Saliency Detection in Text Documents using Policy-Driven Reinforcement Learning Methodologies

Gaurav Meena, Sarika Choudhary, Ravi Raj Choudhary
Department of Computer Science, Central University of Rajasthan, Ajmer, Rajasthan, India-305817
E-mail: gaurav.meena@curaj.ac.in, www@curaj.ac.in

Abstract. As the amount of information grows, it is challenging to find concise information. Thus it is necessary to build a system that could present human quality summaries. Saliency detection is a tool that provides abstracts or keywords of a given document. In this paper, three different approaches have been implemented for saliency detection. In all these three approaches, sentences are represented as a feature vector. In the first approach, features like root words, vocabulary intersections, words, and inclusion of numerical data use. This model is trained by using general Algorithms, Like Porter’s Stemmer, Spell check. In the second approach, apart from the features used in the first approach, TF-IDF scores, Mean, Standard Deviation, and a Threshold value of a word is also used as features. In the third approach, Maximal Marginal Relevance (MMR) algorithm is used to generate a summary.

Keywords: Saliency, Summaries, Porter’s Stemmer, Spell check, TF-IDF, Mean, Standard Deviation, Threshold value, Maximal Marginal Relevance (MMR).

1. Introduction
With the happening to the data transformation, electronic reports are turning into a standard media of business and scholarly data. A large number of electronic archives are created and made accessible on the web every day. Be that as it may, it is difficult to peruse every single archive. So there is a need for introducing the data as notable words so that there is no compelling reason to peruse the entire record. An enormous sum of information may be accessible around net-like. Anyhow retrieving the right what more critical information is is not time permits. Search engines, for example, such that Google, Yahoo. Try to serve similarity as a data get agenize. They recover what is more rank the data similarity as for every those user’s appeal. In any case, due to those absences of profound comprehensions of standard dialect also humankind’s intelligence, they best perform shallow string. Saliency identification in content reports is the way toward distinguishing the most critical data from the archive or set of related records and speaking to the principle thought in significantly lesser space, usually by a variable of three to ten than the original content. On a basic level, Saliency location in content archives is conceivable on account of the usually happening repetition in content and because imperative or striking data is spread unevenly in literary reports information. Those data recovery will be at distant more excellent over that client canwood handles what is more oversee. Those clients need to dissect the searched outcomes one by one until the palatable majority of the data may be required.

It is a chance devouring and wasteful, Assuming that report canwood be speaking to for fewer expressions with key vital words, after that, this aggravates our existence more straightforward.
Automatic Text Summarization [1] may be a quick mining undertaking that extracts first sentences with a blanket just about every last one of the main ideas of those archives.

The main benefit of this is that the time consumed by the user to read the whole document to find the useful information reduces. With record outline judgment available, a client might undoubtedly choose it is pertinence on their investment also procure wanted documents for a significant part very less mental load. The main challenge is to identify redundancy.

2. Literature Review

2.1. Pioneering Works:

Luhn et al., 1958 [2] Excerpts of specialized papers that are more magazine articles that fill those needs of regular digests had been made entirely via programmed implies. In the exploratory investigation depicted, the total content of an article to machine-discernable state is examined by an IBM 704 information transforming machine and broke down as per a standard program.

Edmundson et al., 1969 [3] portray new strategies for naturally extricating reports to screening reasons for existing, i.e., the PC choice of sentences hosting the best possibility to passing on to the individual using the substance of the report card. The exploration strategy incorporates strategies to those courses of action of the required lexicons, those sets of the control parameters, and the near assessment of the programmed separates with physically created removes.

Baxendale et al., 1958 in [4] tell about Machine strategies for lessening specialized foul documents on their segregating indices, which investigated. The PC program mimicked human checking designs in choosing "point sentences" and expressions made out of things and modifiers. The measure of buildup coming about because of every technique and the relative consistency in files are analyzed. It demonstrated that the organized file gave by the expression is more significant and segregating.

