepts could be words, named entities, syntactic subtrees or semantic relations, for example. While deeper semantics make more appealing concepts, their extraction and weighting are much more error-prone. Any error in concept extraction can result in a biased objective function, leading to poor sentence selection.

3 Inference by ILP

Each model presented above can be formalized as an Integer Linear Program, with a solution representing an optimal selection of sentences under the objective function, subject to a length constraint. McDonald observes that the redundancy term makes for a quadratic objective function, which he coerces to a linear function by introducing additional variables s_{ij} that represent the presence of both sentence i and sentence j in the summary. Additional constraints ensure the consistency between the sentence variables (s_i, s_j) and the quadratic term (s_{ij}). With l_i the length of sentence i and L the length limit for

[2] Strictly speaking, exact inference for the models discussed in this paper is NP-hard. Thus we use the term "scalable" in a purely practical sense.

the whole summary, the resulting ILP is:

$$\text{Maximize: } \sum_i Rel_i s_i - \sum_{ij} Red_{ij} s_{ij}$$

$$\text{Subject to: } \sum_j l_j s_j \leq L$$

$$s_{ij} \leq s_i \quad s_{ij} \leq s_j \quad \forall i,j$$

$$s_i + s_j - s_{ij} \leq 1 \quad \forall i,j$$

$$s_i \in \{0,1\} \quad \forall i$$

$$s_{ij} \in \{0,1\} \quad \forall i,j$$

To express our concept-based model as an ILP, we maintain our notation from section 2, with c_i an indicator for the presence of concept i in the summary and s_j an indicator for the presence of sentence j in the summary. We add Occ_{ij} to indicate the occurrence of concept i in sentence j, resulting in a new ILP:

$$\text{Maximize: } \sum_i w_i c_i$$

$$\text{Subject to: } \sum_j l_j s_j \leq L$$

$$s_j Occ_{ij} \leq c_i, \quad \forall i,j \qquad (1)$$

$$\sum_j s_j Occ_{ij} \geq c_i \quad \forall i \qquad (2)$$

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

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

Note that Occ, like Rel and Red, is a constant parameter. The constraints formalized in equations (1) and (2) ensure the logical consistency of the solution: selecting a sentence necessitates selecting all the concepts it contains and selecting a concept is only possible if it is present in at least one selected sentence. Constraint (1) also prevents the inclusion of concept-less sentences.

4 Performance

Here we compare both models on a common summarization task. The data is part of the Text Analysis Conference (TAC) multi-document summarization evaluation and involves generating 100-word summaries from 10 newswire documents, each on a given topic. While the 2008 edition of TAC also includes an update task—additional summaries assuming some prior knowledge—we focus only on the standard task. This includes 48 topics, averaging 235 input sentences (ranging from 47 to 652). Since the mean sentence length is around 25 words, a typical summary consists of 4 sentences.

In order to facilitate comparison, we generate summaries from both models using a common pipeline:

1. Clean input documents. A simple set of rules removes headers and formatting markup.
2. Split text into sentences. We use the unsupervised Punkt system (Kiss and Strunk, 2006).
3. Prune sentences shorter than 5 words.
4. Compute parameters needed by the models.
5. Map to ILP format and solve. We use an open source solver[3].
6. Order sentences picked by the ILP for inclusion in the summary.

The specifics of step 4 are described in detail in (McDonald, 2007) and (Gillick et al., 2008). McDonald's sentence relevance combines word-level cosine similarity with the source document and the inverse of its position (early sentences tend to be more important). Redundancy between a pair of sentences is their cosine similarity. For sentence i in document D,

$$Rel_i = cosine(i, D) + 1/pos(i, D)$$
$$Red_{ij} = cosine(i, j)$$

In our concept-based model, we use word bigrams, weighted by the number of input documents in which they appear. While word bigrams stretch the notion of a concept a bit thin, they are easily extracted and matched (we use stemming to allow slightly more robust matching). Table 1 provides some justification for document frequency as a weighting function. Note that bigrams gave consistently better performance than unigrams or trigrams for a variety of ROUGE measures. Normalizing by document frequency measured over a generic set (TFIDF weighting) degraded ROUGE performance.

