IMPROVED SYSTEM WITH CONTENT REALIZATION

Long Cheng, Sheng Bi, Catherine Wang, Vicky Xiang, Carrie Yuan
University of Washington

Approaches

CONTENT REALIZATION

Heuristic Approach (part 1, based on Nenkova(2008))

```python
def resolve_coref(summary):
    nlp = spacy.load("en_core_web_md")  # Load the English language model
    coref_entities = set()
    refined_summary = []
    for line in summary.readlines():
        sentence = line.strip()
        nlp_sentence = nlp(sentence) # spacy object
        for chunk in nlp_sentence.noun_chunks:
            if chunk.root.pos_ == "PRON": # ignore pronouns
                    continue
            head = chunk.root.text # get the head of the NP
            # Find all tokens attached to the noun chunk's root token
            modifiers = []
            for token in chunk.root.subtree:
                # Check if the token is not the root token itself and is not the noun chunk's text
                if token != chunk.root and token.text != chunk.text:
                    modifiers.append(token.text)
            if head not in coref_entities: # this implies the first occurrence of the entity
                coref_entities.add(head)
            else: # if this entity has appeared before, replace the original NP by the head and remove the modifier
                sentence = sentence.replace(chunk.text, head)
                words = nltk.word_tokenize(sentence)
                head_index = words.index(head)
                modifier = ""
                for i in range(head_index + 1, len(words)):
                    if words[i] in modifiers: # update the modifier sequence
                        modifier += words[i]
                    else: # remove the modifier
                        sentence = sentence.replace(modifier, "")
                        break
        refined_summary.append(sentence)
    return refined_summary
```

Heuristic Approach (part 2)

```python
def content_enhance(text): # each text is only one sentence in this approach
    # Remove bylines and editorial content
    cleaned_text = re.sub(r'Byline:\s*[^.,]*\.|Editorial:\s*[^.,]*\.', '', text, flags=re.IGNORECASE)
    # Remove sentence-initial adverbials and conjunct phrases up to the first comma
    sentences = re.split(r'(?<=[.!?])\s+', cleaned_text)  # Split text into sentences
    cleaned_sentences = []
    for sentence in sentences:
        # Remove adverbials and conjunct phrases up to the first comma
        cleaned_sentence = re.sub(r'^[\w\s]*?,', '', sentence)
        if cleaned_sentence.strip():  # Check if the sentence is not empty
            cleaned_sentences.append(cleaned_sentence)
    cleaned_text = ' '.join(cleaned_sentences)

    # Remove relative clause attributives and attributions without quotes
    cleaned_text = re.sub(r',\s*[^,.]+,|,\s*[^,.]+,', ',', cleaned_text)

    # Use regular expression to remove leading and trailing punctuation
    cleaned_text = re.sub(r'^[^\w]+|[^\w]+$', '', cleaned_text)

    # combine the final word and punctuation of each sentence.
    cleaned_text = '. '.join(sentence.strip(string.punctuation + " ") for sentence in cleaned_text.split('.') if sentence.strip()) + "."

    # turn `` and ' into "
    cleaned_text = re.sub(r'``\s([^`]+)\s\'\'', r'"\1"', cleaned_text)

    # remove the space before 's
    cleaned_text = re.sub(r'\s\'s', r"'s", cleaned_text)

    # remove space after decimal point
    pattern = r'(\d+\.)\s+(\d+)'
    cleaned_text = re.sub(pattern, r'\1\2', cleaned_text)

    # capitalize the first letter
    cleaned_text = cleaned_text[0].capitalize() + cleaned_text[1:]
    return cleaned_text.strip()  # Strip leading/trailing spaces
```

Observations

```
Nature preserve workers in northwest China's Gansu Province have formulated a rescue plan to save giant pandas from food shortage caused by arrow bam
Pandas in Province are suffering from hunger because large tracts of arrow bamboo have bloomed and died.
Wolong is a famous giant panda habitat where the world-known China Conservation and Research Center of the Giant Panda is located.
The Qinling panda has been identified as a sub-species of panda that mainly resides in southwestern Sichuan province.
```

The heuristic method we implemented, while effective in reducing redundancy, exhibits several limitations. Notably, it inaccurately handles proper noun phrases such as "Gansu Province," replacing them with generic terms like "Province" upon subsequent mentions. This oversimplification presents a clear challenge, as the optimal replacement should reflect the specific context, suggesting alternatives like "this province" or "that province."

