Knowledge Base Driven Automatic Text Summarization using Multi-objective Optimization

Chihoon Jung[1], Wan Chul Yoon[2]
Graduate School of Knowledge Service Engineering
KAIST
291 Daehak-ro, Yuseong-gu, Daejeon 34141,
Republic of Korea

Rituparna Datta[3], Sukhwan Jung[4]
Department of Computer Science
University of South Alabama
150 Jaguar Dr, Mobile 36688,
Alabama, USA

Abstract—**Automatic Text summarization aims to automatically generate condensed summary from a large set of documents on the same topic. We formulate text summarization task as a multi-objective optimization problem by defining information coverage and diversity as two conflicting objective functions. With this formulation, we propose a novel technique to improve the performance using a knowledge base. The main rationale of the approach is to extract important text features of the original text by detecting important entities in a knowledge base. Next, an improvement on the multi-objective optimization algorithm is also proposed for the automatic text summarization problem. The focus is on improving efficiency of the each steps in the evolutionary multi-objective optimization process which is applicable to all tasks with the same problem formulation. The result summary of the suggested method ensure the maximum coverage of the original documents and the diversity of the sentences in the summary among each other. The experiments on DUC2002 and DUC2004 multi-document summarization task dataset shows that the proposed model is effective compared to other methods.**

Keywords*—*Multi-document summarization; evolutionary multi-objective optimization; knowledge base; named entity recognition

I. Introduction

As text information publication speed outgrows our consumption capability, there have been many approaches to deal with the information overload problem suggested by the research community, such as information retrieval [1], semantic web [2], and text summarization [3]. While the results are, to some extent, successful and promising, we still need to deliver the text content efficiently so that the readers can consume more qualitative content within a limited amount of time.

The goal of this paper is to propose a generic, extractive, and multi-document summarization method. Each of these summarization types has an alternative approach, namely, query-focused, abstractive, and single-document summarization. As opposed to a generic summarization, some keywords are provided for a query-focused summarization task. The summarizers proceed with the summarization using the query term as a guide. Extractive summarization task composes a summary with unaltered sentences selected from the original document set, which is distinguished from an abstractive summarization task where sentence modification or phrase selection and generation are allowed.

To solve a generic extractive summarization problem, the authors propose a model using evolutionary multi-objective optimization. Multi-objective optimization approach to a text summary generation task is gaining attention recently from the research community [4], [5], [6]. Previous research directions mainly focus on applying and testing diverse optimization methods within the multi-objective problem formulation. Here, we begin the discussion by setting the goal as how to define a robust objective function. The objective function suggested in this paper evaluates all the sentences in a document set as a whole, as opposed to a local evaluation approach which evaluates each sentence one at a time causing local optima entrapment. There are many ways to define objective functions. In this research, the main objective functions are the coverage and the diversity functions. Although many other methods use a simple coverage function, we claim that improving the coverage function is especially important. From this perspective, we propose a method on how to utilize the knowledge base which encodes how human thinks and what is important.

The method to calculate the objective function values relies on multiple text mining techniques including term weighting scheme, similarity measurement function, text preprocessing, and Named Entity Recognition (NER). The objective function evaluations are continuously performed based on these techniques as the optimization progresses. The proposed objective functions are based on coverage and diversity evaluations. While these concepts were individually dealt with in other text summarization approaches [7], [8], [9], considering them as two objective functions within a coherent multi-objective optimization framework provides a power to adopt many other improvement strategies, not limited to the specific evaluation concept in question. On top of this framework, the authors suggest a novel knowledge-based named entity topic construction approach for a robust objective function development. The rationale behind the proposed approach is that document summarization can be seen as a multi-objective optimization problem, defining the objective functions to reflect the characteristics of well-generated summaries. Such an optimization task requires binary encoding of data, and assigning one bit for each sentence fits the extractive summarization task. Maximum content coverage and maximum content diversity are deemed equally important and at the same time conflicted. Hence, the problem is modeled as a multi-objective optimization using the two conflicting objective functions.

In this research, we use a high-performance evolutionary multi-objective optimization method Non-dominated Sorting Genetic Algorithm II(NSGA-II), and propose an adaptive NSGA-II for a text summarization task. The technique is a state-of-the-art algorithm for multi-objective optimization tasks producing a population of possible solutions as opposed to single-objective optimization algorithms which can only provide a single solution. The algorithm adds more flexibility to the proposed method as a user can generate as many equally optimal summarizations as one needs by increasing the population size.

