Improved extraction and information ordering

Ahmed Aly, Abdelrahman Baligh, Veljko Miljanic

Overview

System architecture overview

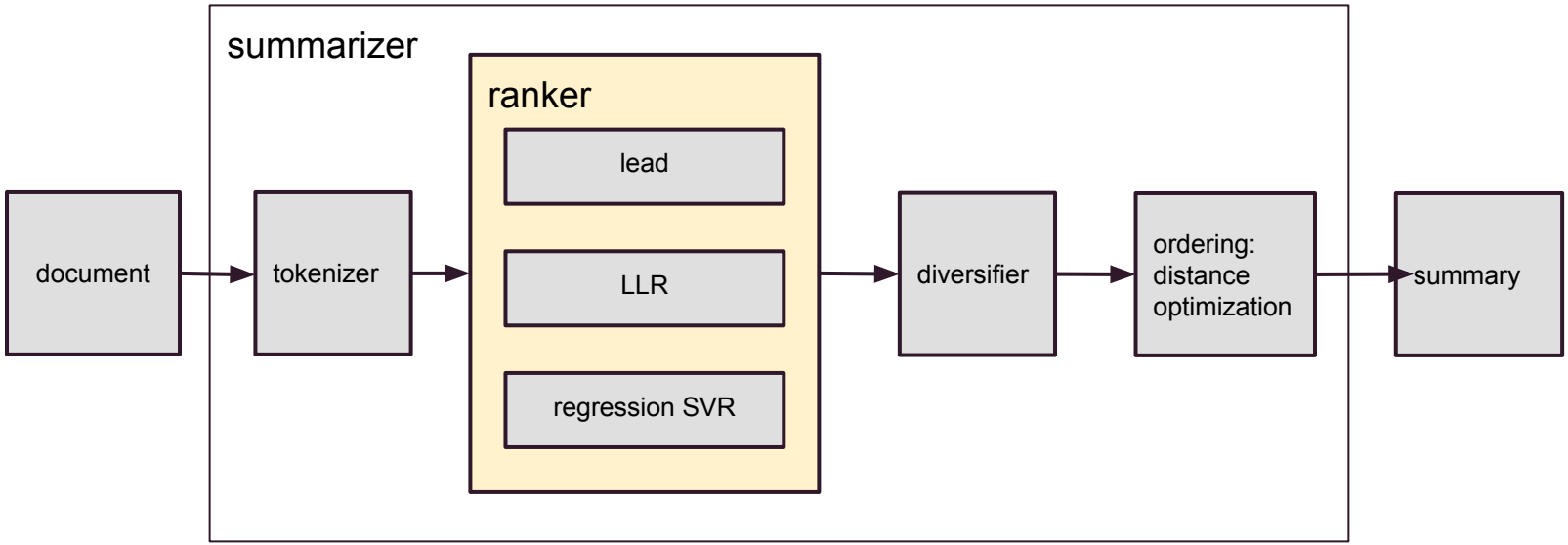
Extraction improvements

- LLR improvement
- Content diversification
- ML Ranker

Information ordering

- COS similarity maximization

System architecture overview



System architecture overview

Content extraction

- Our approach is to solve content extraction as sentence ranking problem
- We want to build ML based ranker that could combine many features to rank sentences
- Baseline systems are Lead and LLR

Ordering

- Maximizing COS similarity between adjacent sentences (TSP)

LLR Improvements

- Added stemming (NLTK, Lancester stemmer)
- Removal of punctuation tokens
- Dynamic LLR threshold selection
  - Idea: adjust threshold for each document
  - Attempt 1: select top N
  - Attempt 2: select top X% of document tokens
  - N / X% are tuned on devtest set
  - Both attempts failed to produce better results :(

LLR Improvements (results)

	ROUGE-1	ROUGE-2	ROUGE-3	ROUGE-4
Lead	0.18369	0.05075	0.01859	0.00666
LLR D2	0.18263	0.04412	0.0155	0.00677
LLR (stem)	0.23349	0.06417	0.02371	0.01011
LLR (stem + punc)	**0.23601**	**0.06504**	**0.02468**	**0.01151**
LLR (stem + punc + topN)	0.23351	0.06303	0.02425	0.0112
LLR (stem + punc + top%)	0.23131	0.06196	0.02401	0.01104

