D4: Final Summary

Selection, Ordering, and Realization

Brandon Gahler
Mike Roylance
Thomas Marsh

Architecture: Technologies

Python 2.7.9 for all coding tasks

NLTK for tokenization, chunking and sentence segmentation.

pyrouge for evaluation

textrazor for entity extraction

attensity for entity and semantic information extraction

Stanford Parser for sentence compression

svmlight for training our ranking classifier

Architecture: Implementation

Reader - Extracts data from topic-focused document clusters

Document And Entity Cache - Entities, Sentences, Semantic Information

Extraction Clusterer - Ranks best sentences for output

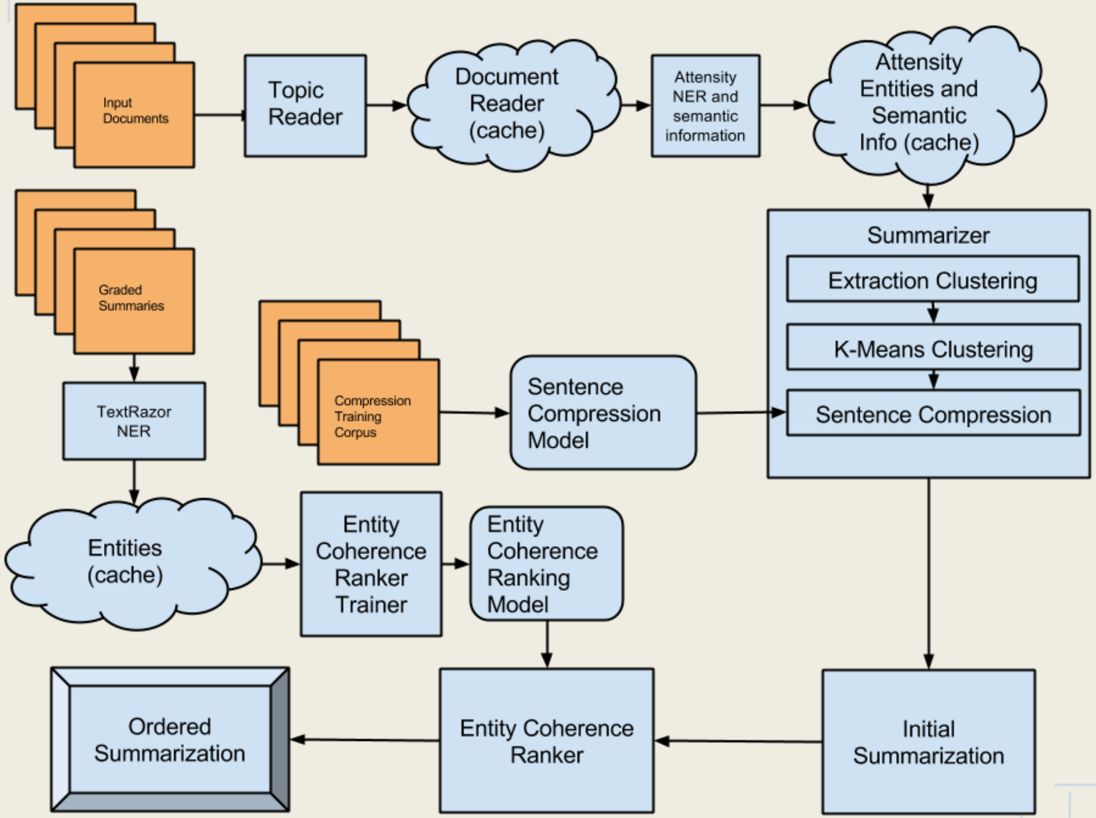
K-Means Clustering - Redundancy Reduction

Compressor - Compresses top sentences inline

Reorderer - Uses entity-coherence ranking to reorder

Evaluator - Uses pyrouge to call ROUGE-1.5.5.pl

Architecture: Block Diagram



Summarizer

Disabling Summary Technique Weighting/Voting Strategy:

Though we have a strong intuition that our technique weighting/voting scheme would eventually bear fruit, we continued to see little evidence for this. The empirical weight generator always appeared to select a single technique at 1.0 and others at 0.0. Because of this, we disabled this mechanism for this deliverable to reduce complexity. We were very sad about this, and hope to resurrect it in the future when we have time to examine what we may have done wrong.

We used the **Extraction Clustering** technique for our single selection strategy.