We first review previous approaches in the text summarization domain. Then we argue that adopting a multi-objective optimization technique combined with a knowledge-based approach shows a state-of-the-art summarization result.

For a thorough comparison, Document Understanding Conference 2002 (DUC2002) and 2004 (DUC2004) multi-document summarization (MDS) task dataset are used as an evaluation source, and the comparisons are done against multiple methods in the literature including non-optimization-based and optimization-based methods.

The next section details related work from past studies. Section III explains how the text summarization problem is formulated into a multi-objective optimization. Section IV explains how the knowledge-based coverage objective function is defined and calculated. Section V explains the multi-objective optimization and the details on the improvement of the NSGA-II optimization method when applying to the text summarization problem. Experiment results are shown in Section VI with a comparison against past studies. The last Section VIII discusses the results and possible future research opportunities.

II. Related Work

In this section, we present a review of the main research fields related to text summarization and optimization. The research on text summarization started in the late fifties [10]. Automatic text summarization is a complex and challenging task ranging over multiple domains of research. There are many issues to consider when generating a summary from multiple documents, such as coverage, diversity, redundancy, temporal dimension, co-reference, sentence ordering, synonymy, and so on. The focus was on the single document summarization at this stage. Researchers started to focus on an MDS problem in later years [11]. Goldstein et al. [12] suggested extraction of sentences from multiple documents approach on top of single-document summarization techniques.

Text summarization can be grouped into two types of tasks. Abstractive summarization aims to generate a new set of informative sentences by utilizing existing phrases [13], [14]. The set of concepts are usually extracted from the given document dataset. Then summarization is automatically written with the selected noun and verb phrases in the conformation of sentence construction constraints and saliency-maximized order. However, abstractive summarization methods generate non-fluent summaries and have high computational complexities; their performance improvements mainly comes from the improvements of other research fields, such as integer linear programming methods and grammatical sentence formulations [14]. On the other hand, the research on extractive MDS was focused on how to select the most relevant sentences to be included in the summary. The extraction and representation of the topics included in the original documents are two of the important issues in this area [15]. Reducing redundancy and maximizing diversity in the generated summary is key to the summary generation task, therefore these are the two objectives that most MDS techniques consider important.

There are numerous other research on redundancy reduction with extractive MDS. Sarkar [16] used local and global trimming rules to tackle the redundancy problem. Carbonell et al. [17] utilized both query-relevance and information-novelty, using the Maximal Marginal Relevance(MMR) to reduce redundancy while preserving query relevance for the summarization. Using an unsupervised approach, Zha [18] explicitly modeled the key phrases and the sentences that contain them. The model is represented as weighted undirected and weighted bipartite graphs to extract the sentences without going through an extensive training phase. Centroid-based summarization has shown success in many past works in redundancy reduction as well. The earlier research suggesting a centroid-based approach for MDS [19] relies on TF-IDF term weighting scheme to measure the centrality of the sentences by comparing the similarity of the sentences to each cluster centroid. LexRank [20] is later suggested where they first represent a graph of sentence relations by using intra-sentence cosine similarity and calculate eigenvector centrality of the sentences. Their main contribution is to measure sentence centrality based on the sentence relations rather than relying on cluster centroids. They assume that sentences with more similar sentences are considered to be more central. Biased LexRank [21], a semi-supervised method on pairwise lexical similarity sentence graph, is later proposed to use both intra-sentence and inter-sentence similarities for the task. The method allows topic, or query, sensitive sentence retrieval with weighted random-walk based on a prior distribution of sentence ranks, performing well on both extractive text summarization and passage retrieval tasks. StarSum [22] focuses more on the intra-sentence similarities and proposes a star-shaped sentence - topic bigram bipartite graph to emphasize intra-sentence topic discrepancy, representing each sentence as a collection of topic phrases. On top of intra-sentence and inter-sentence similarities, intra-document sentence similarity distinguished from inter-document sentence similarity allows Document-Sensitive Ranking (DsR) [23] algorithm to treat multi-documents as individual documents with different topics and information rather than one large document. DsR algorithm utilizes document-sentence and document-document links as well as sentence-sentence links showing top performance on both the DUC2004 and DUC2007 dataset.

