Using Unsupervised System with least linguistic features for TAC-AESOP Task

Niraj Kumar
IIIT-Hyderabad, INDIA
Niraj_kumar@research.iiit.ac.in

Kannan Srinathan
IIIT-Hyderabad, INDIA
srinathan@iiit.ac.in

Vasudeva Varma
IIIT_Hyderabad, INDIA
vv@iiit.ac.in

Abstract

We consider AESOP Task as Topic based evaluation of information content. Means at first stage, we identify the topics covered in given model/reference summary and calculate their importance. At the next stage, we calculate the information coverage in test / machine generated summary, w.r.t. every identified topic. We use the local importance of words in calculation of importance of topics. From experiments it is clear that use of different methods for identification of topics and calculation of information coverage in test documents w.r.t. every identified topic, have different effect on the result. It is important to note that our devised system do not require any linguistic support or learning or training in entire execution of the system.

1 Introduction

Evaluation of machine generated summaries has been of importance both in TAC (Text Analysis Conference) and previously DUC (Document Understanding Conference). In this vein, Automatically Evaluating Summaries of Peers (AESOP) task in TAC 2010 focuses on developing automatic metrics to judge summary quality. The main goal of AESOP task is to produce two sets of numeric summary-level scores i.e.

All Peers case: a numeric score for each peer summary, including the model summaries. The "All Peers" case is intended to focus on whether an automatic metric can differentiate between human and automatic summarizers.

No Models case: a numeric score for each peer summary, excluding the model summaries. The "No Models" case is intended to focus on how well an automatic metric can evaluate automatic summaries.

Evaluation Process: Each AESOP run is evaluated for:

1. Correlation with the manual metric.
2. Discriminative Power compared with the manual metric.
3. Readability

1.1 Overview of the System

In section 2, we present some related work, In section 3, we describe the entire framework of the system. Section-4 contains pseudo code for entire system. In Section 5, we present the evaluation score of devised system. In section 6, we present extension of present system with effective improvement of results.

2 Related Work

(Nenkova et al. 2007), Manual pyramid scores and (Lin and Hovy, 2003), automatic ROUGE metric (considers lexical n-grams as the unit for comparing the overlap between summaries) are generally considered as current state-of-the-art techniques.

(Hovy et al. 2005), (Hovy et al. 2006) proposed basic elements based methods (BE, another state-of-the-art technique), which facilitates matching of expressive variants of syntactically well-formed units called Basic Elements (BEs).

The ROUGE/BE toolkit has become the standard automatic method for evaluating the content of machine-generated summaries, but the correlation

of these automatic scores with human evaluation metrics has not always been consistent and tested only for fixed length human and machine generated summaries.
(Donaway et al., 2000) proposed using sentence-rank-based and content-based measures for evaluating extract summaries, and compared these with recall-based evaluation measures.

3 System Description

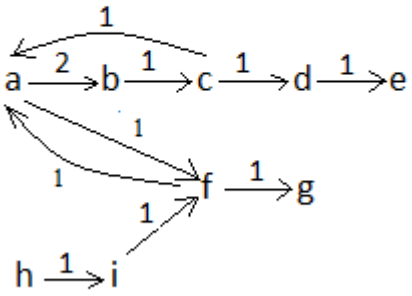
3.1 Input Cleaning

Input cleaning task includes: (1) removal of unnecessary symbols, (2) stemming and (3) sentence filtration. To stem the document we use Porter Stemmer.

3.2 Calculation of Importance of words

To calculate the weight of words, we prepare directed word graph of sentences and then calculate the page rank score of every word. The way to prepare the directed word graph of sentences and calculation of page rank is given below:

Sentences:
S1: a b c d e
S2: c a f g
S3: h i f a b



(a) Directed Word Graph of Sentences

Figure1: directed word graph of sentences, Here S1, S2 and S3 represents the sentences of document and 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h' and 'i' represents the distinct words.

Preparing directed word graph of sentences: For a given set of sentences i.e. S = {S1, S2, ...Sn}, we build a directed word graph by iteratively adding sentences to it. We add a forward directed link for every adjacent word pair of given sentence. See Figure-1. We denote $G=(V,E)$ as a directed graph, Where, $V=\{V_1,V_2,...,V_n\}$ denotes the vertex set and $V \in C$ and link set $(V_j,V_i) \in E$ if there is a link from V_j to V_i.