- Best is LLR on stemmed sentences with no punctuation tokens
- Both attempts to set dynamic LLR threshold failed to produce better results than hardcoded threshold

Regression ranker

- Features:
  - f1: LLR
  - Paragraph: f2:paragraph number
  - Sentence: f3: sentence length, f4: quotation
  - Document: f5:sentence position
- Outputs:
  - Sentence ROUGE-1 F score / R score

Regression ranker results

	ROUGE-1	ROUGE-2	ROUGE-3	ROUGE-4
f1	0.24052	0.06836	0.0246	0.01023
f1 + f2	0.23131	0.06419	0.02006	0.00633
f1 + f3	0.23131	0.064	0.02	0.006
f1 + f4	**0.24227**	**0.06998**	**0.02595**	**0.01137**
f1 + f2 + f3	0.17388	0.04194	0.01201	0.0036
f1 + f2 + f5	0.23703	0.06498	0.0228	0.00929

- We got best scores when using the two features f1 and f4 (Sentence LLR scores + Sentence start with quotes)
- Our results have been improved when we used the sentences ROUGE-1 F Scores as the target values for the ranker instead of the ROUGE-1 Recall
- We will work on improving the way we obtain the training targets as it seems to be the main reason why the ranker is not performing as well as we have expected.

Sentence diversification

- Since we're maximizing expectation for ROUGE score, we need to account for shared information between selected sentences.
- We penalize each sentence for redundant information with what's already selected
- As long as we have place in the summary:
  - Take the top sentence
  - For all remaining sentences penalize shared n-grams with selected summary
  - Repeat

Sentence diversification

- N–gram penalization:

$$\sum_{n=1}^{4}[1 - (\sum redundant\ n.grams\ / \sum total\ n.grams\ in\ sentence)^{1/\alpha_n}]$$

- Where $\alpha 1$, $\alpha 2$, $\alpha 3$, $\alpha 4$ are the penalty weights for unigrams, bigrams, trigrams, quadgrams respectively.
- Our experiments suggest that the optimum values for alphas is 0.25 each.

Ordering

Similar to CLASSY 2006:

- Find order that maximizes sum of COS similarities (tf-idf)

Optimization algorithm

1. start with rank order
2. for each sentence i
    a. for each sentence k
        i. swap sentences i and k if it improves the score
3. if score was improved in last iteration
    a. goto 2.
4. done

Results

	ROUGE-1	ROUGE-2	ROUGE-3	ROUGE-4
Lead	0.18369	0.05075	0.01859	0.00666
LLR	0.23601	0.06513	0.02457	0.01151
LLR+diversify	0.24981	0.07208	0.0266	0.01138
SVR	0.23953	0.06562	0.02457	0.01101
SVR+diversify	**0.24227**	**0.06998**	**0.02595**	**0.01137**
*perfect rank	0.27264	0.09265	0.04187	0.02061