There are a number of other MDS research utilizing graph-based approaches. Cluster-based conditional Markov random walk [24] model overcomes the limitation of directly applying Markov random walk on MDS fields, differentiating sentence clusters (thematic representation of a document) with varying size and importance as well as weighting intra-cluster sentences based on to-centroid distance. The authors also propose a cluster-based hyperlink-induced topic search (HITS) model to analyze graph link in different perspectives and show both models perform well on the DUC2001 and DUC2002 dataset. iSpreadRank [25] aims to improve the sentence ranking phase of extractive MDS task with the concept of spreading activation theory. The method recursively spread sentence-specific scores to their neighbors in the graph model of document sentences, allowing utilization of neighbor importance as well as neighbor counts when ranking individual sentences. Generic summary methods tailored to iSpreadRank are tested on the DUC2004 dataset, with the best performing variant having 0.38068 ROUGE-1 score better than the second-best system. Centrality is an importance measure for an individual node within a graph. Introduction of super-vertices to the sentence graph each representing subset of sentences within a document set allows the calculation of super-centrality, an importance measure for the super-vertex hence sentence group to the document set [26]. This allows a robust non-redundant sentence selection compared to the existing MMR method and shows performance improvement over the aforementioned LexRank method. More generic graph-based framework applicable to all generic, query-based, update, and comparative MDS tasks is proposed based [27] on using the minimum dominating set, a minimum subset of a graph where every vertex in the graph is either member or neighbor of it, as a basis for the document summary. Other graph-based MDS research includes semantic linkage analysis, where relationships between sentences are measured by their semantic relationships [28]. Sentence fusion technique [29], including bottom-up local multi-sequence alignment, is proposed to shift the MDS research field from extractive summarization tasks to abstract summary generation tasks as well.

Clustering-based approach is also proposed to deal with MDS where documents and in turn their sentences are clustered together by using features such as cosine similarity, and sentences with best scores within each cluster are retrieved to form a summary [30]. Clustering on extractive MDS task works in three steps; sentence clustering, ordering, and cluster representative sentence selection. Histogram based clustering, content-word weight-based cluster ordering, and local/global word importance-based sentence selection shows ROUGE-1 score higher than the second-best system of the DUC2004 in task 2 [31]. Term-vector based document clustering, feature profile based sentence selection from clusters followed by chronological ordering creates summaries with higher sentence scores compared to a centroid-based clustering algorithm and showed it can extract relevant sentences across multiple documents [32]. Jung et. al. [5] was the one of the early research that proposed topic-based MDS using multi-objective optimization. Here the topics are defined as clusters of terms.

Several interdisciplinary research is done on the topic of MDS, and the use of topic modeling is one of them. Probabilistic latent semantic analysis (PLSA) on sentence-level topic distribution produced three variants of query-focused extractive summarization method all of which showed high ROUGE-1, ROUGE-2 and ROUGE-SU4 scores on a par with best reported on the DUC2006 and DUC2007 [39]. The Pyramid method [40] is a manual evaluation approach opposite to the commonly used ROUGE [41] method, evaluating automated summaries against human-annotated summaries. Hennig [42] further expanded on unsupervised semantic analysis on text units by mapping topic models towards summary content units used with the Pyramid method, proving a trained probabilistic topic model exhibit structures similar to the human model summaries. The use of the term-document matrix is proposed to overcome the semantic limit of the term-sentence matrix commonly used in existing MDS methods to realize hidden topics embedded within the document collection themselves. Bayesian sentence-based topic models (BSTM) [34] uses both term-sentence and term-document matrices to build document-sensitive sentence topic models and show higher ROUGE scores than six existing summarization method, nearing the scores of the best team in both the DUC2002 and DUC2004. Numerous multidisciplinary research is done on topic models for MDS. Fuzzy logic, in combination with a topic model, showed that the topic words can be replaced with fuzzy elements to build a fuzzy inference summarization system producing automated summaries focusing on divergence and similarity [43]. Distributed processing framework such as MapReduce is also proposed to overcome the computing intensity of MDS using topic modeling, nearly halving the computation time with four nodes when the dataset grew up to 3890 documents [44].