Furthermore, our approach to coreference resolution falls short in addressing the nuanced disambiguation of pronouns. While we successfully replaced noun phrases with their respective heads, we encountered difficulties in reconciling pronouns within the summary with their original contexts in the news. This issue underscores the need for more sophisticated techniques to ensure coherence and grammatical accuracy.

Upon conducting human evaluations, we observed that the coverage score of LLR is relatively low. Upon reviewing the code, it appears that we might be able to enhance the results by adjusting the circled code segment. Instead of **breaking** the loop, **continuing** the iteration could potentially improve results. This adjustment would allow us to iterate through more sentences and select an appropriate one, rather than halting immediately when the current total length exceeds the upper bound of the summary length (set at 100 in this case).

```
while True: # select sentences
    if ordered_sentences == []:
        break
    chosen = ordered_sentences.pop() # chose the most weighted sentence
    if curr_length + chosen[1][0] <= summary_length:
        include_sentence = True
        chosen_embedding = get_embedding(" ".join(tokenized_sentences[chosen[0]])) # embedding of the currently chosen sentence
        # get the similarity between the chosen sentence and each of the selected sentences
        for embedding in sentence_embeddings:
            if 1 - cosine(embedding, chosen_embedding) >= similarity_threshold:
                include_sentence = False
                break
        if include_sentence:
            selected_sentences_indices.append(chosen[0]) # append index of the chosen sentence
            sentence_embeddings.append(chosen_embedding)
            curr_length += chosen[1][0]   # length
    else:
        break
```

Ani Nenkova, 2008

Step 1 Estimate the importance of each content word w_i based on its frequency in the input n_i, $p(w_i) = \frac{n_i}{N}$.

Step 2 For each sentence S_j in the input, estimate its importance based on the words in the sentence $w_i \in S_j$: the weight of the sentence is equal to the average weight of content words appearing in it.

$$Weight(S_j) = \frac{\sum_{w_i \in S_j} p(w_i)}{|w_i \in S_j|}$$

Step 3 Select the sentence with the highest weight.

Step 4 For each maximum noun phrase NP_k in the selected sentence

4.1 For each coreferring noun phrase NP_i, such that $NP_i \equiv NP_k$ from all input documents, compute a weight $Weight(NP_i) = F_{RW}(w_r \in NP_i)$.

4.2 Select the noun phrase with the highest weight and insert it in the sentence in place of the original NP. In case of ties, select the shorter noun phrase.

Step 5 For each content word in the rewritten sentence, update its weight by setting it to 0.

Step 6 If the desired summary length has not been reached, go to step 2.

Ani Nenkova, 2008

Maximum noun phrases:

📘 They are defined in a dependency parse tree as the subtree that has as a root a noun such that there is no other noun on the path between it and the root of the tree. (examples on paper)

📘 By definition, a **maximum NP** includes all nominal and adjectival premodifiers of the head, as well as postmodifiers such as prepositional phrases, appositions, and relative clauses.

Ani Nenkova, 2008

Coreference classes

📘 A coreference class is the class of all maximum noun phrases in the input that refer to the same entity.

📘 Here we make a simplifying assumption: all noun phrases **that have the same noun as a head** belong to the same coreference class.

Note: not able to solve issue related to pronouns

Ani Nenkova, 2008

rules for building coref class:

(1) keys can only be proper nouns (if not, the results can be very misleading)
(2) values must contain at least one uppercase letter.
(3) tf-idf score > the median

spacy usage:

➡ noun_chunks for including premodifiers

➡ token.child.dep_ in {relcl, appos, prep, acl, …} for postmodifiers

Ani Nenkova, 2008

In the original context: content realization is blended into content selection.

Main Adaptation:

✅ content realization happens after content selection.