*perfect rank is ranker that is using sentence ROUGE scores directly

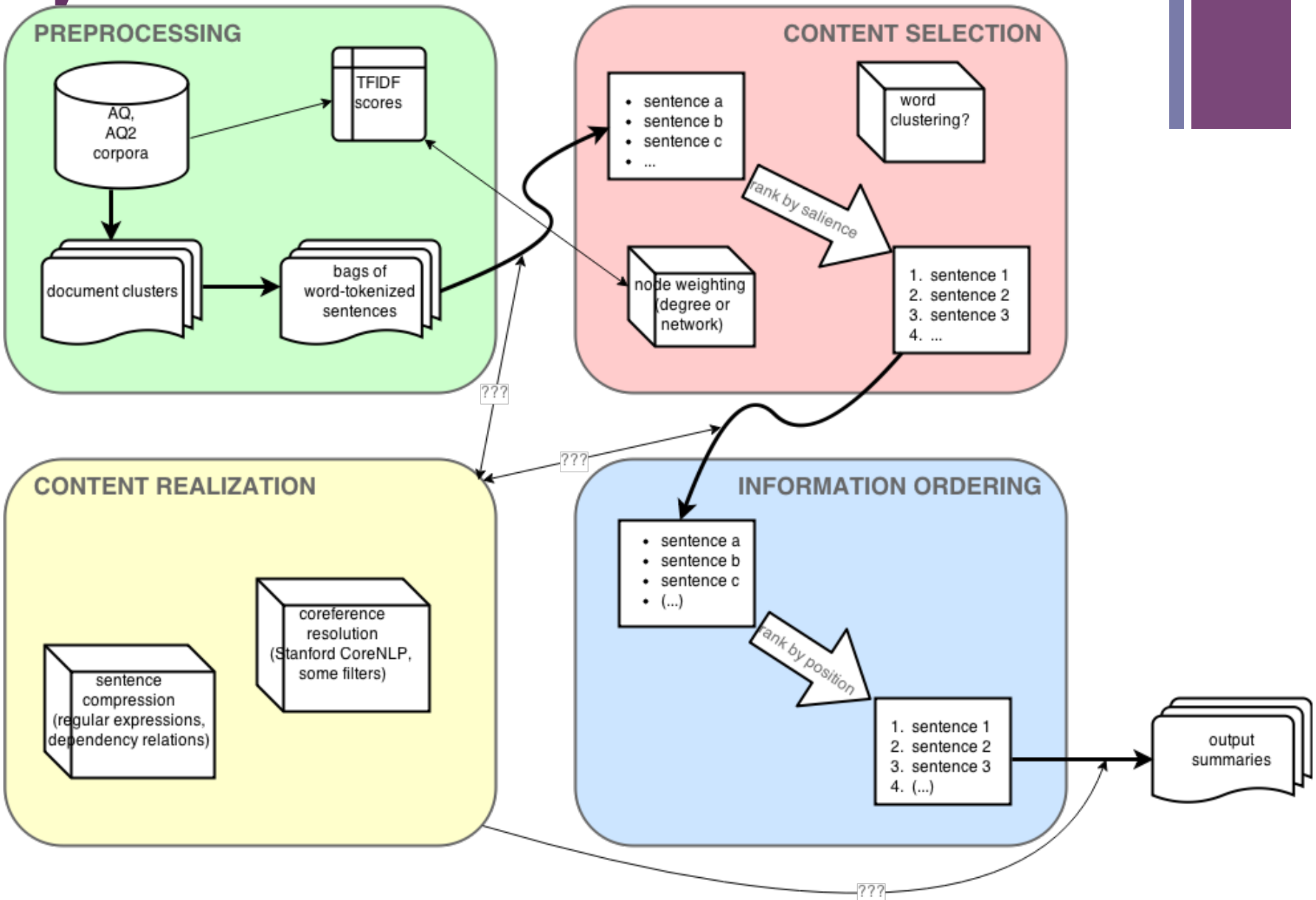
The End

P.A.N.D.A.S.

(Progressive Automatic Natural Document Abbreviation System)

Ceara Chewning, Rebecca Myhre, Katie Vedder

System Architecture



General Improvements

- Improved modularity of overall system
  - Optional components can be turned on or off via command line tags
- IDF scores are collected from entirety of ACQUAINT, ACQUAINT-2 corpora.

Content Selection

Basics

- Graph-based, lexical approach inspired by (Erkan and Radev, 2004)
- IDF-modified cosine similarity equation:

$$sim_{x,y} = \frac{\sum_{w \in x,y} tf_{w,x} tf_{w,y} (idf_w)^2}{\sqrt{\sum_{x_i \in x} (tf_{x_i,x} idf_{x_i})^2} \sqrt{\sum_{y_i \in y} (tf_{y_i,y} idf_{y_i})^2}}$$

  As of D3, IDF scores are collected from entirety of ACQUAINT, ACQUAINT-2 corpora.