Deep learning based approaches are getting attention recently. Most of the recent state-of-the-art performance models for the NLP tasks are based on Pre-trained Transformer-based [45] deep neural network models such as BERT [46]. Deep learning models are applied to extractive sumarization task as shown by Liu [47]. Computing a feature space of sentences from a single document with deep auto-encoder (AE) is reported to improve the feature space recall by 11.2% on average [48]. The approach uses an ensemble noisy auto-encoder (ENAE) which aggregates sentence selection over multiple runs by adding random noise to the word vector, allowing more robust behavior even with a smaller vocabulary.

Summary optimization is a more recent approach to MDS. One of the key features dictating clustering performances in this approach is its criterion or objective function. A differential evolution algorithm is proposed [49] to optimize an objective function of clusters found by normalized google distance (NGD) [50], where each candidate sentence is represented as a gene in a chromosome with the number of clusters as its value range. A self-adaptive differential evolutionary algorithm is applied to redundancy in MDS with the optimization goal of reducing semantic redundancy in summary sentences. Sentence scores are measured based on other sentences within a summary guaranteeing both diversity and coverage to be measured as a discrete optimization problem, which is mediated by the introduction of self-adaptive crossovers within the DE algorithm [9]. The use of optimization and machine learning technique on document summarization is one of the main approaches to the MDS problem while relatively new compared to statistic-based methods. Multi-objective optimization(MOO) on MDS is a lesser studied variant of optimization-based MDS where multiple summaries with equal overall quality can be produced by mediating multiple, often conflicting, quality measures. Huang et al. [51] proposed MOO modeling of MDS to overcome sentence redundancy problem in single-objective optimization approach, as well as information coverage, significance, and text coherence. Artificial bee colony optimization [6] method modified with multi-objective capability is used on MDS to show that the approach can be used to enhance the performance of existing work by incorporating both sentence coverage and redundancy as optimization objectives. Sekaran et. al. [52] combined Information Retrieval technique with text summarization to better serve human information needs.

TABLE I. LIST OF DOCUMENT SUMMARIZATION METHODS TO COMPARE

Topic	Main characteristics of the method	Use of coverage or diversity
2001, LSA[7]	Latent Semantic Analysis	coverage & redundancy
2004, LexRank[20]	Graph-based method	
2004, Centroid[19]	Centroid-based method	
2009, NMF[33]	Non-negative Matrix Factorization	
2009, MCKP[8]	Knapsack problem solving using greedy algorithm	coverage
2009, BSTM[34]	Bayesian Sentence-based Topic Model	
2011, FGB[35]	Kullback-Leibler divergence, Factorization with given bases	
2012, WCS[36]	Weighted consensus scheme	
2013, OCDsum-SaDE[9]	Single-objective optimization	coverage & diversity
2016, GO[37]	Genetic-based optimization	coverage & redundancy
2017, CRSum[38]	Contextual Relation-based, Neural Network Model	
2018, ABCO[6]	Artificial bee colony optimization	coverage & redundancy
DUC	Best result of the participants in the DUC	
Random	Random summary selection strategy for a baseline	

Among diverse approaches to tackle the problem of extractive summarization, there are some advantages of multi-objective optimization method. From the practical perspective, it is sometimes more useful to generate multiple alternative summaries rather than single result summary. One of the advantages of the proposed approach is that it produces multiple non-dominating solutions where a person can choose to select a final solution with varying properties after the multiple candidate solutions are generated. In the case where one final solution is required, the authors suggest a solution selection strategy amongst the multiple summaries. This feature provides higher generalizability to the proposed method, granting utility in the research, industrial, and personal aspects. Additionally, multi-objective summarization provides users more choices to select summarization based on combinations of various – and possibly conflicting – characteristics, such as coverage and diversity. In the context of an interactive environment where people can interactively read multiple summaries, the multi-objective optimization can offer various candidates for users to select from. This cannot be achieved by single-objective summary generation methods. Table I shows the list of methods we compare to including their use of either coverage or diversity as a feature.