2.1.1. Statistical Method

Salton et al., 1988 in [5] demonstrated that that pertinence from claiming record terms is conversely proportional to the amount from claiming documents in the corpus holding those expressions. The mathematical statement for term pertinence assessment given by tfi x idfi, where tfi is the recurrence of term i in the report, and idfi is the reversed recurrence of archives containing this term. Sentences might get a chance to be in this way scored, for example, by summing the significance of terms in the sentence.

Kupiec et al., 1995 in [6] said that to condense is to lessen in many-sided quality, and consequently long while holding a portion of the fundamental characteristics of the first. This work principally centered on record extricates, a specific sort of registered archive rundown. Report extricates comprising about 20 percent of the first can be as enlightening as the full content of a record, which recommends that even shorter concentrates might be valuable characteristic synopses.

Eduard Hovy et al., 1999 [7] gave a condition on SUMMARIST, and it is an endeavor to make a robust computerized content synopsis framework, because of the condition: rundown = point distinguishing proof + translation + era. Each of these stages contains a few free modules, vast numbers of them prepared on extensive corpora of content.

3. Methodology and Implementation

Machine learning is [8] may be a method for data examination that mechanizes legitimate model creation. Using calculations that iteratively pick up starting with data, machine learning in permits machines to find shrouded encounters without being unequivocally programmed the place with taking a look. Arthur Samuel (1959). Machine Learning: field of study that provides for Pcs those limits to take without being explicitly programmed.

Tom Mitchell, Well-postured learning in Problem: a workstation system gets similarly as a matter for knowledge (E), concerning a portion undertaking (T), What is more, a percentage
execution measure (P). In its execution for T, Likewise measured Toward P, enhances with knowledge E. Two of the majority comprehensively grasped machine learning in methodologies used as supervised learning and unsupervised learning. In any case, there are also separate strategies for machine learning in the different methods may be reinforcement learning.

3.1. Supervised learning
Algorithms would be readied utilizing denoted cases, for instance, a piece of information where the place the desired yield may be referred. For example, a touch for mechanical assembly might bring information $F$ (Failed) or $R$ (runs).

3.2. Unsupervised learning
Unsupervised learning utilizes information that needs no chronicled denote. Those structures will not be completely frank the “right reply.” Those computations must fathom what is shown up. The objective is to investigate the information also discover a few structures inside. Unsupervised learning in capacities amazingly on regard built information.

3.3. Semi-supervised learning
Semi-supervised learning utilized for vague requisitions from supervised learning. Over whatever case, it employments both checked and unlabelled information to getting to prepared a minimal measure from claiming named information with a tonal of unlabelled information (in light of the reality that unlabelled information will be more sensible furthermore less push will get).

3.4. Reinforcement learning
Reinforcement learning now uses for robotics, gaming, and route. With help taking in, those figuring figures through experimentation which activities yield the best remunerates. This sort of learning has three essential parts: those master (the learner alternately boss), that world (everything those director interfaces with), and exercises (what the driver can do).

3.5. Input Dataset
We input data in the program for processing in the form of XML (Extensible Markup Language) files. From the XML files, we need the only text portion of the file for the further processing of the program. For extracting the text from the XML files, we use the JAXB API of the JAVA. The input dataset considered from the DUC2001 dataset.

3.6. Exclude List
Prepare an Exclude List, which contains all those words which are not available in the dictionary like the name of persons, countries, cities. It is a manually created list.

3.7. Dictionary
We are using a Dictionary from WordNet for the checking of the spellings of words extracted from the input documents.

3.8. Stopwords
Stopwords are expressions that are separated when alternately following preparing information (text). If stop expressions generally allude to the relatable point expressions in a language, there is no single widespread rundown about stop expressions utilized by all tools. In fact, not know devices utilize such a rundown. Any arrange for talk maybe pegged concerning illustration those stop expressions to a habituated plan.
3.9. Tokenizing
Tokenizing is a treat to alter an edict into root words. The sentence segmentation revolves around the composing delimiter, which is space (" "). The scheme in this regard deal with will then be continued with stemming.