- Sentences scored by degree of vertex
- Redundancy accounted for with a second threshold

Failed Attempts: Prestige-Based Node Weighting

- Tried to implement iterative method that weighted node scores based on prestige of adjacent nodes:

$$S_{new}(u) = \frac{d}{N} + (1-d) \sum_{v \in adj(u)} \frac{S_{old}(v)}{deg(v)}$$

- Didn't outperform naïve, degree-based node scoring
- Not included in D3 version of our system

Failed Attempts: Topic Orientation

- For each sentence in the similarity graph, we incremented it's score by an amount proportional to the number of query words the sentence contained.
- Depending on the weighting method, this was done either once, as a reranking step after the degree-based scoring had been assessed, or several times, as a part of the iterative node scoring process.
- None of these approaches improved our ROUGE scores, and topic orientation was not included in the D3 version of our system.

Failed Attempts: Word Sense Clustering

- Wanted to create clusters of words based on the words that co-occur with them in their context window, then use those clusters to have similar words count as one word when measure sentence similarity- i.e.
- Used Word2Vec to make the word vectors and calculate similarity, then sklearn.cluster's Kmeans to do unsupervised clustering over all the words in the document cluster. K = size of vocabulary/ 5
- When calculating new tfidf scores, replace words with their word cluster ID if it exists, and do the same for all documents as the background corpus.

Used this tutorial to lean Word2Vec and Kmeans:
https://www.kaggle.com/c/word2vec-nlp-tutorial/details/part-3-more-fun-with-word-vectors

Failed Attempts:
Pre-Selection Sentence Compression

- We tested performing sentence compression before sentence selection, but this depressed ROUGE scores across the board
  - Sentence compression discussed in detail later

+

Information Ordering

Information Ordering

Sentences are ordered by position of sentence within the original document:

$$pos(s) = \frac{I(\text{sentences in which } s \text{ occurs})}{C(\text{sentences in document})}$$

Information Ordering: A Cherry-Picked Example

BEFORE ORDERING

"Theo didn't want any police protection," of van Gogh in a telephone interview.

Van Gogh received many threats after the film was shown but always laughed them off.

The friends and family of Van Gogh had asked for people to make as much noise as possible in support of the freedom of speech.

Writer-director Theo van Gogh, a descendant of the artist Vincent van Gogh, was attacked shortly before 9 a.m. as he rode his bicycle through Amsterdam's tree-lined streets toward the offices of his production company.

AFTER ORDERING

Writer-director Theo van Gogh, a descendant of the artist Vincent van Gogh, was attacked shortly before 9 a.m. as he rode his bicycle through Amsterdam's tree-lined streets toward the offices of his production company.

The friends and family of Van Gogh had asked for people to make as much noise as possible in support of the freedom of speech.

"Theo didn't want any police protection," of van Gogh in a telephone interview.

Van Gogh received many threats after the film was shown but always laughed them off.

+

Content Realizaton

Content Realization: Sentence Compression

- Goal: to fit more relevant words into the 100-word limit, and reduce the number of redundant or non-information-full words, to hopefully better our topicality judgements

Content Realization: Sentence Compression

- Regular Expression Substitutions
  - Remove parentheses around entire sentences
  - Turn double-backticks (``) into quotes
  - Do more byline reduction (most of which is done in the preprocessing step)
  - Remove non-absolute dates (eg. "last Thursday", "in March")
- Dependency Tree Operations
  - Remove prepositional-phrase asides (prepositional phrases beginning with a comma)
  - Remove beginning-of-sentence adverbs and conjunctions
  - Remove attributives
- Other
  - Cleanup
  - **Replace all contract-able phrases with their contractions (eg. "did not" => "didn't)**

Failed Attempts: Coreference Resolution

- Wanted to replace pronouns with antecedents, so that sentences referring to (but not explicitly containing) topical NPs would be considered for inclusion in summary.
- Used Stanford CoreNLP, which returned a list of abbreviated NPs referring to a more completely expressed entity, each NP's respective location within the document, the fullest form of the NP being referenced, and that NP's location in the document.

  `(3,5,[5,6]) -> (2,3,[1,4]), that is: "his" -> "Sheriff John Stone"`
- Resolved all coreferences within each document before feeding documents into content selector.