III. PROBLEM STATEMENT

We formalize the text summarization problem as an optimization problem. The goal of the extractive text summarization task is to select the sentences that reflect the original text as much as possible within the given length. A basic approach is to evaluate the sentences individually and select the ones that satisfy a set of predefined criteria. However, this localized approach may suffer from the selected sentences failing to cover the whole content of the original text. The advantage of the optimization approach is that the evaluation is performed as a whole for a set of selected sentences. This allows our approach to overcoming the weakness of the previously described basic approach. The proposed model consists of two large modules continuously working together throughout the optimization process to generate the Pareto optimum summaries. The optimization module is in charge of optimizing the two objective functions using evolutionary multi-objective optimization, and the text processing module accepts the binary-encoded candidate summaries and returns calculated objective function values. Objective function calculation is normally evaluated by solving the algebra equation. The separate text processing module is necessary for this research because the objectives can only be calculated by going through a complicated text processing process including several steps of preprocessing and similarity computation.

A. Text Similarity Model

Throughout the automatic text summarization process, the similarity measurement between two text segments is constantly performed. This is a basis for a high-quality summary. In the proposed method, the most widely used cosine similarity measure is adopted. Cosine similarity measures how much the two given sentences are similar in a term vector space. For this purpose, we represent our text segments using a Vector Space Model(VSM). VSM is a way to represent a text as a vector so that any text segment can be represented in a coherent vector space without considering the order of the terms in the original text. By representing each sentence using VSM, we can use many methods that do not consider term order, such as the cosine similarity measure. If we consider the term order when comparing the sentence similarity the two pieces of sentences that contain similar content may be given a very low score which is not appropriate for our purpose.

For the two objective function calculations, we use cosine similarity as a similarity function. Our main idea is to utilize the power of the binary multi-objective optimization algorithm to the full extent on a text summarization problem. For the algorithm to deal with the text units, we represent any set of sentences as a binary vector. Thus, both a document and a generated summary are fixed-length binary element vectors with the length being the total number of sentences in the document set D. Since we are dealing with multi-document problem, let $D = \{d_1, d_2, ..., d_N\}$. N is the total number of documents that exist in corpus D. As we deal with a sentence as a summarization unit, the document set D is also defined in terms of sentences and defined as $D = \{s_1, s_2, ..., s_n\}$, where n is the total number of sentences in the original document set and $s_i \in \{0, 1\}$. $T = \{t_1, t_2, ..., t_m\}$ represents all the unique terms that appear in document corpus D. This is our vocabulary to be used and a corpus of documents contain m unique terms.

Each sentence consists of a list of terms, shown as a list of real number weights each representing the importance of the term. There are many ways to assign weights to the terms in a sentence. In this paper, we use a modified TF-IDF term weighting scheme which is widely used for text mining tasks. The basic unit of selection is a sentence in this research and the inverse frequency is calculated per sentence instead of per document. In this formulation, a term weight is defined as:

$$w_{ik} = tf_{ik} \cdot isf_k \tag{1}$$

w_{ik} is a weight for term t_k in sentence s_i. tf_k is the number of occurrences of term t_k in sentence s_i. $isf_k = log(n/n_k)$, where n_k is the number of sentences that contains the term t_k. isf_k is calculated by the sentences frequency n_k divided by the total number of sentences n. With this weighting scheme, i_{th} sentence s_i is represented as $s_i = \{w_{i1}, w_{i2}, ..., w_{im}\}$ where m is the total number of terms in the document collection. w_{ik} is a weight value for k_{th} term in i_{th} sentence.

Cosine similarity measures how much the two given sentences are similar. For the two objective function calculations, we use cosine similarity as a similarity function. Let s_i and s_j be the two sentences to be compared, where m is the total number of distinct terms in the document collection. Cosine similarity between the two sentences is defined as:

$$sim(s_i, s_j) = \frac{\sum_{k=1}^{m} w_{ik} w_{jk}}{\sqrt{\sum_{k=1}^{m} w_{ik}^2 \cdot \sum_{k=1}^{m} w_{jk}^2}}, \quad i, j = 1, 2, ..., n. \tag{2}$$

B. Problem Formulation using Multi-objective Optimization

From Section 2, we can observe that most of the research work in MDS considered information coverage, diversity, and a weighted combination of both. These are contradictory goals as often are in most decision-making situations, as higher information coverage ensures a more informative summary whereas higher diversity results in less redundant information in the outcome. These two goals cannot be obtained with a single optimization task. As a result, a contradiction between information coverage and diversity is modeled as a multi-objective optimization problem. Their contradiction tendencies can be shown when the summary size changes; the coverage increases while the diversity decreases when more sentences are extracted for the summary.