Calculating Page Rank Score: we use "" to calculate the page rank score for every word. For any given vertex V_i, let $IN(V_i)$ be the set of vertices that point to it (predecessors), and let $OUT(V_i)$ be the set of vertices that vertex V_i points to (successors). Then the page rank score of vertex V_i can be defined as (Page et al., 1998):

$$S(V_i)=\frac{(1-\lambda)}{N}+\lambda \sum_{j \in IN(V_i)} \frac{S(V_j)}{OUT(V_j)} \qquad (1)$$

Where:
$S(V_i)$ = Rank / score of word / vertex V_i.
$S(V_j)$ =rank/score of word/vertex V_j, from which incoming link comes to word / vertex V_i.
N = Count of number of words/vertex in word graph of sentences.
λ = Damping factor (we use a fixed score for damping factor i.e., "0.85" as used in (Page et al. 1998)).

3.3 Identification of Topics covered in Document

To identify topics covered in document, we use sentence community detection scheme (Kumar et al.,2010 (a), (b)), which identifies the concepts / topics in given document. It treats every sentence as node of graph and creates an undirected graph of sentences. To calculate the weight of edge it uses the count of common words between them. The scheme to calculate the weight of edge is given below:

$$W(E_{(S1,S2)})=\frac{1}{count\_common\_word(S1,S2)} \qquad (2)$$

Where
$W(E_{(S1,S2)})$=weight of edges between sentences S1 and S2
$count\_common\_word(S1,S2)$ =count of common words between sentences, S1 and S2.

Finally, it applies the shortest path betweenness strategy, as applied in (Clauset et al., 2004); (Girvan & Newman, 2004) to calculate the sentence community. Here the faster version of community detection algorithm (Clauset et al., 2004) which are optimized for large networks, used. This algorithm iteratively removes edges from the network to split it into communities. The edges removed being identified using graph theoretic measure of edge betweenness. The edge betweenness can be defined as the number of shortest paths between vertex pairs that go along an edge. In entire calculation the modularity score greater than "0.4" is considered.

3.4 Calculating Strength of each Topic

To calculate the weighted importance of any topic or sentence community (Kumar et al.,2010 (a), (b)) we depends upon sum of weighted importance of all words in the given sentence community. The calculation of weighted importance of any community can be given as:

$$W(C) = \sum W \quad (3)$$

Where,

$W(C)$= Weight of given community 'C'.

$\sum W$ = sum of weight of all words of given community.

Next, we calculate the percentage of weighted information of every identified community. The percentage weighted importance of any identified sentence community can be given as:

$$\%W(C) = \left(\frac{W(C)}{\sum W(C)} \times 100 \right) \quad (4)$$

Where,

$\%W(C)$= percentage weight of given community 'C'.

$\sum W(C)$ =sum of weighted importance of all identified communities.

$W(C)$= Weight of given community 'C'.

3.5 Preparation of Evaluation Set

At this stage we prepare evaluation sets. The number of evaluation set depends upon the number of identified communities of reference summary. Every evaluation set contains two sets, i.e. (1) Set-1: contains set of sentences from reference or model summary and (2) Set-2: contains set of sentences from test/machine generated summary which is mostly related to sentences of Set-1.

To prepare evaluation sets we use jacard similarity coefficient.

$$J(A,B) = \frac{(A \cap B)}{(A \cup B)} \quad (5)$$

Where,

"A" represents the sentences of "Set-1" of given evaluation set and "B" represents the sentence from test/machine generated summary.

We calculate the evaluation score of every sentence of test/machine generated summary w.r.t. every identified sentence community. We put the test sentence to Set-2, of evaluation set, for which it gets the highest correlation score.

3.6 Final Evaluation

We apply four different evaluation strategies at this step. The evaluation schemes are given below:

Scheme 1: At this step, we take every evaluation set one by one and check, if it contains uniquely mapped sentence(s) from test / target document then we calculate the score for every such evaluation set. Now we apply following formula to calculate the weighted score in any given evaluation set S_i.

$$Score(S_i) = \left(\frac{\sum Count_{match}(word)}{\sum Count(word)} \times 100 \right) \times \left(\frac{\%W(C)}{100} \right)$$

This implies:

$$Score(S_i) = \left(\frac{\sum Count_{match}(word)}{\sum Count(word)} \right) \times (\%W(C)) \quad (6)$$

Where:

$Score(S_i)$= Evaluation score obtained at evaluation set S_i. This is a percentage score.