Extraction Clustering

- Different extractions used for comparison
- **Entity** (Named Entity Recognition)
  - Semantic information
    - Text
    - Domain Role (person, location etc)
- **Triple**
  - Subject, Predicate, Object
- **Fact**
  - Case frame building blocks
  - Element and mode
- **Keyword**
  - Root and POS

Extraction Clustering

Algorithm Enhancements:
We made several incremental refinements to our Extraction Clustering technique for this iteration:

- Normal “most important sentence(s)” extraction with score
- Added a new layer, K-Means Clustering to reduce redundancy.
    - Tried from 20-30 clusters
    - Shot for an average of 30-50 “points” per cluster (minimum of 1)
    - Forced to pick 1 sentence from each cluster.
    - Picked the top scored sentences (from Extraction Clustering)
- Explored root bigrams (word and noun) -
    - I loved to visit Essex. (loved->love) (morphology)
    - (I/PRONOUN, love/VERB), (love/VERB, to/INFINITIVE_TO)
    - (to/INFINITIVE_TO, visit/VERB), (visit/VERB, Essex/NOUN)

Extraction Clustering

Peripheral Enhancements:

We also made some peripheral enhancements to help our overall selection performance:

- Fixed a bug where our sentences were a bit too long, causing our reordering mechanism to actually be doing selection, and thereby changing our rouge scores.
- Removed all sentences with quotes.   A pox on quotes. Forever. Amen.
- Finally removed those pesky info media headers once and for all with some awesome regular expression fu.
- Removed all sentences which did not have a verb.
- Normalized for sentence length to “other” compared sentence length

Sentence Compression

Overall Strategy:

Keep/delete sequence labeling with linear-chain CRF

- Linear SVM
- Written News Compression Corpus
- Features:
  - Current word features + 2 previous
  - Feature selection: top 10% chi-squared
  - Word level features
    - within X of start/end of sentence
    - capitalization
    - negation/punctuation/stopword
    - in upper X% of tfidf relative to rest of the sentence
    - stem and suffix

Sentence Compression

- Features:
  - Syntax features
    - Tree depth
    - Within a X phrase
    - 2 immediate parents
    - X from the left within parent phrase
  - Dependency features
    - Dependency tree depth
    - Mother/daughter of a X dependency
- Just before sentences are added to initial summaries (before ordering) we run the sentence through the compressor and output the compressed sentence instead.

Sentence Compression

Results

- 79.4% accuracy w/ word features, 82.7% with syntax and dependency
- Tendency to remove entire sections, rather than individual superfluous words
    - A co-defendant in the O.J. Simpson armed robbery case told a judge Monday he would plead guilty to a felony and testify against Simpson and four others in the hotel room theft of sports collectibles from two memorabilia dealers.
    - If it were fully loaded, the ship's deck would be lower to the water, making it easier for pirates to climb aboard with grappling equipment and ladders, as they do in most hijackings.
- No rouge score improvement
- Not used in final version

Sentence Ordering

Entity-Based Coherence solution similar to Barzilay and Lapata (2005).

- **NER**: We used a named entity recognizer to extract entities to use in the transition grids.
    - Entities were originally extracted via TextRazor https://www.textrazor.com/

Entity Coherence

Improvements:

1. Removed unused entities from transition graph
2. Added Tuning Parameter for entity frequency
3. Trained on graded summaries
4. Greatly improved performance

Ski resort	Command and control	Galtür	Valzur	Avalanche	Snow	United States	Resort	Austria	Germany	Storm	Winter storm	Helicopter	Ski	Switzerland	Gargallen
–	–	–	–	O	–	–	–	X	–	–	S	–	–	–	–
–	–	–	–	X	–	–	–	–	–	–	–	–	–	–	–
–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
–	–	–	–	–	–	S	–	–	–	–	–	O	–	–	–
–	S	–	–	–	–	–	–	–	–	–	–	–	–	–	–
–	–	–	O	S	–	–	–	–	–	–	–	–	–	–	–
–	–	–	–	S	–	–	–	–	–	–	–	–	–	–	X

Entity Coherence

Improvements:

1. **Removed unused entities from transition graph**
2. Added Tuning Parameter for entity frequency
3. Trained on graded summaries
4. Greatly improved performance

Ski resort	Command and control	Galtür	Valzur	Avalanche	Snow	United States	Resort	Austria	Germany	Storm	Winter storm	Helicopter	Ski	Switzerland	Gargallen
–	–	–	–	O	–	–	–	X	–	–	S	–	–	–	–
–	–	–	–	X	–	–	–	–	–	–	–	–	–	–	–
–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
–	–	–	–	–	–	S	–	–	–	–	–	O	–	–	–
–	S	–	–	–	–	–	–	–	–	–	–	–	–	–	–
–	–	–	O	S	–	–	–	–	–	–	–	–	–	–	–
–	–	–	–	S	–	–	–	–	–	–	–	–	–	–	X