The main purpose of multi-objective optimization is to identify all non-dominated solutions that are different from each other and hold equal importance. These solutions provide more flexibility to the complex problem compared to the outcome of single-objective optimizations, as any solution can be chosen to satisfy the specific requirements of the user. NSGA-II [53] is used to produce non-dominating solutions by considering the conflicting objectives. When a single solution is required, we select the Pareto optimal solution with the maximum coverage as our final summary.

The mean vectors of the original document set and the solution summaries are compared to allow the similarity comparison independent of the solution length. Sentence by sentence comparison will result in a biased similarity measure, where solutions with more sentences receive higher scores when compared to the whole set. The center of sentences in the given vector set is used instead to remove the number of sentences during the calculation, where the similarity is measured term by term instead. Mean vector $o = [o_1, o_2, ..., o_m]$ is used to represent a center of a set of sentence vectors in the original document set D, and the mean summary vector $o^s = [o_1^s, o_2^s, ..., o_m^s]$ represents a center of a set of extracted sentence vectors and is used to compare the similarity between o and the solution summary. Each element for the kth term is defined as:

$$o_k = \sum_{i=1}^{n} w_{ik}, k = 1, 2, ..., m. \tag{3}$$

$$o_k^s = \sum_{i=1}^{n} w_{ik} \cdot x_i, k = 1, 2, ..., m. \tag{4}$$

o and o^s are used in the coverage objective function calculation, where the differences between the original set D and the solution summary $\bar{D}$ is given by the inclusion variable $x_i \in \{0, 1\}$, which is a binary variable representing whether a sentence s_i from D is selected to be included in $\bar{D}$. The coverage objective function is to evaluate the similarity between the summary and the original document set and is defined as:

$$f_{coverage}(\mathcal{X}) = sim(o, o^s) \cdot \sum_{i=1}^{n} \frac{sim(o, s_i) \cdot x_i}{n} \tag{5}$$

We normalize the summation of the similarity values to mitigate the bias towards including more sentences with fewer terms. Otherwise, the summary will be biased towards including shorter sentences.

The diversity objective function is defined as Equation 6. There is another similar concept called redundancy in the literature. The difference between the diversity and the redundancy is that redundancy is calculated as the sum of cosine similarity, whereas the diversity is calculated as the sum of cosine distance, where $cosine\_distance = 1 - cosine\_similarity$.

$$f_{diversity}(\mathcal{X}) = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \frac{(1 - sim(s_i, s_j)) \cdot x_i \cdot x_j}{n \cdot (n-1)} \tag{6}$$

IV. Knowledge based Driven Keyword Extraction

A. Knowledge Base

Defining an objective function that highly reflects how human summarizers perform is a guide to finding successful solutions using an evolutionary algorithm. To improve the basic coverage function defined in Section III, we utilize human knowledge and incorporate it into our coverage objective function. The proposed knowledge base driven text summarization relies on entity information encoded in an underlying knowledge base. As the term knowledge base driven implies, the coverage objective function we propose guides the optimization process towards the near-optimal solutions regarding human cognition. In this work, we use DBpedia as our source knowledge base and DBpedia-spotlight as an interface to annotate a text. DBpedia [54] is a knowledge-base

TABLE II. EXAMPLE TEXT SNIPPETS FROM TOPIC D061 OF DUC2002 DATASET AND THE TYPES

Example: "The **National Hurricane Center** said a hurricane watch was in effect on the coast from **Brownsville** to **Port Arthur** and along the **coast** of northeast **Mexico** from **Tampico** north. The National Hurricane Center said Gilbert was the most intense storm on record in terms of barometric pressure."

