Procesamiento del Lenguaje Natural, Revista nº 58, marzo de 2017, pp. 69-76 recibido 30-11-2016 revisado 13-01-2017 aceptado 08-02-2017

# Merging Multiple Features to Evaluate the Content of Text Summary

## *Combinación varias Caracteristicas para evaluar el contenido del resumen de texto*

**Samira Ellouze, Maher Jaoua, Lamia Hadrich Belguith**
ANLP-RG, MIRACL Laboratory, FSEG Sfax, University of Sfax, Sfax, Tunisia
{Samira.Ellouze, maher.jaoua, l.belguith}@fsegs.rnu.tn

**Abstract:** In this paper, we propose a method that evaluates the content of a text summary using a machine learning approach. This method operates by combining multiple features to build models that predict the PYRAMID scores for new summaries. We have tested several single and "Ensemble Learning" classifiers to build the best model. The evaluation of summarization system is made using the average of the scores of summaries that are built from each system. The results show that our method has achieved good performance in predicting the content score for a summary as well as for a summarization system.
**Keywords:** Text summary, content summary evaluation, machine learning

**Resumen:** En este artículo proponemos un método que evalúa el contenido de un resumen de texto utilizando un enfoque de aprendizaje automático. Este método funciona combinando múltiples Caracteristicas para construir modelos que predicen las puntuaciones PYRAMID para nuevos resúmenes. Hemos probado varios clasificadores individuales y "Ensemble Learning" para construir el mejor modelo. La evaluación del sistema de resumen se realiza utilizando el promedio de las puntuaciones de los resúmenes que se construyen a partir de cada sistema. Los resultados muestran que nuestro método ha logrado un buen rendimiento en la predicción de la puntuación de contenido para un resumen, así como para un sistema de resumen.
**Palabras clave:** Resumen del texto, Evaluación de resumen de contenido, aprendizaje automático

## *1 Introduction*

In recent years, several automatic summary systems have been developed. The evaluation of these systems is important to determine their ability to perform the assigned summarization task. It is in this context that several studies have been conducted to develop evaluation metrics which are applicable to manual and/or automatic summarization. However, and in order to have a common data set, several evaluation conferences such as SUMMAC, DUC (Document Understanding Conference), TAC(Text Analysis Conference), etc., were held to evaluate the performance of summaries generated automatically. For instance, the TAC conference adopted three manual measures, namely PYRAMID (content score), readability (linguistic quality) and overall responsiveness (score that reflects both content and linguistic quality of text summary) to assess the quality of text summary. Most metrics developed in the field of automatic evaluation of content summaries address the assessment using a surface analysis (lexical or syntactic) since a deep analysis that affects the syntactic and the semantic level requires meta-knowledge for modeling the contents of text summary. It is in this context that we have targeted as a field of study the evaluation of content summary while trying to address some aspects of syntactic and semantic level. So the objective is to build models able to predict manual content metric by combining automatic metrics and features defined on the candidate summary(CS). The choice of combining these features as a strategy has a number of advantages. For instance, one can benefit from the use of content features that operate on different levels of analysis. The combination of features is performed using algorithms based

ISSN 1135-5948

on regression techniques. The remainder of this article is structured as follows. In Section 2, we give an overview of the principal works that have addressed the problem of content summary evaluation. Then in Section 3, we describe the proposed method which operates by means of machine learning techniques. In Section 4, we give the details of each machine learning step. In Section 5, we present our experiments and the obtained results.