Results

	ROUGE-1	ROUGE-2	ROUGE-3	ROUGE-4
Top N	0.21963	0.05173	0.01450	0.00461
Random	0.16282	0.02784	0.00812	0.00334
MEAD	0.22641	0.05966	0.01797	**0.00744**
PANDAS:				
D2	0.24886	0.06636	0.02031	0.00606
D3	**0.24948**	**0.06730**	**0.02084**	0.00662

Next Steps

- Continue to improve sentence compression; use James Clarke's compressed sentence corpus and create a machine-learning model for sentence compression.
- Improve coreference resolution, word clustering to see if it can be made productive.
- Look at topic orientation and network scoring components to see why they might have been unhelpful; improve them if there is time.

Related Reading

Günes Erkan and Dragomir R. Radev. 2004. LexRank: Graph-based Lexical Centrality as Salience in Text Summarization. *Journal of Artificial Intelligence Research*, 22:457–479.

Christopher D. Manning, Mihai Surdeanu ad John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of 52nd Annual Meeting of the As- sociation for Computational Linguistics: System Demonstrations*, pages 55–60.

Automatic Summarization Project

Ling573 - Deliverable 3

Eric Garnick
John T. McCranie
Olga Whelan

Outline

- Information Ordering
- Baseline System Upgrades
- Deadends
- New Results
- Influences and Inspiration

Information Ordering - D2

- Sentence arrangement follows document order by time stamp
- Intra-document order is disregarded.
- Sentences are unlikely to come from the same source.

summarizer_output.A/D0908B-A

::::::::::::::

With streets filled with traffic and shops operating normally.

King Gyanendra denied his takeover was a coup, although soldiers surrounded the houses of Prime Minister Sher Bahadur Deuba and other government leaders.

The EU has supported multiparty democracy and constitutional monarchy in Nepal.

Pakistani Prime Minister Shaukat Aziz on Thursday had a telephonic conversation with King Gyanendra Bir Bikram Shah Dev of Nepal.

The prime minister conveyed to King Gyanendra that recent developments in Nepal were its internal matter.

King Gyanendra on Tuesday sacked prime minister Sher Bahadur Deuba for the second time in three years and assumed control.

D3: three more strategies to reorder already selected sentences

Ordering Strategy I - majority ordering

- heuristic topological search;
- Sentences are given relative positions.
- Pairs of sentences are weighted by the number of times one appears before the other across articles.

summarizer_output.B/D0908B-A

::::::::::::::

The prime minister conveyed to King Gyanendra that recent developments in Nepal were its internal matter.

The EU has supported multiparty democracy and constitutional monarchy in Nepal.

Pakistani Prime Minister Shaukat Aziz on Thursday had a telephonic conversation with King Gyanendra Bir Bikram Shah Dev of Nepal.

King Gyanendra denied his takeover was a coup, although soldiers surrounded the houses of Prime Minister Sher Bahadur Deuba and other government leaders.

With streets filled with traffic and shops operating normally.

King Gyanendra on Tuesday sacked prime minister Sher Bahadur Deuba for the second time in three years and assumed control.

Ordering Strategy II - chronological ordering

- Sentences are ordered by publication date of the article and their relative position in that file.
- The dates are extracted from file names.

summarizer_output.C/D0908B-A
::::::::::::::
The prime minister conveyed to King Gyanendra that recent developments in Nepal were its internal matter.
The EU has supported multiparty democracy and constitutional monarchy in Nepal.
Pakistani Prime Minister Shaukat Aziz on Thursday had a telephonic conversation with King Gyanendra Bir Bikram Shah Dev of Nepal.
King Gyanendra denied his takeover was a coup, although soldiers surrounded the houses of Prime Minister Sher Bahadur Deuba and other government leaders.
King Gyanendra on Tuesday sacked prime minister Sher Bahadur Deuba for the second time in three years and assumed control.
With streets filled with traffic and shops operating normally.

Also attempted augmented chronological ordering - dividing input articles into topic-based blocks and ordering themes within them.

Ordering Strategy III - similarity-based

- Traveling salesman problem
- Find possible permutations of all sentence pairs
- Ordering minimizes the sum of cosine distances between each pair of adjacent sentences.

summarizer_output.D/D0908B-A

::::::::::::::

The prime minister conveyed to King Gyanendra that recent developments in Nepal were its internal matter.

