Sentence Position revisited: A robust light-weight Update Summarization 'baseline' Algorithm

Rahul Katragadda rahul_k@research.iiit.ac.in
Prasad Pingali pvvpr@iiit.ac.in
Vasudeva Varma vv@iiit.ac.in

Language Technologies Research Center
IIIT Hyderabad

Abstract

In this paper, we describe a sentence position based summarizer that is built based on a sentence position policy, created from the evaluation testbed of recent summarization tasks at Document Understanding Conferences (DUC). We show that the summarizer thus built is able to outperform most systems participating in task focused summarization evaluations at Text Analysis Conferences (TAC) 2008. Our experiments also show that such a method would perform better at producing short summaries (upto 100 words) than longer summaries. Further, we discuss the baselines traditionally used for summarization evaluation and suggest the revival of an old baseline to suit the current summarization task at TAC: the Update Summarization task.

1 Introduction

Document summarization received a lot of attention since an early work by Luhn (1958). Statistical information derived from word frequency and distribution was used by the machine to compute a relative measure of significance, first for individual words and then for sentences. Later, Edmundson (1969) introduced four clues for identifying significant words (topics) in a text. Among them *title* and *location* are related to position methods, while the other two are *presence of cue words* and *high frequency content words*. Edmundson assigned positive weights to sentences according to their ordinal position in the text, giving more weight to the first sentence in the first paragraph and last sentence in the last paragraph.

Position of a sentence in a document or the position of a word in a sentence give good clues towards importance of the sentence or word respectively. Such features are called locational features, and a *sentence position* feature deals with presence of key sentences at specific locations in the text. Sentence Position has been well studied in summarization research since its inception, early in Edmundson's work (1969). Earlier, Baxendale (1958) investigated a sample of 200 paragraphs to determine where the important words are most likely to be found. He concluded that in 85% of the paragraphs, the first sentence was a topic sentence and in 7% of the paragraphs, the final one.

Recent advances in machine learning have been adapted to summarization problem through the years and locational features have been consistently used to identify salience of a sentence. Some representative work in 'learning' sentence extraction would include training a binary classifier (Kupiec et al., 1995), training a Markov model (Conroy et al., 2004), training a CRF (Shen et al., 2007), and learning pairwise-ranking of sentences (Toutanova et al., 2007).

In recent years, at the Document Understanding Conferences (DUC[1]), Text Summarization research evolved through task focused evaluations ranging from '*generic single-document summarization*' to '*query-focused multi-document summarization* (QFMDS)'. The QFMDS task models the real-world complex question answering task wherein, given a topic and a set of 25 relevant documents, the

[1] http://duc.nist.gov/

Proceedings of CLIAWS3, Third International Cross Lingual Information Access Workshop, pages 46–52,
Boulder, Colorado, June 2009.

task is to synthesize a fluent, well-organized 250-word summary of the documents that answers the question(s) in the topic statement. Recent focus in the community has been towards *query-focused update-summarization* task at DUC and the Text Analysis Conference (TAC[2]). The update task was to produce short (~100 words) multi-document update summaries of newswire articles under the assumption that the user has already read a set of earlier articles. The purpose of each update summary will be to inform the reader of new information about a particular topic.

The rest of the paper is organized as follows. In Section 2, we describe a Sub-optimal Position Policy (SPP) based on Pyramid Annotated Data, then we derive a simple algorithm for summarization based on the SPP in Section 3, and show evaluation results. Next, in Section 4, we explain the current baselines and evaluation for Multi-Document Summarization and finally in Section 5, we discuss the need for an older baseline in the current context of the short summary task of update summarization.

2 Sub-Optimal Sentence Position Policy

Given a large text collection and a way to approximate the relevance for a reasonably large subset of sentences, we could identify significant positional attributes for the genre of the collection. Our experiments are based on the work described in (Lin and Hovy, 1997), whose experiments using the Ziff-Davis corpus gave great insights on the selective power of the position method.

2.1 Sentence Position Yield and Optimal Position Policy (OPP)

Lin and Hovy (1997) provide an empirical validation for the position hypothesis. They describe a method of deriving an Optimal Position Policy for a collection of texts within a genre, as long as a small set of topic keywords is defined for each text. They defined *sentence yield* (strength of relevance) of a sentence based on the mention of topic keywords in the sentence.