## *2 Previous Works*

The summary evaluation task started as a manual and time-consuming evaluation. One of the famous metrics of content summary evaluation is PYRAMID (Nenkova and Passonneau, 2004) which is based on identifying the common ideas between a candidate summary and one or several reference summaries. These ideas are represented as semantic information units called "Semantic Content Units (SCUs)". Because of the time required to evaluate summaries with manual metrics, many studies are conducted to find ways to automatically assess the content of the summary. One of the standards in automatic evaluation is ROUGE (Lin, 2004). It measures overlapping content between a candidate summary and reference summaries. ROUGE metric scores are obtained through the comparison of common words: N-grams. In order to circumvent the limitations of ROUGE metric (Hovy et al., 2006) proposed a new metric called BE (Basic Elements) which is based on the decomposition of each sentence in minimum semantic units called "Basic Elements"(BE). This metric calculates the overlap between a candidate summary and reference summaries using BE units. Later, Giannakopoulos et al. (2008) introduced AutoSummENG metric, which is based on statistical extracting of textual information from the summary. The information extracted from the summary, represents a set of relations between n-grams in this summary. The n-grams and the relations are represented as a graph where the nodes are the N-grams and the edges represent the relations between them. The calculation of the similarity is performed by comparing the graphs of the candidate summary with the graph of each reference summary. Afterwards, the SIMetrix measurement was developed by (Louis and Nenkova, 2013); it assesses a candidate summary by comparing it with the source documents. The SIMetrix computes ten measures of similarity based on the comparison between the source documents and the candidate summary. Among the used similarity measures we cite the cosine similarity, the divergence of Jensen-Shannon(JS), etc. Recently, Cohan (2016) have developed the SERA (Summarization Evaluation by Relevance Analysis) metric, which is designed to evaluate scientific articles. This metric relies on relevant content in common between a candidate summary and reference summaries. Cohan (2016) use an information retrieval based method which treats summaries as search queries and then measures the overlap of the retrieved results.

## *3 The Proposed Method*

The basic idea of the proposed evaluation methodology is based on the prediction of the manual score PYRAMID for a candidate summary. This prediction is obtained by the extraction of features from the candidate summary itself, from comparing the candidate summary with the source documents or with reference summaries. The choice of the prediction of PYRAMID score is motivated by its importance on the one hand and their availability in the manual evaluations of the DUC and TAC evaluation conferences, on the other hand. Since PYRAMID is based on the manual extraction of SCUs by human judges, SCUs cannot be identified from a summary that does not have a good linguistic quality. Thus, it is interesting to include linguistic features to ensure a better prediction of the PYRAMID score.To get the best prediction model, we tried to combine the relevant traits by using multiple regression-based algorithms. In the next section, we will detail the machine learning phase, which represents the mainstay of the proposed method.

## *4 Machine learning phase*

### 4.1 Features extraction

This first step identifies for each summary the values of all the features. In order to calculate some features related to linguistic quality, we have to use various natural language processing tools such as the Stanford parser (Klein and Manning, 2003), the Stanford Tagger (Toutanova et al., 2003), the Stanford NER (Finkel, Grenager, and Manning, 2005), the Stanford Coref (Lee et al., 2011), the srilm toolkit (Stolcke, 2002), etc. In this

phase we use some new features and other features that are successfully used in the assessment of content. For the linguistic features that have been used, we have tried to cover many linguistic aspects (e.g. grammaticality, non-redundancy, Structure and coherence, etc). In this work, we have included all the classes of features that were used in (Ellouze, Jaoua, and Hadrich Belguith, 2013) and (Ellouze, Jaoua, and Hadrich Belguith, 2016): traditional readability measure features, shallow features, language modeling features, part-of-speech(POS) features, syntactic features, Named Entity based features, local coherence features, ROUGE/BE scores, AutoSummENG scores and Adapted ROUGE scores. Table 1 and Table 2 gives respectively the list of content and of linguistic quality features used in (Ellouze, Jaoua, and Hadrich Belguith, 2013) and in (Ellouze, Jaoua, and Hadrich Belguith, 2016). Furthermore, we have added the features cited subsequently.

#### 4.1.1 Shallow features

We have added to the shallow features cited in Table 3 a set of lexical diversity features which are based on Type/token ratio where tokens refer to the number of words in a summary and types refer to the number of distinct words in a summary. A high score of these features can ensure that the sentences of a summary are less repetitive and have a rich vocabulary. In addition, we have determined for each candidate summary (CS) features based on paragraph length since a short paragraph can be more easily understood and can have fewer problems of co-referencing. Table 3 gives the list of added features.

#### 4.1.2 Part-of-Speech features

We have added same POS features which are related to nouns and verbs which are the most important and essential part of content words for a text summary. This is because a summary must content less description details (i.e., less adjectives and adverbs) and more important actions expressed by nouns and verbs. The added features which are calculated for a CS are cited in the Table 4.