$\sum Count_{match}(word)$ =count of co-occurrences of all such words in Set-1 of given evaluation set, (1) which co-occur in both sets i.e. Set-1 and Set-2. Here, we use synonym list to broaden our vision of matching entries.

$\sum Count(word)$ = Count of all words in Set-1 of given evaluation set.

Note: In any given evaluation set, if there does not exist any mapped sentences in Set-2, then we set the evaluation score of that evaluation set to zero. i.e.

$$Score(S_i) = 0; \quad (7)$$

Calculating Final Score: For this we just add the score of all evaluation sets. This can be given as:

$$Final\_Score = \sum_{i=1}^{m} Score(S_i) \quad (8)$$

Where:

$Final\_Score$ =sum of percentage scores obtained from all evaluation sets.

m = Denote the total number of evaluation sets.

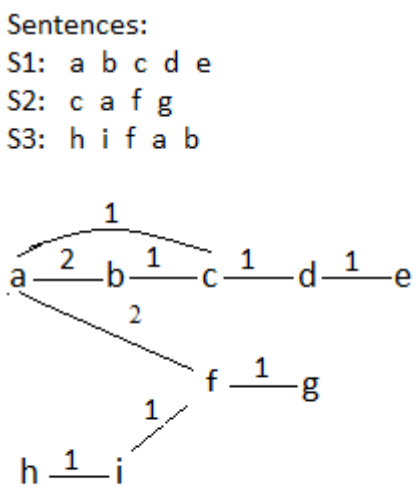
Scheme 2: Scheme 2 is Bigram version of Scheme 1. In this scheme, we applied following changes in "eq-7".

$\sum Count_{match}(word)$ =count of co-occurrences of all bigrams in Set-1 of given evaluation set, (1) which co-occur in both sets i.e. Set-1 and Set-2.

$\sum Count(word)$ = Count of all bigrams in Set-1 of given evaluation set.

Scheme 3: In this scheme we use the concept of shortest path. For this we prepare the undirected word graph of sentences for Set-1 and Set-2 of given evaluation set.

Undirected Word Graph of Sentences: For a given set of sentences i.e. S = {S1, S2, ...Sn}, we build an undirected word graph by adding an undirected link for every adjacent word pair of given sentence. We denote $G = (V, E)$ as an undirected graph, Where, $V = \{V_1, V_2, ..., V_n\}$ denotes the vertex set and link set $(V_j, V_i) \in E$ if there is a link between V_j and V_i.



(a) Undirected Word Graph of Sentences

Figure 2: undirected word graph of sentences, Here S1, S2 and S3 represents the sentences of document and 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h' and 'i' represents the distinct words.

Calculating Shortest path based metric: For this we take evaluation set one by one, prepare separate word graph of sentences for both sets i.e. Set-1 and Set-2. Next, we calculate the shortest path of every co-occurring word from starting word in both sets. Then, we calculate the sum of the differences of shortest path length of all co-occurring words in both sets. Finally we take the reverse of this sum. Here the value will be maximum if the shortest path difference between the co-occurring words is minimum.

Now, Evaluation score at each evaluation set can be given as:

$$Score(SET_i) = \frac{1}{\sum \Delta(SP_{Set-1}, SP_{Set-2})} \quad (9)$$

Where,

$Score(SET_i)$ =Evaluation score of SET_i due to shortest path based metric

$\sum \Delta(SP_{Set-1}, SP_{Set-2})$=sum of the difference of shortest path lengths of co-occurring words.

The Final score can be calculated by adding the sum of scores of all evaluation sets.

$$Final\_Score = \sum Score(SET_i) \quad (10)$$

Scheme 4: In this scheme we use the concept of closeness centrality score.

Closeness centrality: The closeness centrality of any node V_i is defined as the mean geodesic distance (i.e., the shortest path) between a node V_i and all of the nodes reachable from V_i as follows, where n≥2 is the size of the connected component reachable from V_i.

$$C_C(V_i) = \frac{(n-1)}{\sum_{t \in V/V_i} d_G(V_i, t)} \quad (11)$$

Where,

$C_C(V_i)$= closeness centrality of node / vertex V_i

$d_G(V_i, t)$ = sum of geodesic distance from V_i to't'.