With streets filled with traffic and shops operating normally.

King Gyanendra denied his takeover was a coup, although soldiers surrounded the houses of Prime Minister Sher Bahadur Deuba and other government leaders.

The EU has supported multiparty democracy and constitutional monarchy in Nepal.

Pakistani Prime Minister Shaukat Aziz on Thursday had a telephonic conversation with King Gyanendra Bir Bikram Shah Dev of Nepal.

King Gyanendra on Tuesday sacked prime minister Sher Bahadur Deuba for the second time in three years and assumed control.

Improvements to Baseline

- Tuned parameters
- Focused summaries with topic signatures
- Refined sentence filtering process

Topic Signature Creation

- Combine document cluster titles, article headlines.
  - At least 7 headlines present per 10-doc cluster
  - Remove stop words
  - Calculate LLR for remaining words, keep words scoring > 10
- Result: 9-25 topic signature terms:

Title: Recall Food United States
Signature terms: thorn, deserves, meat, tainted, borne, heinz, ark, recall, listeria, beef, baffle, distributor, recalls, batches, bacteria, listeriosis, food, poisoning

Topic Signature Comparison

- Compare using tf-idf cosine similarity between candidate sentence and topic signature
  - Use all sentences in the document cluster for idf, candidate sentence and topic signature for tf
  - Scores mostly fell between 0.002 - 0.5
  - Augment original sentence weight:

    $w_i = w_i \times (1 + \log(s_i \times 1000))$ if $s_i \geq 0.001$
    otherwise w_i is unchanged
  - ROUGE-1 scores increased ~2%

Parameter Tuning

- Keep alphanumeric tokenization
  - Purely alphabetic lowers scores
- Split on unwanted symbols
  - e.g. hyphenated-word -> hyphenated word
- Bonus for ideal sentence length
  - Best score from 14-token sentences
  - Double scores for ideal length, smaller boost for shorter or longer

$w_i = w_i \times (1 + 1.5^{-diff})$

$diff$ = | length(sentence) – ideal length |

Sentence Filtering

Continued pruning extraneous words relying on regular expressions and word lists. Now removing:

- attributions marked by reporting verbs (’says’, ’said’, ’told’);
- some adverbials and conjunctions that begin sentences (‘and’, ‘in fact’);
- sentence-initial dates (’January 2’);
- all-caps lead-in lines (‘NEWYORK TIMES-COLO DENV’) making sure not to remove useful abbreviations.

More ungrammatical sentences, but good overall:

Original sent: The EU has consistently supported multiparty democracy and constitutional monarchy in Nepal," it said.

Cut down: The EU has supported multiparty democracy and constitutional monarchy in Nepal.

Deadends

- Coreference Resolution
  - coreNLP coref: misaligned tokenization with current system, poor results for coreference chains
- Sentence position weighting
  - Increased weight for 1st, 1st *n* equally, 1st *n* decreasing, last - all lowered ROUGE scores.
- Stemming/Lemmatizing
  - NLTK WordNetLemmatizer + PorterStemmer: overall negligible change in scores.

Results

Original system (D-2):

System	Precision	Recall	F
ROUGE-1	0.25619	0.26501	0.25988
ROUGE-2	0.07232	0.07484	0.07338
ROUGE-3	0.02379	0.02491	0.02428
ROUGE-4	0.00962	0.01029	0.00993

Improved parameter tuning and topic focus:

System	Precision	Recall	F
ROUGE-1	0.28497	0.28448	0.28422
ROUGE-2	0.08476	0.08417	0.08432
ROUGE-3	0.02928	0.02925	0.02923
ROUGE-4	0.01244	0.01246	0.01244

Influences and Inspiration

- chronological, augmented chronological, majority ordering: **Barzilay et al., 2008**
- travelling salesman problem, bolder rules for sentence trimming: **Conroy et al., 2006**

-- -- -- -- -- -- -- --

Future work:

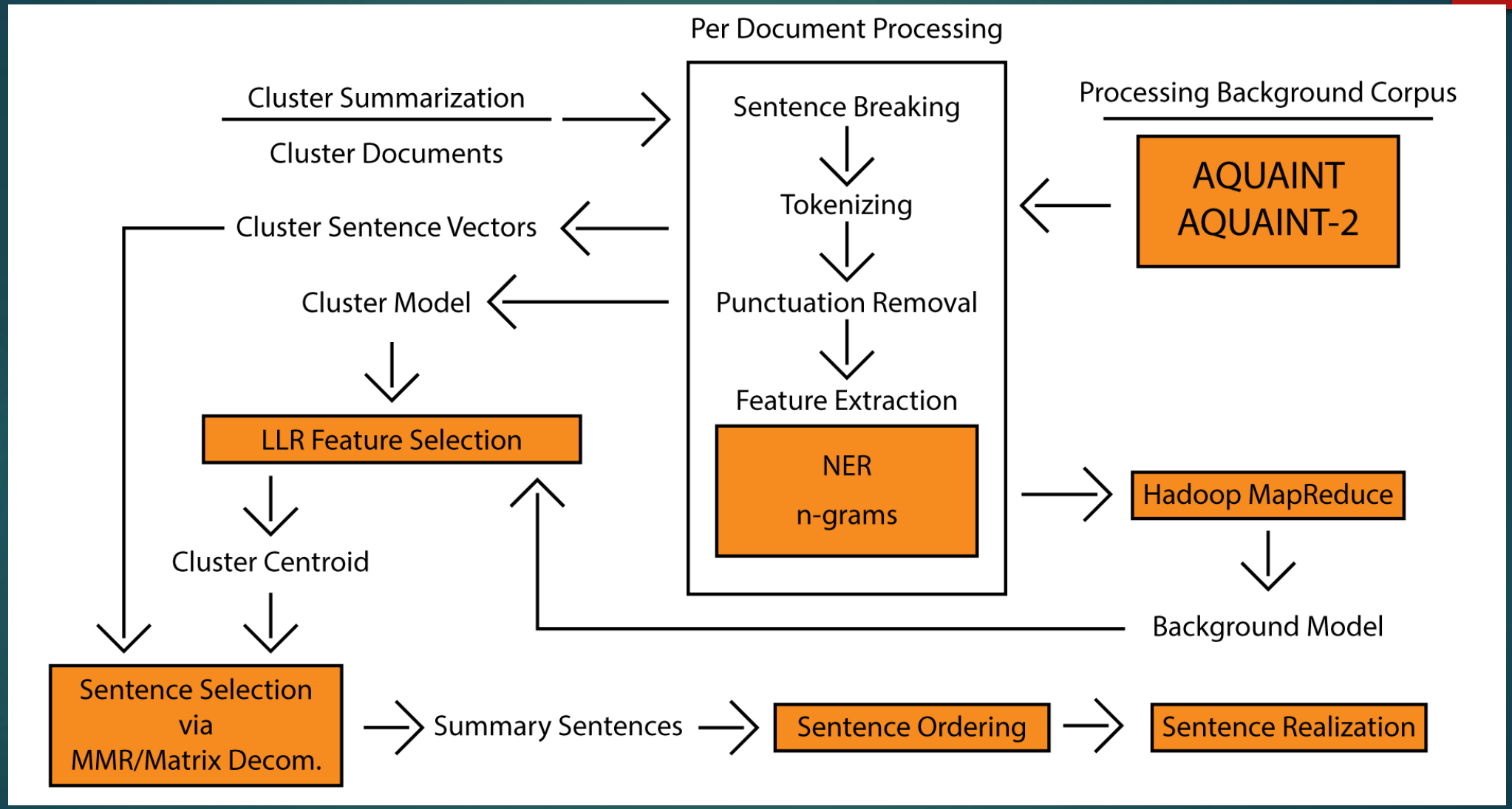
- improve content selection / information ordering
- formal evaluation of information ordering
- develop content realization?