#### 4.1.3 SIMetrix scores features

We have used all the ten scores calculated by SIMetrix (Louis and Nenkova, 2013) such as the Kullback-Leibler (KL) divergence (Kullback and Leibler, 1951) between the source documents (SDs) and the CS (KLInputSummary), the KL divergence between the CS and the SDs (KLSummaryInput), the unsmoothed version of Jensen Shannon divergence (Lin, 1991) between the SDs and the CS (unsmoothedJSD) and the smoothed one (smoothedJSD), the cosine similarity between the SDs and the CS (cosineAllWords), the percentage of the descriptive words of the SDs that appear in the CS (percentTopicTokens), the percentage of the CS composed of the more descriptive words from the SDs. (fractionTopicWords"fTW"), the cosine similarity between the CS and the most descriptive words in the SDs (topicWordOverlap), the probability of uni-grams of the CS given SDs (unigramProb), multinomial probability of the CS given SDs (multinomialProb).

#### 4.1.4 Coreference Features

We have used the "Stanford Coref"(Lee et al., 2011) to allows us identify the different co-reference relations in a summary and the sentences where the co-reference and its antecedent are. From those pieces of information, we have extracted the number of times a pronoun has no antecedent (CorefWithoutAnt), the number of times a pronoun has antecedent (corefWithAnt), whether its antecedent is in the current sentence (AntSameSent), in the previous sentence (AntPrevSent)or not in the same sentence or in the previous sentence (AntOtherSent). In addition, we have determined the ratio between the number of co-references without antecedent to the total number of co-references with antecedent (RatWithAntWithoutAnt) and vice versa (RatWithoutAntWithAnt), the number of pronouns without antecedent to the total number of words (RatWithoutAntNbWord) and the number of pronouns without antecedent to the total number of pronouns (RatWithoutAntNbPron).

#### 4.1.5 Redundancy features

To calculate these features, we compared each sentence in the CS with the other sentences by using a lexical similarity measure. For each measure of similarity, the average similarity between sentences and the average maximum(Max) similarities between each sentence and other sentences of the CS were determined. The following features are calculated for each CS: AVG and Max redundancy with DICE coefficient (RedondAVGdice, RedondMaxDice), with overlap coefficient (RedondAVGover, RedondMaxO-

| Feature | Description |
|---|---|
| ROUGE-1, ROUGE-2, ROUGE-3, ROUGE-4, ROUGE-SU4, ROUGE-L and ROUGE-W | ROUGE score based on respectively uni-grams, bi-grams, tri-grams, four grams, skip-bigrams and uni-grams, Longest Common Subsequence of n-grams and Weighted Longest Common Subsequence of n-grams |
| BE | Score based on syntactic units called BE (Basic Elements) |
| ROUGE-$1_{Ad}$, ROUGE-$2_{Ad}$, ROUGE-$3_{Ad}$, ROUGE-$4_{Ad}$, ROUGE-$5_{Ad}$, ROUGE-$L_{Ad}$, ROUGE-$S4_{Ad}$ and ROUGE-$W_{Ad}$ | ROUGE adapted score based on respectively uni-grams, bi-grams, tri-grams, four grams, five grams, Longest Common Subsequence of n-grams, skip-bigrams and Weighted Longest Common Subsequence of n-grams |
| AutoSummENG-$W_{123}$, AutoSummENG-$W_{333}$, AutoSummENG-$W_{253}$, AutoSummENG-$C_{123}$ | AutoSummENG with n-grams of words of length between respectively [1..2], [3..3], [2..5] and of characters of length between [1..2] with window size of 3 for all used variants |