3.10. Stemming
Stemming is a process of transforming a word to return it to its root word by applying specific rules so that each word has the same representation. Stemming is in this research uses Porter’s Stemmer Algorithm (1980). This Algorithm deals with confix, prefix, and suffix [9].

3.11. TF-IDF
Tf-idf remains for term frequency-inverse document frequency, and the tf-idf weight is a weight regularly utilized done data recovery and mining. This weight is an accurate measure used to assess how critical a statement will be within a document to an accumulation or corpus. Those significance builds proportionally to structure about times a saying gives the idea in the report; however, is wipe out separately normality of clearing in the corpus. Varieties of the tf-idf weighting plan need aid regularly utilized by search engines as a vital apparatus for scoring and positioning a document’s pertinence provided for a client inquiry.

3.11.1. TF: Term Frequency, which measures how habitually a term happens to a document. Since without a unique case point of interest may be uncommon over a piece, it will be accessible that expression might appear significantly all the more introduce long documents over abbreviated ones. Thus, those expression frequency is much-branched detail limit (those aggregate amount from claiming terms in the document) separately similarly as an approach of normalization:

\[ TF(t) = \frac{\text{Number of times term } t \text{ shows up in an document}}{\text{Total number of terms in the document}} \]

3.11.2. IDF: Inverse Document Frequency, which measures how vital expression is. Same time registering TF, the sum terms need aid viewed as just as imperative. Nonetheless, it may be referred to that certain terms, for example is, of, and that, might show up a considerable measure of times, however, need little significance. Accordingly, we have to weigh down the incessant terms same time scale up those extraordinary ones, by registering those following:

\[ IDF(t) = \log \frac{\text{Total number of documents}}{\text{Number of documents with term } t} \]

TF-IDF Weighting: We now consolidate the definitions of term frequency and inverse document frequency, to prepare a composite weight for each term in each document. That tf-idf weighting plan will term \( t \) a weight to document \( d \) provided for by

\[ tf-idf_{t,d} = tf_{t,d} \times \text{idf}_t \]

Clinched alongside other words, \( tf-idf_{t,d} \) assigns to term \( t \) a weight over document \( d \) that is

- Most elevated the point when \( t \) happens a lot of people times inside a little number of documents (thus loaning helter-skelter segregating control with the individual’s documents);
- Bring down the point when the expression happens fewer times for a document, or happens in large portions documents (thus putting forth a less maintained pertinence signal);
- Least the point when those term happens clinched alongside virtually every last bit of documents.
3.12. Cosine Similarity

Cosine similarity \[10\] is a basic calculation in order to obtain a similarity score between two vectors. Cosine similarity calculation consists of two stages which are:

- Calculating relevance between document and query Calculating Cosines of two vectors which are \( W \) (weight) of every document with \( W \) (weight) of the query.
- Calculating similarity between documents Calculating Cosine angle of two vectors which are \( W \) (weight) of a word and \( W \) (weight) of another word (another document).

Cosine similarity equation is as follows:

\[
\text{SIM}(S_1, S_2) = \frac{\sum t_1 t_2}{\sqrt{\sum t_1^2} \sqrt{\sum t_2^2}}
\]

Information:

\( S_1 = \) the vector of the candidate query or document
\( S_2 = \) the vector of the document beside the candidate
\( t_i = \) weight of word

3.13. Maximal Marginal Relevance (MMR)

The vast majority of modern IR search engines available a positioned rundown for retrieved documents requested by the declining importance of the user’s inquiry. In vary, we inspired those requirements to “relevant novelty” as a prone possibly predominant paradigm. An initial close estimation should correct mapping variety think as of farthest point importance and variety independently and provide a straight mix as those metrics. We bring those straight consolidation ”marginal relevance” - i.e., a document needs minor secondary importance if it may be both fittings of the inquiry and incorporates nominal analogy to formerly chose documents \[11\].