The *positional yield* is defined as the average *sentence yield* for that position in the document. They computed the yield of each sentence position in each document by counting the number of different keywords contained in the respective sentence in each document, and averaging over all documents. An Optimal Position Policy (OPP) is derived based on the decreasing values of *positional yield*.

Their experiments grounded on the assumption that abstract is an ideal representation of central topic(s) of a text. For their evaluations, they used the abstract to compare whether the sentences found based on their Optimal Position Policy are indeed a good selection. They used precision-recall measures to establish those findings.

At our disposal we had data from pyramid evaluations that provided sentences and their mapping to any content units in the gold standard summaries. The annotations in the data provide a unique property that each sentence can derive for itself a score for relevance.

2.2 Documents

There are a wide variety of document types across genre. In our case of newswire collection we have identified two primary types of documents: *small document* and *large document*. This distinction is made based on the total sentences in the document. All documents that have the number of sentences above a threshold should be considered large. We experimented on thresholds varying from 10 to 35 sentences and figured out that documents' distribution into the two categories was acceptable when threshold-ed at 20 sentences. This decision is also well supported by the fact that the last sentences of a document were more important than the others in the middle (Baxendale, 1958).

Sentence Position Yield (SPY) is obtained separately for both types of documents. For a small document, sentence positions have values from 1 through 20. Meanwhile, for a large document we compute SPY for position 1 through 20, then the last 15 sentences labeled 136 through 150 and '*any other sentence*' is labeled 100. It can be seen in figure 3 that sentences that do not come from leading or trailing part of large documents do not contribute much content to the summaries.

[2]http://www.nist.gov/tac/

```
<document name="APW20000824.0204">
<line>
A lawyer who specializes in bankrupting hate groups is going after the Aryan Nations, whose compound in the Idaho
woods has served as a clubhouse for some of America's most violent racists.</line>
<line>
In a lawsuit that goes to trial Monday, attorney Morris Dees of the Southern Poverty Law Center is representing a
mother and son who were attacked by security guards for the white supremacist group.<annotation scu-count="1" sum-
count="8" sums="13,14,15,23,24,29,30,9">
<scu uid="24" label="SPLC takes legal action against civil rights abuses" weight="3"/>
</annotation>
</line>
<line>
The victims are suing the Aryan Nations and founder Richard Butler.<annotation scu-count="0" sum-count="1" sums="2
9"/>
</line>
```

Figure 1: A sample mapping of SCU annotation to source document sentences. An excerpt from mapping of topic D0701A of DUC 2007 QF-MDS task.

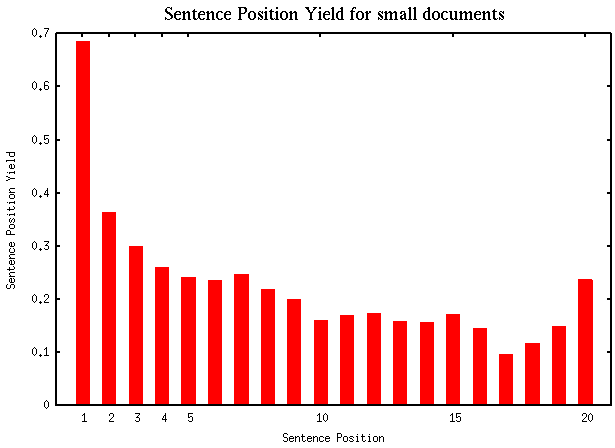

Figure 2: Sentence Position Yield for small documents.

2.3 Pyramid Data

Summary content units, referred as SCUs hereafter, are semantically motivated, sub-sentential units that are variable in length but not bigger than a sentential clause. SCUs emerge from annotation of a collection of human summaries for the same input. They are identified by noting information that is repeated across summaries, whether the repetition is as small as a modifier of a noun phrase or as large as a clause. The weight an SCU obtains is directly proportional to the number of reference summaries that support that piece of information. The evaluation method that is based on overlapping SCUs in human and automatic summaries is described in the Pyramid method (Nenkova et al., 2007).

The University of Ottawa has organized the pyramid annotation data such that for some of the sentences in the original document collection (those that were picked by systems participating in pyramid evaluation), a list of corresponding content units is known (Copeck et al., 2006). We used this data to identify locations in a document from where most sentences were being picked, and which of those locations were being most content responsive to the query.