Table 1: List of content features used previously

| Feature | Description |
|---|---|
| NbDET, NbCC, NbPSC, NbPRP, NbN, NbV, NbADJ and NbADV | Number(NB) of respectively determinants, coordinating conjunctions, prepositions and subordinating conjunctions, personal pronouns, nouns, verbs, adjectives (ADJ) and adverbs(ADV) |
| AVgDET, AVgCC, AVgPSC, AVgPRP, AVgADJ, AVgV, AVgN, AVgADV | Average(AVG) NB of determinants, coordinating conjunctions, prepositions and subordinating conjunctions, personal pronouns, adjectives, verbs, nouns and adverbs per sentence |
| Dens_DET, Dens_CC, Dens_PSC, Dens_PRP, Dens_ADJ, Dens_V, Dens_N and Dens_ADV | Density of respectively determinants, coordinating conjunctions, prepositions and subordinating conjunctions, personal pronouns, adjectives, verbs, nouns and adverbs |
| FleschK_Ind, FleschR_Ind, Aut_Read_Ind and Gun_Fog_Ind | Readability measures of respectively Flesch-Kincaid Index, Flesch Reading Ease, Automated Readability Index and Gunning Fog Index |
| AVGSyllWord, AVGCarWord, | AVG NB of respectively syllables, characters per word |
| AVGWordSent | AVG NB of words per sentence |
| RatWordMaxWord | Ratio between CS size and maximum size allowed by TAC campaign |
| logSent, logCar, logWord | Logarithm of the NB of respectively sentences, characters and words |
| AvgNPsent, AvgVPsent, AvgPPsent and AvgSBARsent | AVG NB of respectively noun phrases, verb phrases, prepositional phrases and clauses per sentences |
| NbNP, NbVP, NbPP and NbSBAR | NB of respectively noun phrases, verb phrases, prepositional phrases and clauses |
| AVG_Height_PT | AVG height of the parse tree |
| AVG_NB_dep_sent | AVG NB of dependency relations by sentence |
| logProbUnigram, logProbBigram, logProbTrigram | Log probability of respectively uni-grams, bi-grams and tri-grams of the CS |
| pplUnigram, pplBigram, pplTrigram | Measure of perplexity for respectively unigrams, bi-grams and tri-grams normalized by the NB of uni-grams, bi-grams and tri-grams |
| ppl1Unigram, ppl1Bigram and ppl1Trigram | Measure of perplexity for uni-grams, bi-grams and tri-grams with exclusion of the sentence end tags |
| NbEnt, DensEnt and AvgEntSent | NB, Density and AVG of Named entities in the candidate summary |
| AVGLevenDist, AVGCosSim, AVGJacSim, AVGJSDiver, AvgKLdiv, AVGPearCor, AVGdiceInd, AVGoverlapCoef | the AVG of respectively Levenshtein distance, cosine similarity, Jaccard distance, divergence of JS, Kullback-Leibler divergence , Pearson correlation, Dice index and overlap coefficient between adjacent sentences |

Table 2: List of linguistic quality features used previously

| Feature | Description |
|---|---|
| Nb_DistWord | NB of distinct words |
| TTR | Type/token ratio |
| Rac_Dens_DistW | Root of the density of distinct words |
| Dens_Corr_DistW | Correct density of distinct words |
| Bilog_Dens_DistW | Bi-logarithmic density of distinct words |
| Uber_Index | Uber index |
| AVGSentParag | AVG NB of sentences per paragraph |
| AVGWordParag | AVG NB of words per paragraph |
| Dens_stopWords | Density of stop words |

Table 3: List of added Shallow features

ver), with Jaccard index (RedondAVGjacc, RedondMaxJacc) and with cosine similarity (RedondAVGcos, RedondMaxCos).

The content features cited in Table 1 and the SIMetrix scores have proved their usefulness in the field of text summary evaluation (Lin, 2004), (Hovy et al., 2006), (Giannakopoulos et al., 2008) and (Louis and Nenkova, 2013). In addition, most linguistic quality features cited previously have shown their utility in the assessment of the content (Ellouze, Jaoua, and Hadrich Belguith, 2013) and the linguistic quality (Ellouze, Jaoua, and Hadrich Belguith, 2016), (Pitler, Louis, and Nenkova, 2010), etc. While for some other features we have tried to test their performance (non-redundancy, coreference, etc).

## 4.2 Selection of relevant features

This step allows us to select the most relevant features that must be kept for the training step. In general, the selection of relevant features is as important as the choice of the learning algorithm. To select the relevant features, we use the "wrapper"method (Kohavi

| Feature | Description |
|---|---|
| Dens_V_N, Rat_N_V and AVG_N_V | Density, Ratio and AVG of verbs and nouns |
| Rat_NV_AdjAdv | Ratio between the NB of nouns and verbs and the NB of ADJs and ADVs |
| Rat_InfV_V, Rat_ImpV_V, Rat_PartV_V and Rat_ModV_V | Ratio between the NB of respectively infinitive, imperative, participle and model verbs, and the total NB of verbs |

Table 4: List of Added POS Features

and John, 1997) which is based on the evaluation of subsets of features which allows to detect the possible interactions between features. After training models using each subset, the best subset of features is retained. Using the "wrapper"method, we have obtained the relevant features for the best predictive model, in each evaluation task.