Using Closeness centrality based metric: For this we take evaluation set one by one, prepare separate word graph of sentences for both sets i.e. Set-1 and Set-2 (See Undirected word graph of

sentences). Next, we calculate the closeness centrality score of every co-occurring word from starting word in both sets. Then, we calculate the sum of the differences of closeness centrality score of all co-occurring words in both sets. Finally we take the reverse of this sum. Here the value will be maximum if the closeness centrality score difference between the co-occurring words is minimum. The maximum value indicates the maximum similarity between both sets (i.e. Set-1 and Set-2) of given evaluation set.

Now, Evaluation score at each evaluation set can be given as:

$$Score(SET_i) = \frac{1}{\sum \Delta(CC_{Set-1}, CC_{Set-2})} \quad (12)$$

Where,

$Score(SET_i)$ =Evaluation score of SET_i due to Closeness Centrality based metric

$\sum \Delta(CC_{Set-1}, CC_{Set-2})$ =sum of the difference of closeness centrality based scores of co-occurring words.

The Final score can be calculated by adding the sum of scores of all evaluation sets.

$$Final\_Score = \sum Score(SET_i) \quad (13)$$

4 Pseudo Code

The Pseudo code for entire system is given below:

Input: CASE 1: (1) source / model summary, (2) target / machine generated summary, both in ASCII format.
Output: %score, which can be further normalized to "0-1" scale.
Algorithm:

1. Apply input cleaning for source / model summary and target / machine generated summary (See Sub-Section 3.1).
2. Calculate the weight of every word of source / model summary. (See Sub-Section 3.2).
3. Identify the sentence community(s) in source / model document (also addressed as topic(s); see Sub-Section-3.3).
4. Calculate the weighted importance of every identified sentence community (see Sub-Sectionsec-3.4).
5. Prepare separate evaluation set for every identified sentence community of source / model summary by uniquely mapping the sentences from target / machine generated summary (see Sub-Section 3.5).
6. Use all Evaluation sets and apply evaluation scheme to generate the final score (See Sub-Section 3.6).

5 Evaluation

Baselines: In TAC 2011, total three baselines are used:

i. Baseline-1: ROUGE-2, with stemming and keeping stopwords.
ii. Baseline-2: ROUGE-SU4, with stemming and keeping stopwords.
iii. Baseline-3: Basic Elements (BE). Summaries were parsed with Minipar, and BE were extracted and matched using the Head-Modifier criterion.

Evaluation: At this section, we present the evaluation score of (1) All-peers and (2) No Model case. For each automatic metric submitted to the AESOP task, NIST calculated Pearson's, Spearman's, and Kendall's correlations with Pyramid and Overall Responsiveness, as well as the discriminative power of the automatic metric in comparison with these two manual metrics. The results of our system are given in Table-1, Table-2, Table-3, Table-4 and Table-5 shows the performance of our system. In all tables, "OUR SYSTEM -1" shows the result of Scheme-1, "OUR SYSTEM -2" shows the result of Scheme-2, "OUR SYSTEM -3" shows the result of Scheme-3 and "OUR SYSTEM -4" shows the result of Scheme-4 (see Sub-Section 3.6 for all four evaluation schemes).

In the case of discriminative power, our system also got highest Discriminative Power with Automatic-Models

6 Additional Experiments

In section-3.2, we use community detection scheme to identify the sentence community. To improve the accuracy of the system we use Group average agglomerative clustering scheme (GAAC). GACC, uses average similarity across all pairs within the merged cluster to measure the similarity of two clusters. In this scheme average similarity

between two clusters (say, c_i and c_j) can be computed as:

$$sim(c_i, c_j) = \frac{1}{|c_i \cup c_j|(|c_i \cup c_j| - 1)} \sum_{\vec{x} \in (c_i \cup c_j)} \sum_{\vec{y} \in (c_i \cup c_j): \vec{y} \neq \vec{x}} sim(\vec{x}, \vec{y})$$

Among three major agglomerative clustering algorithms, i.e. single-link, complete-link, and average-link clustering. Single-link clustering can lead to elongated clusters. Complete-link clustering is strongly affected by outliers. Average-link clustering is a compromise between the two extremes, which generally avoids both problems. This is the main reason of use of group average agglomerative clustering algorithm for clustering the sentences. In the entire evaluation we use the threshold "0.4".