\[
\text{MMR} = \text{ARG}_{\max\ C_{\kappa R/S}}[\lambda \text{Sim}_1(Q, D_i) - (1 - \lambda)_{\max} \times C_{\kappa S}\text{Sim}_2(D_i, D_j)]
\]

Where, \( D_i \): Documents collection, \( Q \): Query, \( R \): Relevant documents, \( S \): Current result set.

Provided for those over definition, MMR \[11\] computes incrementally those standard relevance-

Figure 1. MMR Score for all documents
ranked rundown when those parameter $\lambda = 1$, and computes a maximal differences positioning around those documents in R when $\lambda = 0$. To intermediate qualities from claiming $\lambda$ in the interim $[0, 1]$, a linear combination from claiming both criteria may be optimized. Clients longing to example those majority of the data space around the query, ought set $\lambda$ at an more modest value, and the individuals longing will centring done around different possibly covering or reinforcing important documents, ought further bolstering situated $\lambda$ to an worth closer to 1. We found that an especially successful search method (reinforced by those client ponder examined below) may be on begin for a little $\lambda$ (e.g. $\lambda = 0.3$) in place will comprehend the majority of the data space in the locale of the query, and on concentrate on those most vital parts utilizing an reformulated inquiry (possibly through relevance feedback) and a bigger quality for $\lambda$ (e.g. $\lambda = 0.7$).

4. Result and Discussion
In the pre-processing part, we get the list of words with their TF-IDF scores in document wise as well as in descending order also. So, we use the document wise score in calculating the similarity between the documents and the score, which is in descending order for creating the Query List. Furthermore, find the similarity between the Query List and each input document.

![TF-IDF score list](image1.png)

We performed a pilot test with seven documents for news articles. The reason for examining to figure out if they might advise those contrast the middle of standard ranking technique and MMR. Starting with these scores should be obvious there will be no vast measurable contrast in the middle of those $\lambda = 0.7$ and $\lambda = 0.3$ scores. This is often demonstrated by instances the place the $\lambda = 1$ summary judgement neglected to lift up a bit from claiming important majority of the data and the re-ranking for $\lambda = 0.7$ or 0.3 might.

Those MMR-flow choice techniques to summarization meet expectations surpass for longer documents. MMR may be likewise beneficial to plunge of excursions from multiplex points harshly those same topics. News stories cool an excellent deal repeat of history in-formation.
Table 1. MMR(15%) Scores at different values of $\lambda$

| Documents | $\lambda = 0.0$ | $\lambda = 0.1$ | $\lambda = 0.2$ | $\lambda = 0.3$ | $\lambda = 0.4$ |
|-----------|-----------------|-----------------|-----------------|-----------------|-----------------|
| D1        | -0.984561744    | -0.796186568    | -0.607811392    | -0.419436215    | -0.231061039    |
| D2        | -0.995201172    | -0.813461284    | -0.631721397    | -0.449981509    | -0.268241621    |
| D3        | -0.968851794    | -0.785248336    | -0.601644877    | -0.418041419    | -0.23443796     |
| D4        | -0.976726974    | -0.798918083    | -0.621109191    | -0.433002999    | -0.265491408    |
| D5        | -0.98422112     | -0.804535412    | -0.622648712    | -0.440762011    | -0.258875311    |
| D6        | -0.991420629    | -0.806195342    | -0.620970556    | -0.435744769    | -0.250519483    |
| D7        | 0.428779638     | 0.514535565     | 0.600291493     | 0.68604742      | 0.771803348     |