### 4.3 Training and Validation of the Predictive Model

This step helps to build and validate the predictive model of the PYRAMID score. To build the predictive model, we have used several basic algorithms (single algorithms), implemented by the Weka environment (Witten, Frank, and Hall, 2011), using a regression method such as "GaussianProcesses". Moreover, we tried to produce models by using the "ensemble learning" which usually produces more accurate solutions than a basic learning algorithm. In our experiment, we use three "ensemble learning" algorithms which are implemented in the Weka environment:

- "Bagging" (Breiman, 1996) divides the training data into separate samples. Then it creates a model for each sample with the same algorithm. Next, it aggregates the generated models using averaging or majority voting
- "Vote"(Kuncheva, 2004) allows the combination of several predictive models trained on the same dataset using a combination rule like "Majority Voting".
- "Stacking"(Wolpert, 1992) combines several models (made from different basic learning algorithm) that are learned from a classification or a regression task using the same dataset. The combination of the constructed models is made using a machine learning algorithm.

After testing the algorithms, we adopt the one that produces the best predictive model. The validation of each model is performed by cross-validation method with 10 folds.

## *5 Experimentations*

We experimented our method for summary level evaluation on initial summary task (task A) and update summary task (task B) by trying to predict PYRAMID scores. On the system level, we will just average the predicted scores of all the candidate summaries produced by the same summarization system.

### 5.1 Data Set

The Data Set used in the study consists of the source documents, the manual summaries (reference summaries) and the system summaries presented in the TAC 2008 conference on the update summarization task. This task includes two subtasks, initial summary task and update summary task. In initial summary task, each summarization system had to summarize a set of documents (A) which deals with a particular event. Then, in update summary task, it should summarize a set (B) of documents which addresses the evolution of the same event and considers the knowledge of the set (A). This corpus includes 48 collections, each collection contains a set (A) and a set (B) of documents. Moreover, it includes 2784 (58*48) system summaries that are automatically generated from the set (A) of the 48 collections and by the 58 participating systems, in initial summary task. And 2784 system summaries in update summary task. The corpus also includes reference summaries produced manually by 8 human summarizers. For each collection, 4 reference summaries are produced for set (A) and 4 reference summaries are produced for set (B). In total, 384 (96 * 4) reference summaries. Thus, each system summary can be assessed by comparing the four reference summaries. Similarly, a reference summary can be evaluated by comparing it with the other three reference summaries. Furthermore, the corpus contains the PYRAMID and the linguistic quality of each reference and system summary. The linguistic quality score is an integer between 1 and 5 which reflects five linguistic qualities. In our experiments in summary level evaluation, each model is pro-

| Features | Initial Summary |
|---|---|
| Content | ROUGE-1, ROUGE-2, ROUGE-3 ROUGE-4, ROUGE-SU4, ROUGE-W, AutoSummENG_w333, AutoSummENG_w123, AutoSummENG_w253 KLInputSummary, KLSummaryInput, unsmoothedJSD, smoothedJSD, cosineAllWords, percentTopicTokens, fractionTopicWords, TopicWordOverlap, unigramProb, multinomialProb |
| Linguistic quality | NbDET, NbPSC, Dens_DET, Dens_N, Dens_V_N, Uber_Index, AvgSBARsent, AvgPPsent, NB_SBAR,AVG_Height_PT, AVG_NB_dep_sent, logProbUnigram, logProbBigram, NbEnt, AvgKLdiv, AntPrevSent, RatWithoutAntNbWord, RedondAVGdice |

Table 5: List of Selected Features to Predict Content Score for initial summary task

duced using 2976 CSs where 2784 are system summaries and 192 are reference summaries.