To apply the GACC on sentences we use a sentence vector representation of entire document. That is, we represent single sentence per line with words as columns. The results with all four schemes are given in Table 1 to 4. We applied this scheme only with our system-4 (see scheme-4 of sub-section 3.6).

7 Conclusion and Future Work

In this paper we, present an unsupervised system, which uses least linguistic information in automatic evaluation of summary.

As future work, we are planning to use better sentence filtration scheme and Wikipedia based knowledge to improve the performance of entire system.

References

Ani Nenkova, Rebecca Passonneau, and Kathleen McKeown. 2007. The pyramid method: Incorporating human content selection variation in summarization evaluation. ACM Trans. Speech Lang. Process., 4(2):4.

Chin-Yew Lin and Eduard Hovy. 2003. Automatic evaluation of summaries using n-gram co-occurance statistics. In Proceedings of HLT-NAACL 2003.

Teufel, S. and H. van Halteren. 2004. Evaluating Information Content by Factoid Analysis: Human Annotation and Stability. Proceedings of the NLP 2004 conference. Barcelona, Spain.

Nenkova, A. and R. Passonneau. 2004. Evaluating Content Selection in Summarization: The Pyramid Method. Proceedings of the HLT-NAACL 2004 conference.

Hovy, E.H., C.Y. Lin, and L. Zhou. 2005. Evaluating DUC 2005 using Basic Elements. Proceedings of DUC-2005 workshop.

Hovy, E.H., C.Y. Lin, L. Zhou, and J. Fukumoto.2006. Automated Summarization Evaluation with Basic Elements. Full paper. Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC 06). Genoa, Italy.

Conroy, J.M. and H.Trang Dang. 2008. Mind the Gap: Dangers of Divorcing Evaluations of Summary Content from Linguistic Quality. Proceedings of the COLING conference. Manchester, UK.

Clauset, A., Newman, M. E. J., Moore, C. (2004). Finding community structure in verylarge networks. Physical Review E, 70:066111.

Newman, M. E. J., Girvan, M. (2004). Finding and evaluating community structure in networks. Physical review E, 69:026113, 2004.

Kumar, N., Srinathan, K. (2008). Automatic Keyphrase Extraction from Scientific Documents Using N-gram Filtration Technique. In the Proceedings of ACM DocEng 2008, ACM 978-1-60558-081-4/08/09.6.

Donaway R, Drummey K, Mather L.A Comparison of Rankings Produced by Summarization Evaluation Measures. In Proceeding of ANLP/NAACL Workshop on Automatic Summarization, pages 69-78, 2000.

Porter Stemming Algorithm for suffix stripping, web –link http://telemat.det.unifi.it/book/2001 /wchange/ download/ stem_porter.html.

Niraj Kumar, Kannan Srinathan and Vasudeva Varma; Evaluating Information Coverage in Machine Generated Summary and Variable Length Documents; COMAD-2010-a, Nagpur, India.

Niraj Kumar, Kannan Srinathan, and Vasudeva Varma;An Effective Approach for AESOP and Guided Summarization Task; TAC-2010-b workshop; (www.nist.gov/tac).

L. Page, S. Brin, R. Motwani and T. Winograd.1998. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.

Table-1: AESOP-ALL Peers (Initial Summary), correlation with Pyramid, Responsiveness, Readability

	Correlation with Pyramid			Correlation with Responsiveness			Correlation with Readability		
System	Pearson	Spearman	Kendall	Pearson	Spearman	Kendall	Pearson	Spearman	Kendall
Baseline-1	0.572	0.864	0.703	0.725	0.779	0.609	0.663	0.498	0.374
Baseline-2	0.763	0.886	0.723	0.733	0.810	0.629	0.682	0.533	0.400
Baseline-3	0.781	0.878	0.720	0.752	0.784	0.590	0.683	0.531	0.387
OUR SYSTEM-1	0.956	0.901	0.760	0.937	0.845	0.678	0.894	0.663	0.490
OUR SYSTEM-2	0.786	0.932	0.799	0.747	0.860	0.693	0.711	0.614	0.460
OUR SYSTEM-3	0.950	0.894	0.752	0.929	0.834	0.665	0.884	0.647	0.479
OUR SYSTEM-4	0.975	0.924	0.776	0.965	0.857	0.674	0.906	0.604	0.446
OUR SYSTEM-4 (with GAAC)	**0.975**	**0.927**	**0.790**	**0.968**	**0.871**	**0.693**	**0.921**	**0.663**	**0.501**