Entity Coherence

Improvements:

1. **Removed unused entities from transition graph**
2. Added Tuning Parameter for entity frequency
3. Trained on graded summaries
4. Greatly improved performance

Command and control	Valzur	Avalanche	United States	Austria	Winter Storm	Helicopter	Gargallen
–	–	O	–	X	S	–	–
–	–	X	–	–	–	–	–
–	–	–	–	–	–	–	–
–	–	–	–	–	–	–	–
–	–	–	S	–	–	O	–
S	–	–	–	–	–	–	–
–	O	S	–	–	–	–	–
–	–	S	–	–	–	–	X

Entity Coherence

Improvements:

1. Removed unused entities from transition graph
2. **Added Tuning Parameter for entity frequency**
3. Trained on graded summaries
4. Greatly improved performance

Command and control	Valzur	Avalanche	United States	Austria	Winter Storm	Helicopter	Gargallen
–	–	O	–	X	S	–	–
–	–	X	–	–	–	–	–
–	–	–	–	–	–	–	–
–	–	–	–	–	–	–	–
–	–	–	S	–	–	O	–
S	–	–	–	–	–	–	–
–	O	S	–	–	–	–	–
–	–	S	–	–	–	–	X

Entity Coherence

Improvements:

1. Removed unused entities from transition graph
2. **Added Tuning Parameter for entity frequency**
3. Trained on graded summaries
4. Greatly improved performance

Avalanche

Final Results

Average ROUGE scores for the Devtest Data:

ROUGE Technique	Recall	Precision	F-Score
ROUGE1	0.23577	0.29921	0.26186
ROUGE2	0.07144	0.09095	0.07949
ROUGE3	0.02821	0.03621	0.03151
ROUGE4	0.01271	0.01624	0.01419

Final Results

Average ROUGE scores for the Evaltest Data:

ROUGE Technique	Recall	Precision	F-Score
ROUGE1	0.26140	0.27432	0.26699
ROUGE2	0.06851	0.07162	0.06984
ROUGE3	0.02268	0.02342	0.02298
ROUGE4	0.00950	0.00976	0.00960

Final Results

Change in Average ROUGE scores From D3 to D4 for DevTest Data:

ROUGE Technique	Recall		Precision		F-Score	
ROUGE1	0.23577	-0.02%	0.29921	+21.02%	0.26186	+8.79%
ROUGE2	0.07144	+14.21%	0.09095	+41.07%	0.07949	+25.44%
ROUGE3	0.02821	+41.40%	0.03621	+76.81%	0.03151	+56.14%
ROUGE4	0.01271	+92.87%	0.01624	+141.67%	0.01419	+113.70%

Final Results

Apples to Oranges: D3 Devtest results compared to D4 Evaltest results:

ROUGE Technique	Recall		Precision		F-Score	
ROUGE1	0.26140	+10.85%	0.27432	+10.95%	0.26699	+10.92%
ROUGE2	0.06851	+9.53%	0.07162	+11.09%	0.06984	+10.21%
ROUGE3	0.02268	+13.68%	0.02342	+14.36%	0.02298	+13.88%
ROUGE4	0.00950	+44.16%	0.00976	+45.24%	0.00960	+44.58%

Future Work

1. **Add Coreference Resolution to Entity Coherence:** This is next! We have coref resolution in the project, we just haven't hooked it up to the Entity Coherence feature.

2. **Reenable voting-based technique aggregation** and run machine-learning algorithms to generate the best weights.

3. **Fix some bugs we found.** we found some.

References

Heinzerling, B and Johannsen, A (2014). pyrouge (Version 0.1.2) [Software]. Available from https://github.com/noutenki/pyrouge

Lin, C (2004). ROUGE (Version 1.5.5) [Software]. Available from http://www.berouge.com/Pages/default.aspx

Roylance, M (2015). Attensity ASAS (Version 0.1) [Software]. Available from http://www.attensity.com

Crayston, T (2015). TextRazor (Version 1.0) [Software]. Available from https://www.textrazor.com/

Jaochims, T (2002a). SVMLight (Version 6.02) [Software]. Available from http://svmlight.joachims.org/

Barzilay, R., & Lapata, M. (2008). Modeling local coherence: An entity-based approach. Computational Linguistics, 34(1), 1-34.

Jurafsky, D., & Martin, J. H. (2009). Speech & language processing. Pearson Education India.

Radev, D, et al. (2006). MEAD (Version 3.12) [Software]. Available from http://www.summarization.com/mead/