## 5.2 Evaluation

### 5.2.1 Summary level

In this subsection, we begin by citing in Table 5 the selected features for the prediction of content score in initial summary task. From Table 5, we remark the selection of most content scores in addition to many linguistic quality features. We have observed the presence of features related to reference clarity and redundancy (AntPrevSent, RedondAVGdice). This means that when evaluating the content, we need to have a candidate summary with clear reference resolution and without redundancy. In addition, we remark the presence of Language modeling features which can be indicators of the fluency and the grammaticality (logProBigram) of a text summary. Now, we give in Table 6 the list of used features in update summary task. From this table, we remark that also in update summary task (task B), many linguistic quality features are selected as relevant ones. Besides, the importance and the necessity of including linguistic quality features is clearly shown in update summary by the use of features related to diverse aspects of linguistic quality like referential clarity (RatWithoutAntNbSent), non-redundancy (RedondMaxDice, RedondMaxOver), etc. We examine the usefulness of the selected features in the prediction of the content score by training them using single and "ensemble learning". The Pearson's correlation (Pearson, 1895) and the RMSE generated by each classifier are presented in Table 7. In fact, the RMSE (Root Mean Square Error) is a measure that determines the differences between score values predicted by a model and the actual score values (in our case PYRAMID manual score). Table 7 shows the performance of the selected features in building models using several single and ensemble of classifiers in the initial and update summary tasks. In the initial summary task, the results show that the model built from the classifier "GaussianProcesses" produced the best correlation(0.769) and the lowest RMSE(0.1189). In the update summary level, Table 7 indicates that the best "ensemble learning" classifier is the "Vote" which provides a model having a correlation of 0.8063 and an RMSE of 0.1128. Another notable observation is that the correlation in the update summary task is more important than the one in the initial summary task.

We pass now to the comparison between the performance of the best obtained model and the baseline metrics that were adopted by the TAC conference as baseline metrics

| Features | Update Summary |
|---|---|
| Content | ROUGE-1 ROUGE-2, ROUGE-4, ROUGE-SU4, ROUGE-L, ROUGE-W, ROUGE-BE, ROUGE-$3_{Ad}$, ROUGE-$4_{Ad}$, ROUGE-$5_{Ad}$, ROUGE-$S4_{Ad}$, AutoSummENG_$W_{123}$, KLInputSummary, KLSummaryInput, unsmoothedJSD, smoothedJSD, cosineAllWords, percentTopicTokens, topicWordOverlap |
| Linguistic quality | NbCC, NbV, AVgPSC, AVgV, Dens_CC, Dens_PRP, Dens_V, Dens_V, Dens_ADV, Dens_V_N, Rat_InfV_V, Rat_ImpV_V, Rat_PartV_V, Rat_ModV_V, AVGSyllWord, AVGCarWord, AVGSentParag, RatWordMaxWord, Dens_DistWord, Rac_Dens_DistW, Bilog_Dens_DistW, logSent, logCar, logWord, AvgNPsent, AvgPPsent, AVGCosSim, AVGJSDiver, AVGdiceInd, RatWithoutAntNbSent, RedondMaxDice, RedondMaxOver, RedondMaxCos, RedondAVGcos |

Table 6: List of Selected Features to Predict Content Score for Update Summary

| Classifiers | Task A | Task B |
|---|---|---|
| **Single classifiers** | | |
| GaussianProcesses | **0.7690(0.1185)** | 0.7965(0.1156) |
| LinearRegression | 0.7421(0.1241) | 0.7416(0.1270) |
| SMOReg | 0.7391(0.1250) | 0.7972(0.1155) |
| MultiPerceptron | 0.7079(0.1311) | 0.7111(0.1370) |
| **"Ensemble learning" classifiers** | | |
| Vote | 0.7470(0.1231) | **0.8063(0.1128)** |
| Bagging | 0.7424(0.1240) | 0.8009(0.1142) |
| Stacking | 0.7453(0.1234) | 0.8052(0.1130) |

Table 7: Pearson Correlation with PYRAMID and RMSE (between brackets) for Various Single and Ensemble learning Classifiers

| Scores | Task A | Task B |
|---|---|---|
| **Baselines** | | |
| ROUGE-2 | 0.5990(0.1482) | 0.5830(0.1548) |
| ROUGE-SU4 | 0.5090 (0.1399) | 0.6205(0.1495) |
| BE | 0.4493(0.1653) | 0.5540(0.1587) |
| AutoSummENG_$W_{123}$ | 0.5405(0.1557) | |
| AutoSummENG_$C_{333}$ | | 0.6487(0.1451) |
| SIMetrix_fTW | 0.3382(0.1742) | 0.3389(0.1793) |
| **Our experimentations** | | |
| Combining ROUGE Scores | 0.6075(0.147) | 0.6440(0.1458) |
| Combining AutoSummENG scores | 0.6841(0.135 ) | 0.6134(0.1505) |
| Combining SIMetrix scores | 0.4648(0.1639) | 0.3594(0.1779) |
| Combining content scores | 0.7330(0.1260) | 0.7570(0.1248) |
| Combining selected features (CSF) | **0.7690(0.1185)** | **0.8063(0.1128)** |
| CSF without ROUGE/BE | 0.759(0.1207) | 0.7797(0.1194) |
| CSF without AutoSummENG | 0.7631(0.1198) | 0.7997(0.1145) |
| CSF without SIMetrix | 0.7414(0.1243) | 0.7919(0.1164) |
| CSF without new features | 0.7532(0.1219) | 0.7510(0.1260) |