A sample of SCU mapping is shown in figure 1. Three sentences are seen in the figure among which two have been annotated with system IDs and SCU weights wherever applicable. The first sentence has not been picked by any of the summarizers participating in Pyramid Evaluations, hence it is unknown if the sentence would have contributed to any SCU. The second sentence was picked by 8 summarizers and that sentence contributed to an SCU of weight 3. The third sentence in the example was picked by one summarizer, however, it did not contribute to any SCU. This example shows all the three types of sentences available in the corpus: unknown samples, positive samples and negative samples.

For each SCU, a weight is associated in pyramid annotations. Thus a sentential score could be defined as sum of weights of all the contributing SCUs of the sentence. For an unknown sample and a negative sample, sentential score is 0. For example, in the second sentence in figure 1 the score is 3, contributed by a single SCU. While the same for the first and third sentences is 0.

For each sentence position the sentential score is averaged over all documents, which we call Sentence Position Yield. SPY for small and large documents is shown in figures 2 and 3. Based on these values for various positions, a simple Position Pol-

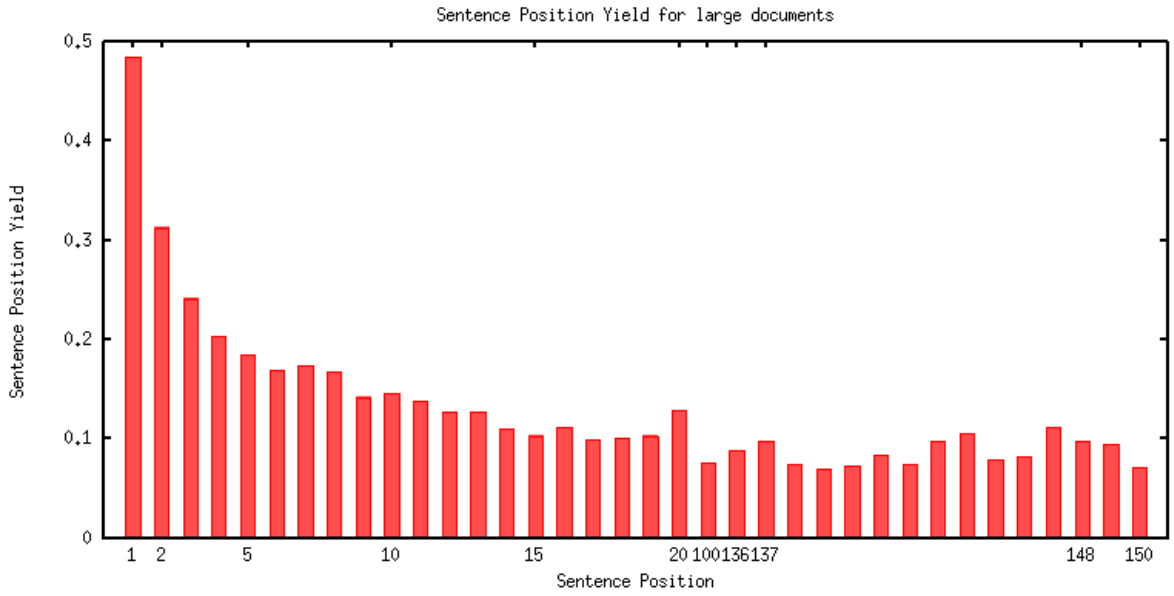

Figure 3: Sentence Position Yield for large documents

icy was framed as shown below. A position policy is an ordered set consisting of elements in the order of most importance. Within a subset, each sub-element is equally important and treated likewise.

$$\{s1, S1, \{s2, S2, s3\}, \{S3, s4, s5, s6, s7, s8, s20\}, \{S4, s9\} \dots\}$$

In the above position policy, sentences from small documents and large documents are represented by *si* and *Sj* respectively.

The position policy described above provides an ordering of ranked sentence positions based on a very accurate 'relevance' annotations on sentences. However, there is a large subset of sentences that are not annotated with either positive or negative relevance judgment. Hence, the policy derived is based on a high-precision low-recall corpus[3] for sentence relevance. If all the sentences were annotated with such judgements, the policy could have been different. For this reason we call the above derived policy, a Sub-optimal Position Policy (SPP).