Ani Nenkova, 2008

rules for replacement:

(1) longer than the original
(2) higher tf-idf
(3) only terms seen first time will be treated
(4) few overlapping tokens

why…

(1) under above rules, only minor changes.
(2) same sentences as previous, readily comparable

Filippova, 2011

Human Evaluation Process

- Subset of evaltest files
- 2 human annotators shown outputs for each docset
    - Blind annotation; anotators are in-group
- Annotators view original documents but not gold standard summaries
- Human annotation has two scores:
    - **Coverage**: how well the summary covers information in the documents; comparable to ROUGE recall
    - **Quality** of the summaries
- Scores averaged and normalized between 0 and 1

D1106-A

A:
Two international fishing industry groups called for limits to tuna catches Wednesday to stop the population of the fish from deteriorating due to overfishing .

The Tokyo-based Organisation for the Promotion of Responsible Tuna Fisheries and the World Tuna Purse Seine Organisation , headquartered in Seoul , adopted a joint statement demanding a moratorium on the use of additional large tuna vessels , officials said .

The statement would be sent to five international tuna conservation bodies , which would hold their first joint meeting in Kobe , Japan , Monday through Friday next week , the officials said .

B:
About 300 representatives from five international conservation bodies and more than 60 countries and regions gathered in the Japanese port city of Kobe on Monday for a five-day meeting to talk about optimal and sustainable ways to strengthen the conservation of tuna as the stocks are rapidly declining due to overfishing.
International fisheries officials are expected to push for a global tracking system that would certify the origin of tuna headed to market at an unprecedented conference that convenes Monday to reverse a sharp decline in tuna catches.

C:
The statement would be sent to five international tuna conservation bodies , which would hold their first joint meeting in Kobe , Japan , Monday through Friday next week , the officials said .
Experts also pointed out at the meeting that world demand for tuna has been on a rise since developing countries like China are adding the burden on global tuna stocks .
Representatives of 60 countries or areas are taking part , officials said .
Japan is the world 's biggest tuna consumer , eating one quarter of the global catch .
The country accounts for 12 percent of the global catch of 2.06 million tons .

D:
International groups gather in japan to talk about tuna conservation
The tuna talks will bring the five regional fishery management organizations
The meeting in the Japanese city of Kobe will look at ways to share information to monitor tuna numbers said
The statement would be sent to five international tuna conservation bodies
About 300 representatives bodies gathered in the Japanese port city of Kobe to talk about to strengthen the conservation of
Representatives from the commercial fishing industry environmental groupss were set to discuss ways to strengthen information sharing to track and manage tuna stocks
Attendees will seek the creation of a framework requiring fishermen all species
The plan calls for closer communication in sharing data on Kyodo said

Results

	Baseline	**Peer systems**	**LLR**	**SumBasic**	**LexRank**
ROUGE-1 Recall	32.14	28.88	24.51	28.61	23.11
ROUGE-2 Recall	6.29	5.32	3.91	4.54	4.28
ROUGE-1 Precision	31.10	30.41	33.71	27.65	36.96
Human coverage score	6.21	-	4.67	5.50	5.17
Human quality score	6.54	-	5.58	5.63	4.50

Evaluation - (new) InfoLM

The averages are calculated excluding the outliers, red marks a **decrease in magnitude** compared to the earlier calculation, blue marks **an increasing trend**

Content Selecrtion	**KL Divergence**	**L1**	**L2**	**L_inf**
LLR	-5.251322222 -5.276104444	0.4218152174 0.4212043478	0.07556304348 0.07614347826	0.0288326087 0.03005869565
SumBasic	-5.749272727 -5.852161364	0.435573913 0.43845	0.07690217391 0.07721304348	0.02929565217 0.02958913043
LexRank	-4.08011087 -4.740382609	-0.4206369565 0.4140065217	-0.07503043478 0.0744673913	-0.02848478261 0.02884130435

Evaluation - Pearson Correlation

Content Selection Method	Pearson Correlation Score
LLR	0.1646631006
SumBasic	0.1667934136
LexRank	0.1847522537