Updated System

RUTH MORRISON

ANDREW BAER

FLORIAN BRAUN


Per Document Processing
Cluster Summarization
Cluster Documents
Sentence Breaking
Tokenizing
Punctuation Removal
Feature Extraction
NER
n-grams
Processing Background Corpus
AQUAINT
AQUAINT-2
Cluster Sentence Vectors
Cluster Model
LLR Feature Selection
Cluster Centroid
Hadoop MapReduce
Background Model
Sentence Selection via MMR/Matrix Decom.
Summary Sentences
Sentence Ordering
Sentence Realization

Information Ordering

- Inspired by entity coherence based ordering
  - Barzilay and Lapata, 2005
- Simplified version with no SVM learning
- Integrated with chronological ordering
- Originally planned to find coreference chains referring to the same reference in different documents in the same cluster, and go with the idea that sentences that contained many references to the same things would be more likely to occur near each other.
  - Used named entities to identify entities instead, because we already had NER implemented.

Information Ordering

- Current algorithm picks the first sentence of the first document as the first sentence of the summary.
- Subsequent sentences are given a score equal to the reverse index of the sentence when all were ordered chronologically added to twice the number of shared entities between the sentence and the last sentence added to the summary
  - 'reverse index' example: with four sentences, the chronologically first scores 3, the next one 2, and so on.
- The sentence with the greatest score is added to the summary.
- The process repeats until there is only one sentence left to add.

Improvements

- Changed from Matrix Redundancy Reduction to Maximal Marginal Relevance (MMR) reduction.
  - MMR applies a penalty to the score of a sentence as factor of it and the most similar sentence to it that is already in the summary.
  - In our use we chose the highest scoring sentence to be the first sentence of the summary
  - From there, the next sentence chosen for the summary is the one from the remaining sentences that maximizes the MMR score until we reach the summary length limit.
  - We found the best delta value to be 0.7

Improvements

- MMR lead to an improvement in ROUGE scores when combined with unigram features.
- Matrix reduction had the best scores when using trigram features, but it still scored worse then MMR on unigrams.
- We attempted to add NER to the features, but the improvement was statistically insignificant.
  - P value of .997
- Vector sparsity has remained a problem throughout the project, and MMR deals with this far better than Matrix Reduction.

	ROUGE-1	ROUGE-2	ROUGE-3	ROUGE-4
Trigrams	**0.23115**	**0.06297**	**0.02200**	**0.00900**
Unigrams	0.21506	0.05213	0.01481	0.00481
Named Entities	0.22417	0.05498	0.01585	0.00453
Trigrams +Unigrams	0.21547	0.05345	0.01578	0.00543
Trigrams+NE	0.22972	0.06087	0.02064	0.00727
Unigrams+NE	0.21655	0.05244	0.01508	0.00491
All	0.21725	0.05270	0.01607	0.00527

ROUGE Scores with Matrix Reduction on 2010 Data

	ROUGE-1	ROUGE-2	ROUGE-3	ROUGE-4
Unigrams	**0.27583**	**0.07720**	**0.02544**	**0.00912**
Trigrams	0.23294	0.06437	0.02239	0.00823
Named Entities	0.22686	0.05900	0.01722	0.00536
Trigrams +Unigrams	0.25538	0.07086	0.02465	0.00952
Trigrams+NE	0.23800	0.06572	0.02139	0.00640
Unigrams+NE	0.27081	0.07304	0.02322	0.00798
All	0.25032	0.06943	0.02351	0.00779

ROUGE scores with MMR on 2010 data

Issues and Successes

- MMR was certainly a success. We saw an improvement in score due to it.
- The information ordering as it was implemented is functional, though it is not what we initially set out to do.
  - Difficulty getting the Stanford Coref system to ‘play nice’ with Python
  - Several other Coreference systems that didn’t work in the right way or work at all
  - All this created time constraints.

Further Reading

- Regina Barzilay and Mirella Lapata. 2005. Modeling local coherence: An entity-based approach. In Proceedings of the 43rd Annual Meeting of the ACL, pages 141–148. Association for Computational Linguistics.
  - Inspired the information ordering approach
- Jaime Carbonell and Jade Goldstein. 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries
  - Inspired the MMR redundancy reduction approach