Table 2. MMR(15%) Scores at different values of $\lambda$

| Documents | $\lambda = 0.5$ | $\lambda = 0.6$ | $\lambda = 0.7$ | $\lambda = 0.8$ | $\lambda = 0.9$ | $\lambda = 1$ |
|-----------|-----------------|-----------------|-----------------|-----------------|-----------------|----------------|
| D1        | -0.042685863    | 0.145689313     | 0.334064489     | 0.522439665     | 0.710814841     | 0.899190017    |
| D2        | -0.086501734    | 0.095238154     | 0.276978041     | 0.458717929     | 0.640457817     | 0.822197704    |
| D3        | -0.050834502    | 0.132768956     | 0.316372415     | 0.499975873     | 0.683579332     | 0.86718279     |
| D4        | -0.087682516    | 0.090126376     | 0.267935267     | 0.445744159     | 0.62553051      | 0.801361942    |
| D5        | -0.076988611    | 0.104898089     | 0.28678479      | 0.46867149      | 0.65055819      | 0.83244889     |
| D6        | -0.065294196    | 0.119331091     | 0.305156377     | 0.490381664     | 0.67560695      | 0.860832237    |
| D7        | 0.428779638     | 0.514535565     | 0.600291493     | 0.68604742      | 0.771803348     | 0.857559275    |

Figure 4. Graph of MMR scores

5. Conclusion

Our System is for saliency detection in text documents based on the extraction method. This salient expression alternately expressions extraction will increase a chance to be a tremendous sufficient for a spectator to comprehend the primary clue of a document. Machine learning methods help in saliency detection in content archives frame firmly related fields, for example, information retrieval or text mining.

As future work we can survive this work for all types of saliency detection in text documents. Integrating the user’s prior knowledge into models will be a decisive factor in saliency detection in text documents. We plan to incorporate Reinforcement Learning Methodologies or dynamically improving the algorithm performance on quasistationary data. In particular, frequently occurring concepts will finalize, and novel ideas will include over a timeline additionally.
first-order Markov Chain. Assumptions can attempt, and Q-Learning algorithms will be incorporated.

References
[1] Neto, Joel Larocca et. al. “Automatic text summarization using a machine learning approach.” Brazilian Symposium on Artificial Intelligence. Springer Berlin Heidelberg, 2002.
[2] Luhn, Hans Peter. “The automatic creation of literature abstracts.” IBM Journal of research and development 2.2 (1958): 159-165.
[3] Edmundson, Harold P. “New methods in automatic extracting.” Journal of the ACM (JACM) 16.2 (1969): 264-285.
[4] Baxendale, Phyllis B. “Machine-made index for technical literature—an experiment.” IBM Journal of Research and Development 2.4 (1958): 354-361
[5] Salton, Gerard, and Christopher Buckley. “Term-weighting approaches in automatic text retrieval.” Information processing and management 24.5 (1988): 513-523.
[6] Kupiec, Julian, Jan Pedersen, and Francine Chen. “A trainable document summarizer.” Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 1995.
[7] Hovy, Eduard, and Chin-Yew Lin. “Automated text summarization and the SUMMARIST system.” Proceedings of a workshop on held at Baltimore, Maryland: October 13-15, 1998. Association for Computational Linguistics, 1998.
[8] Yulita, Winda, and Feddy Setio Pribadi. “The Implementation of Maximum Marginal Rel-evance Method on Online National and Local News Portal.” Proceeding of Engineering International Committee 4 (2015).
[9] Carbonell, Jade, and Jade Goldstein. “The use of MMR, diversity-based reranking for re-ordering documents and producing summaries.” Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 1998.
[10] Huang, Anna. "Similarity measures for text document clustering." Proceedings of the sixth new zealand computer science research student conference (NZCSRSC2008), Christchurch, New Zealand. Vol. 4. 2008.
[11] Carbonell, Jaime G., and Jade Goldstein. "The use of MMR, diversity-based reranking for reordering documents and producing summaries." SIGIR. Vol. 98. 1998.