Table 8: Pearson Correlation with PYRAMID Score and RMSE (between brackets)for Summary Level

such as R-2, R-SU4 and BE and also we add the best variante of each of the two others famous metrics AutoSummENG and SIMetrix. Table 8 details the different correlations and RMSEs of baseline metrics and our experiments. It should be noted that, in the initial summary task, the models built in our experiments use all the "GaussianProcesses" classifier. In addition, we note that in the update summary task, the models built in our experiments use all the "vote" ensemble learning. From Table 8 and in both tasks, we see the gap between baseline metrics and our experiments, regardless of whether we used the selected features or just content scores. Moreover, we noticed that the inclusion of linguistic quality features in the best model produced improves the performance of this model compared to the model containing just content scores. We note also that the elimination of the new added features in this article, decreases the correlation between the predictive score and PYRAMID score. Furthermore, we find that the elimination of one of the content score classes, reduces the correlation of the predictive score with PYRAMID score.

### 5.2.2 System Level

Remember that the system level evaluation allows us to estimate the quality of a summarization system; in other words, the system assessment is done by taking into account the quality of all the summaries that are produced by this system. In this article, we tried to calculate the quality of a system $Score_{system}$ by determining the average of the predicted score for summaries produced by the same system. To evaluate this method of calculating the content score for a system, we study the correlation of Pearson"P", Spearman"S" (Spearman, 1910) and Kendall"K" (Kendall, 1938) between the PYRAMID score and the $Score_{system}$ score. Indeed, those correlation measures have been used in the DUC and the TAC conferences to determine the correlation between automatic and manual evaluation metrics. Table 9 details the different correlations between the PYRAMID score and the $Score_{system}$ score or the baseline metrics.In this evalution level, we use as baselines, ROUGE-2, ROUGE-SU4, BE, AutoSummENG_$W_{123}$ and SIMetrix_fractionTopic. As can be seen in Table 9, the best correlation is obtained by our $Score_{system}$. It has the best correlation with the PYRAMID score in both tasks and with the three types of correlation measures.

## 6 Conclusions and Future Works

In this paper, we presented a method of content evaluation for text summaries. Our work has been motivated by the lack of efficient and accurate automatic tools that evaluate the content of a summary. The proposed method is based on the construction of models that combine selected features which come from multiple feature classes such as ROUGE scores, SIMetrix scores, modeling language features, Syntactic features, etc. The combination of features is performed by testing many single and "ensemble learning" classifiers. Then, we have selected the best algorithm for the prediction of the PYRAMID score. At the initial summary level and in order to evaluate the predictive power of the model constructed using the selected features to predict content score, we have compared the correlation of this model with baselines and with a model containing only content scores. In both tasks, the obtained results show that there is an important gap between baselines and the model combining selected features. We also note that adding linguistic quality features to a model predicting PYRAMID, improves the results.

In system level evaluation, for a specific task and a predicted content score "$Score_{system}$", we have calculated the average of the predicted score values of all the summaries that were built from the same summarization system. In both tasks, the average

| | P | S | K | P | S | K |
|---|---|---|---|---|---|---|
| **Scores** | **Initial Summary** | | | **Update Summary** | | |
| ROUGE-2 | 0.8718 | 0.9364 | 0.8050 | 0.9009 | 0.9588 | 0.8322 |
| ROUGE-SU4 | 0.8741 | 0.9007 | 0.7477 | 0.8458 | 0.9323 | 0.7796 |
| BE | 0.9188 | 0.9329 | 0.7889 | 0.9188 | 0.9560 | 0.8297 |
| AutoSummENG_W_123 | 0.9051 | 0.9336 | 0.7946 | 0.8955 | 0.9626 | 0.8384 |
| SIMetrix_fTW | 0.5523 | 0.7764 | 0.5922 | 0.4160 | 0.6298 | 0.4570 |
| $Score_{system}$ | **0.9950** | **0.9761** | **0.8901** | **0.9964** | **0.9866** | **0.9204** |

Table 9: Pearson, Spearman and Kendall Correlation with PYRAMID Score on System Level

of the predicted content scores of each system "$Score_{system}$" correlates the best with the PYRAMID score.