3 SPP as an algorithm

The goal of creating a position policy was to identify its effectiveness as a summarization algorithm. The above simple heuristic was easily incorporated as an algorithm based on simple scoring for each distinct set in the policy. For instance, based on the policy above, all $s1$ get the highest weight followed by next best weight to all $S1$ and so on.

As it can be observed, only the first sentence of each document could end up comprising the summary. This is okay, till we don't get redundant information in the summary. Hence we also used a simple unigram match based redundancy measure that doesn't allow a sentence if it matches any of the already selected sentences in at least 40% of content words in it. We also dis-allow sentences greater than 25 content words.

We applied the above algorithm to generate multi-document summaries for various tasks. We have applied it to Query-Focused Multi-Document Summarization (QF-MDS) task of DUC 2007 and Query-Focused Update Summarization task of TAC 2008.

3.1 Query-Focused Multi-Document Summarization

The *query-focused multi-document summarization* task at DUC models the real world complex question answering task. Given a topic and a set of 25 relevant documents, this task is to synthesize a fluent, well-organized 250 word summary of the documents that answers the question(s) in the topic state-

[3]DUC 2005 and 2006 data has been used for learning the SPP. In further experiments in section 3, DUC 2007 and TAC 2008 data have been used as test data.

ment/narration.

The summaries from the above algorithm for the QF-MDS were evaluated based on ROUGE metrics (Lin, 2004). The average[4] recall scores are reported for ROUGE-2 and ROUGE-SU4 in Table 1. Also reported are the performance of the top performing system and the official baseline(s). This algorithm performed worse than most systems participating in the task that year and performed better[5] than only the 'first x words' baseline and 3 other systems.

system	ROUGE-2	ROUGE-SU4
'first x words' baseline	0.06039	0.10507
'generic' baseline	0.09382	0.14641
SPP algorithm	***0.06913***	***0.12492***
system 15 (top system)	0.12448	0.17711

Table 1: ROUGE 2, SU4 Recall scores for two baselines, the *SPP algorithm* and a top performing system at Query-Focused Multi-Document Summarization task, DUC 2007.

3.2 Update Summarization Task

The update summarization task is to produce short (~100 words) multi-document update summaries of newswire articles under the assumption that the user has already read a set of earlier articles. The initial document set is called ***cluster A*** and the next set of articles are called ***cluster B***. For cluster A, a query-focused multi-document summary is expected. The purpose of each 'update summary' (summary of *cluster B*) will be to inform the reader of new information about a particular topic. Summaries from the above algorithm for the Query Focused Update Summarization task were evaluated based on ROUGE metrics. This algorithm performed surprisingly better at this task when compared to QF-MDS. The rouge scores suggest that this algorithm is well above the median for cluster A and among the top 5 systems for cluster B.

It must be noted that consistent performance across clusters (both A and B) shows the ***robustness*** of the '*SPP algorithm*' at the update summarization task. Also, it is evident that such an algorithm is computationally simple and ***light-weight***.

These surprisingly high scores on ROUGE metrics prompted us to evaluate the summaries based on Pyramid Evaluation (Nenkova et al., 2007). Pyramid evaluation provides a more semantic approach to evaluation of content based on SCUs as discussed in Section 2.3. The average[6] modified pyramid scores of cluster A and cluster B summaries is shown in Table 2, along with the average recall scores for ROUGE-2, ROUGE-SU4 scores. The pyramid evaluation[7] suggests that this algorithm performs better than all other automated systems at TAC 2008. Table 3 shows the average performance (across clusters) of 'first x words' baseline, SPP algorithm and two top performing systems (System ID=43 and ID=11). System 43 was adjudged best system based on ROUGE metrics, and system 11 was top performer based on pyramid evaluations at TAC 2008.

	ROUGE-2	ROUGE-SU4	pyramid
cluster A	0.08987	0.1213	0.3432
cluster B	0.09319	0.1283	0.3576

Table 2: Cluster wise ROUGE 2, SU4 Recall scores and modified Pyramid Scores for SPP algorithm at the Update Summarization task.