[3] gnu.org/software/glpk

Bigrams consisting of two stopwords are pruned, as are those appearing in fewer than three documents.

We largely ignore the sentence ordering problem, sorting the resulting sentences first by source document date, and then by position, so that the order of two originally adjacent sentences is preserved, for example.

Doc. Freq. (D)	1	2	3	4	5	6
In Gold Set	156	48	25	15	10	7
Not in Gold Set	5270	448	114	42	21	11
Relevant (P)	0.03	0.10	0.18	0.26	0.33	0.39

Table 1: There is a strong relationship between the document frequency of input bigrams and the fraction of those bigrams that appear in the human generated "gold" set: Let d_i be document frequency i and p_i be the percent of input bigrams with d_i that are actually in the gold set. Then the correlation $\rho(D, P) = 0.95$ for DUC 2007 and 0.97 for DUC 2006. Data here averaged over all problems in DUC 2007.

The summaries produced by the two systems have been evaluated automatically with ROUGE and manually with the Pyramid metric. In particular, ROUGE-2 is the recall in bigrams with a set of human-written abstractive summaries (Lin, 2004). The Pyramid score arises from a manual alignment of basic facts from the reference summaries, called Summary Content Units (SCUs), in a hypothesis summary (Nenkova and Passonneau, 2004). We used the SCUs provided by the TAC evaluation.

Table 2 compares these results, alongside a baseline that uses the first 100 words of the most recent document. All the scores are significantly different, showing that according to both human and automatic content evaluation, the concept-based model outperforms McDonald's sentence-based model, which in turn outperforms the baseline. Of course, the relevance and redundancy functions used for McDonald's formulation in this experiment are rather primitive, and results would likely improve with better relevance features as used in many TAC systems. Nonetheless, our system based on word bigram concepts, similarly primitive, performed at least as well as any in the TAC evaluation, according to two-tailed t-tests comparing ROUGE, Pyramid, and manually evaluated "content responsiveness" (Dang and Owczarzak, 2008) of our system and the highest scoring system in each category.

System	**ROUGE-2**	**Pyramid**
Baseline	0.058	0.186
McDonald	0.072	0.295
Concepts	0.110	0.345

Table 2: Scores for both systems and a baseline on TAC 2008 data (Set A) for ROUGE-2 and Pyramid evaluations.

5 Scalability

McDonald's sentence-level formulation corresponds to a quadratic knapsack, and he shows his particular variant is NP-hard by reduction to 3-D matching. The concept-level formulation is similar in spirit to the classical maximum coverage problem: Given a set of items X, a set of subsets S of X, and an integer k, the goal is to pick at most k subsets from S that maximizes the size of their union. Maximum coverage is known to be NP-hard by reduction to the set cover problem (Hochbaum, 1996).

Perhaps the simplest way to show that our formulation is NP-hard is by reduction to the knapsack problem (Karp, 1972). Consider the special case where sentences do not share any overlapping concepts. Then, the value of each sentence to the summary is independent of every other sentence. This is a knapsack problem: trying to maximize the value in a container of limited size. Given a solver for our problem, we could solve all knapsack problem instances, so our problem must also be NP-hard.

With n input sentences and m concepts, both formulations generate a quadratic number of constraints. However, McDonald's has $O(n^2)$ variables while ours has $O(n + m)$. In practice, scalability is largely determined by the sparsity of the redundancy matrix Red and the sentence-concept matrix Occ. Efficient solutions thus depend heavily on the choice of redundancy measure in McDonald's formulation and the choice of concepts in ours. Pruning to reduce complexity involves removing low-relevance sentences or ignoring low redundancy values in the former, and corresponds to removing low-weight concepts in the latter. Note that pruning concepts may be more desirable: Pruned sentences are irretrievable, but pruned concepts may well appear in the selected sentences through co-occurrence.

Figure 2 compares ILP run-times for the two