As futur work, we project to apply this method of building models to other manual scores like the overall responsiveness score. In addition, we aim to add same features related to semantic level.

## *References*


Breiman, L. 1996. Bagging predictors. *Machine learning*, 24:123–140.

Cohan, A, G. N. 2016. Revisiting summarization evaluation for scientific articles. In *Proceedings of LREC conference*, pages 806–813.

Ellouze, S., M. Jaoua, and L. Hadrich Belguith. 2013. An evaluation summary method based on a combination of content and linguistic metrics. In *Proceedings of RANLP conference*, pages 245–251.

Ellouze, S., M. Jaoua, and L. Hadrich Belguith. 2016. Automatic evaluation of a summary's linguistic quality. In *Proceedings of NLDB 2016 conference*, pages 392–400.

Finkel, J. R., T. Grenager, and C. Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the Annual Meeting on ACL*, pages 363–370.

Giannakopoulos, G., V. Karkaletsis, G. Vouros, and P. Stamatopoulos. 2008. Summarization system evaluation revisited: N-gram graphs. *ACM Trans. Speech Lang. Process.*, 5(3):1–39.

Hovy, E., C. Lin, L. Zhou, and J. Fukumoto. 2006. Automated summarization evaluation with basic elements. In *Proceedings of the LREC conference.*

Kendall, M. G. 1938. A new measure of rank correlation. *Biometrika*, 30:81–89.

Klein, D. and C. D. Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of the Annual Meeting on ACL*, pages 423–430.

Kohavi, R. and G. H. John. 1997. Wrappers for feature subset selection. *Artif. Intell.*, 97(1-2):273–324.

Kullback, S. and R. A. Leibler. 1951. On information and sufficiency. *Ann. Math. Statist.*, 22(1):79–86.

Kuncheva, L. I. 2004. *Combining pattern classifiers: Methods and Algorithms.* Wiley-Interscience.

Lee, H., Y. Peirsman, A. Chang, N. Chambers, M. Surdeanu, and D. Jurafsky. 2011. Stanford's multi-pass sieve coreference resolution system at the conll-2011 shared task. In *Proceedings of CoNLL conference*, pages 28–34.

Lin, C. 2004. Rouge: a package for automatic evaluation of summaries. In *Text Summarization Branches Out: Proceedings of the ACL Workshop*, pages 25–26.

Lin, J. 1991. Divergence measures based on the shannon entropy. *IEEE Trans. Inf. Theory*, 37:145–151.

Louis, A. and A. Nenkova. 2013. Automatically assessing machine summary content without a gold standard. *Computational Linguistics*, 39(2):267–300.

Nenkova, A. and R. J. Passonneau. 2004. Evaluating content selection in summarization: The pyramid method. In *HLT-NAACL*, pages 145–152.

Pearson, K. 1895. Mathematical contributions to the theory of evolution, ii: Skew variation in homogeneous material. *Philosophical Transactions of Royal Society London (A)*, 186:343–414.

Pitler, E., A. Louis, and A. Nenkova. 2010. Automatic evaluation of linguistic quality in multi-document summarization. In *Proceedings of the Annual Meeting of the ACL*, pages 544–554.

Spearman, C. E. 1910. Correlation calculated from faulty data. *British Journal of Psychology*, 3:271–295.

Stolcke, A. 2002. Srilm-an extensible language modeling toolkit. In *Proceedings of ICSLP*, pages 257–286.

Toutanova, K., D. Klein, C. D. Manning, and Y. Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings OF HLT-NAACL*, pages 252–259.

Witten, I. H., E. Frank, and M. A. Hall. 2011. *Data mining: Practical machine learning tools and techniques.* Morgan Kaufmann Publishers Inc.

Wolpert, D. H. 1992. Stacked generalization. *Neural Networks*, 5:241–259.