3.3 Discussion

It is interesting to observe that the algorithm that performs very poorly at QF-MDS, does very well in the Update Summarization task. A possible explanation for such behavior could be based on summary length. For a 250 word summary in the QF-MDS task, human summaries might provide a descriptive answer to the query that includes information nuggets accompanied by background information. Indeed, it has been earlier reported that humans appreciate receiving more information than just the answer to the query, whenever possible (Lin et al., 2003; Bosma, 2005).

Whereas, in the case of Update Summarization task the summary length is only 100 words. In such a short length humans need to trade-off between answer sentences and supporting sentences, and usually answers are preferred. And since our method

[4]Averaged over all the 45 topics of DUC 2007 dataset.

[5]Better in a statistical sense, based on 95% confidence intervals of the two systems' evaluation based on ROUGE-2.

[6]Averaged over all the 48 topics of TAC 2008 dataset.

[7]Pyramid Annotation were done by a volunteer who also volunteered for annotations during DUC 2007.

system	**ROUGE-2**	**ROUGE-SU4**	**pyramid**
'first x words' baseline	0.05896	0.09327	0.166
SPP algorithm	***0.09153***	***0.1245***	***0.3504***
System 43 (top in ROUGE)	0.10395	0.13646	0.289
System 11 (top in pyramid)	0.08858	0.12484	0.336

Table 3: Average ROUGE 2, SU4 Recall scores and modified Pyramid Scores for baseline, SPP algorithm and two top performing systems at TAC 2008.

identifies sentences that are known to be contributing towards the needed answers, it performs better at the shorter version of the task.

Another possible explanation is that as a shorter summary length is required, the task of choosing the most important information becomes more difficult and no approach works well consistently. Also, it has often been noted that this baseline is indeed quite strong for this genre, due to the journalistic convention for putting the most important part of an article in the initial paragraphs.

4 Baselines in Summarization Tasks

Over the years, as summarization research followed trends from *generic single-document summarization*, to *generic multi-document summarization*, to *focused multi-document summarization* there were two major baselines that stayed throughout the evaluations. Those two baselines are:

1. First N words of the document (or of the most recent document).
2. First sentence from each document in chronological order until the length requirement is reached.

The first baseline was in place ever since the first evaluation of *generic single document summarization* took place in DUC 2001. For multi-document summarization, first N words of the most recent document (chronologically) was chosen as the baseline 1. In the recent summarization evaluations at Text Analysis Conference (TAC 2008), where update summarization was evaluated; baseline 1 still persists. This baseline performs pretty poorly at content evaluations based on all manual and automatic metrics. However, since it doesn't disturb the original flow and ordering of a document, linguistically these summaries are the best. Indeed it outperforms all the automated systems based on linguistic quality evaluations.

The second baseline had been used occasionally with multi-document summarization from 2001 to 2004 with both generic multi-document summarization and focused multi-document summarization. In 2001 only one system significantly outperformed the baseline 2 (Nenkova, 2005). In 2003 QF-MDS however, only one system outperformed the baseline 2 above, while in 2004 at the same task, no system significantly outperforms the baseline. This baseline as can be seen, over the years has been pretty much untouched by systems based on content evaluation. However, the linguistic aspects of summary quality would be compromised in such a summary.

Currently, for the Update Summarization task at TAC 2008, NIST's baseline is the baseline 1 ('first x words' baseline). And all systems (except one) perform better than the baseline in all forms of content evaluation. Since the task is to generate 100 word summaries (short summaries), based on past experiences, there is no doubt that baseline 2 would perform well.

It is interesting to observe that baseline 2 is a close approximation to the 'SPP algorithm' described in this paper. There are two main differences that we draw between 'baseline 2' and SPP algorithm. First, 'baseline 2' picks only the first sentence in each document, while 'SPP algorithm' could pick other sentences in an order described by the position policy. Second, 'baseline 2' puts no restriction on redundancy, thus due to journalistic conventions entire summary might be comprised of the same 'information nuggets', wasting the minimal real-estate available (~100 words). On the other hand, in our 'SPP algorithm' we consider a simple unigram-overlap measure to identify redundant information in sentence pairs that avoids redundant nuggets in the final summary.