Table-2: AESOP-ALL Peers (Update Summary), correlation with Pyramid and Responsiveness

	Correlation with Pyramid			Correlation with Responsiveness			Correlation with Readability		
System	Pearson	Spearman	Kendall	Pearson	Spearman	Kendall	Pearson	Spearman	Kendall
Baseline-1	0.775	0.851	0.684	0.717	0.869	0.710	0.712	0.550	0.399
Baseline-2	0.730	0.883	0.720	0.675	0.903	0.743	0.686	0.558	0.405
Baseline-3	0.740	0.848	0.686	0.649	0.808	0.637	0.611	0.415	0.287
OUR SYSTEM-1	0.882	0.720	0.546	0.880	0.787	0.596	0.837	0.501	0.367
OUR SYSTEM-2	0.608	0.827	0.654	0.539	0.835	0.666	0.560	0.479	0.353
OUR SYSTEM-3	0.865	0.719	0.545	0.859	0.778	0.589	0.822	0.486	0.359
OUR SYSTEM-4	0.938	0.853	0.676	0.937	0.865	0.694	0.868	0.494	0.360
OUR SYSTEM-4 (with GAAC)	**0.940**	**0.868**	**0.689**	**0.941**	0.898	0.712	**0.881**	0.531	0.380

Table-3: AESOP-NO Models (Initial Summary), correlation with Pyramid and Responsiveness

	Correlation with Pyramid			Correlation with Responsiveness			Correlation with Readability		
System	Pearson	Spearman	Kendall	Pearson	Spearman	Kendall	Pearson	Spearman	Kendall
Baseline-1	0.961	0.894	0.745	0.942	0.790	0.610	0.752	0.398	0.292
Baseline-2	0.981	0.894	0.737	0.954	0.790	0.602	0.784	0.395	0.292
Baseline-3	0.939	0.903	0.746	0.915	0.768	0.567	0.717	0.405	0.291
OUR SYSTEM-1	0.950	0.897	0.755	0.915	0.787	0.610	0.776	0.396	0.290
OUR SYSTEM-2	0.965	0.903	0.758	0.933	0.781	0.596	0.731	0.358	0.242
OUR SYSTEM-3	0.946	0.879	0.725	0.910	0.767	0.591	0.773	0.389	0.282
OUR SYSTEM-4	0.951	0.900	0.753	0.915	0.797	0.620	0.777	0.419	0.304
OUR SYSTEM-4 (with GAAC)	0.971	**0.910**	**0.787**	0.944	**0.810**	**0.641**	**0.799**	**0.423**	**0.314**

Table-4: AESOP-NO Models (Update Summary), correlation with Pyramid and Responsiveness

	Correlation with Pyramid			Correlation with Responsiveness			Correlation with Readability		
System	Pearson	Spearman	Kendall	Pearson	Spearman	Kendall	Pearson	Spearman	Kendall
Baseline-1	0.903	0.802	0.632	0.917	0.840	0.678	0.658	0.373	0.263
Baseline-2	0.885	0.838	0.665	0.912	0.876	0.706	0.672	0.363	0.254
Baseline-3	0.906	0.838	0.684	0.876	0.796	0.625	0.545	0.245	0.162
OUR SYSTEM-1	0.768	0.715	0.550	0.802	0.766	0.586	0.617	0.286	0.203
OUR SYSTEM-2	0.884	0.796	0.628	0.885	0.812	0.638	0.552	0.257	0.180
OUR SYSTEM-3	0.770	0.722	0.549	0.804	0.769	0.590	0.620	0.307	0.210
OUR SYSTEM-4	0.811	0.820	0.639	0.836	0.838	0.662	0.617	0.331	0.248
OUR SYSTEM-4 (with GAAC)	0.905	**0.848**	**0.699**	**0.920**	0.860	0.705	0.659	**0.389**	0.263