5 Discussion and Conclusion

Baselines 1 and 2 mentioned above, could together act as a balancing mechanism to compare for linguistic quality and responsive content in the summary. The availability of a stronger content responsive summary as a baseline would enable steady progress in the field. While all the linguistically motivated systems would compare themselves with baseline 1, the summary content motivated systems would compare with the stronger baseline 2 and get better than it.

Over the years to come, the usage of 'baseline 1' doesn't help in understanding whether there has been significant improvement in the field. This is because almost every simple algorithm beats the baseline performance. Having a better baseline, like the one based on the position hypothesis, would raise the bar for systems participating in coming years, and tracking progress of the field over the years is easier.

In this paper, we derived a method to identify a 'sub-optimal position policy' based on pyramid annotation data, that were previously unavailable. We also distinguish small and large documents to obtain the position policy. We described the Sub-optimal Sentence Position Policy (SPP) based on pyramid annotation data and implemented the SPP as an algorithm to show that a position policy thus formed is a good representative of the genre and thus performs way above median performance. We further describe the baselines used in summarization evaluation and discuss the need to bring back baseline 2 (or the 'SPP algorithm') as an official baseline for *update summarization* task.

Ultimately, as Lin and Hovy (1997) suggest, the position method can only take us certain distance. It has a limited power of resolution (the sentence) and its limited method of identification (the position in a text). Which is why we intend to use it as a baseline. Currently, as we can see the algorithm generates a *generic* summary, it doesn't consider the topic or query to generate a *query-focused* summary. In future we plan to extend the SPP algorithm with some basic method for bringing in relevance.

References

P. B. Baxendale. 1958. Machine-made index for technical literature – an experiment. *IBM Journal of Research and Development*, 2(Non-topical Issue).

Wauter Bosma. 2005. Extending answers using discourse structures. In Horacio Saggion and J. L. Minel, editors, *RANLP workshop on Crossing Barriers in Text summarization Research*, pages 2–9. Incoma Ltd.

John M. Conroy, Judith D. Schlesinger, Jade Goldstein, and Dianne P. O'leary. 2004. Left-brain/right-brain multi-document summarization. In *the proceedings of Document Understanding Conference (DUC) 2004*.

Terry Copeck, D Inkpen, Anna Kazantseva, A Kennedy, D Kipp, Vivi Nastase, and Stan Szpakowicz. 2006. Leveraging duc. In *proceedings of DUC 2006.*

H. P. Edmundson. 1969. New methods in automatic extracting. In *Journal of the ACM*, volume 16, pages 264–285. ACM.

Julian Kupiec, Jan Pedersen, and Francine Chen. 1995. A trainable document summarizer. In *the proceedings of ACM SIGIR'95*, pages 68–73. ACM.

Chin-Yew Lin and Eduard Hovy. 1997. Identifying topics by position. In *Proceedings of the fifth conference on Applied natural language processing*, pages 283–290. ACL.

Jimmy Lin, Dennis Quan, Vineet Sinha, Karun Bakshi, David Huynh, Boris Katz, and David R. Karger. 2003. The role of context in question answering systems. In *the proceedings of CHI'04.* ACM.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *the proceedings of ACL Workshop on Text Summarization Branches Out*. ACL.

H.P. Luhn. 1958. The automatic creation of literature abstracts. In *IBM Journal of Research and Development, Vol. 2, No. 2, pp. 159-165, April 1958.*

Ani Nenkova, Rebecca Passonneau, and Kathleen McKeown. 2007. The pyramid method: Incorporating human content selection variation in summarization evaluation. In *ACM Trans. Speech Lang. Process.*, volume 4, New York, NY, USA. ACM.

Ani Nenkova. 2005. Automatic text summarization of newswire: Lessons learned from the document understanding conference. In Manuela M. Veloso and Subbarao Kambhampati, editors, *AAAI*, pages 1436–1441. AAAI Press / The MIT Press.

Dou Shen, Jian-Tao Sun, Hua Li, Qiang Yang, and Zheng Chen. 2007. Document summarization using conditional random fields. In *the proceedings of IJCAI '07.*, pages 2862–2867. IJCAI.

Kristina Toutanova, Chris Brockett, Michael Gamon, Jagadeesh Jagarlamundi, Hisami Suzuki, and Lucy Vanderwende. 2007. The pythy summarization system: Microsoft research at duc 2007. In *the proceedings of Document Understanding Conference 2007.*