Detected Entity Term	DBpedia type	Freq
National Hurricane Center	DBpedia:Agent	85
	DBpedia:Organisation	49
	DBpedia:GovernmentAgency	16
coast	DBpedia:Place	259
	DBpedia:Location	259
	DBpedia:PopulatedPlace	226
	DBpedia:Settlement	89
Brownsville	DBpedia:Place	259
	DBpedia:Location	259
	DBpedia:ArchitecturalStructure	14
	DBpedia:Infrastructure	6
	DBpedia:Station	1
Port Arthur	DBpedia:SocietalEvent	13
	DBpedia:Event	13
	DBpedia:MilitaryConflict	12
Mexico	DBpedia:Place	259
	DBpedia:Location	259
	DBpedia:PopulatedPlace	226
	DBpedia:Country	102
Tampico	DBpedia:Place	259
	DBpedia:Location	259
	DBpedia:PopulatedPlace	226
	DBpedia:Settlement	89

where the community has put an effort to extract structured information from Wikipedia. In the proposed method, the type information of the entities in DBpedia are used for the knowledge base driven approach, where the term frequencies show reverse correlation to the term specificity; types with a high type frequency are more general, while less frequent types have a more specific meaning. The words with a low type frequency have more significance unlike that of the tern frequency, as their specificity leads to higher chances of them having more important roles in the context of the text. Note that each term may be included in multiple types. Any knowledge bases such as YAGO [55] ontology can be also used for this purpose, and DBpedia Spotlight [56] was used for our knowledge-based NER.

Most of the statistical approaches such as TF-IDF rely on the frequency of the terms. However, this may suffer from rare but important terms being assigned a lower weight value. Knowledge base driven method overcomes this missing low frequency, high importance terms by relying on the source that explicitly encodes this information. It is often the case that the knowledge bases and knowledge graphs are used in a complex entity detection and graph matching of semantic information. Although the complicated methods are very useful in certain cases, the proposed method utilizes the encoded information embedded in the knowledge base during the knowledge base construction. Since knowledge bases are constructed based on the ontology reflecting human thoughts, the types represented in the ontology and the entities that belong to those types are a gist of human knowledge encoded for the computers to access and mirror human intelligent behavior. The proposed method makes use of this explicitly encoded information to detect the keywords. The keywords we use are the Named Entities. As we can see from the other research on NER, detecting the named entities can boost the performance of any model that relies on NLP technology.

Table II shows an example of DBpedia types that can be linked to the terms in the example text. The raw text column contains the NER result based on the DBpedia knowledge base. DBpedia type column lists the types linked to the entities after linking it to the KB using NER. The number shows the occurrence of each type in the corresponding topic. We can see that there can be multiple types linked to each entity. Table II shows that the knowledge-based named entity keyword extraction and topic construction can give higher importance weights to the rare but infrequent terms.

B. Topic Analysis

The necessity of adopting the type of topics for the coverage objective function comes from the bias that may exist in the similarity value aggregation. A term or a sentence level comparison for the text similarity may suffer from a preference bias towards the sentences that cover more range of available terms. If the original document set has many terms describing the same topic, the generated summary has a higher chance to include redundant content. Term and sentence level similarity calculations on the detected topics are done to mitigate the original document's bias towards a small portion of topics.

C. Knowledge based Driven Coverage

Algorithm 1 Knowledge Base Named Entity Extraction.

```
INPUT: Threshold $\theta$
OUTPUT: Key Entity SET(KES): $K$
Initialize $K$, $T$
for each $d \in D$ do
    for each $s \in d$ do
        for each $t \in s$ do
            if $t \in E$ then
                T.add(DBpediaTypeOf(t))
Sort C by count in decreasing order
$\bar{T} \leftarrow$ lower $\theta$ percent of T
for each $d \in D$ do
    for each $s \in d$ do
        for each $t \in s$ do
            if $DBpediaTypeOf(t) \in T$ then
                $K$.add($t$)
return K
```

The process of knowledge base driven named entity extraction is based on the algorithm shown in Algorithm 1. This process extracts the Key Entity Set(KES) and constructs topics from the result. The proposed model first performs named entity detection using a knowledge base. As described earlier, from using knowledge bases, we can also acquire type information from the entity. The types can be seen as an abstract layer of the terms. After the NER, we extract the types of the detected entities. The types collected are defined as T. Then, we filter out the common types using a threshold parameter θ to extract θ percent of least-frequent types. By removing the common types, we can learn the rare but important entities of those type set. From the filtered type set T, a set of terms included in those types from the target text document is collected. This is a Key Entity Set(KES) defined as K. KES is considered as a virtual topic, namely, Named Entity Topic(NET). After NET is collected, we consider these terms the same as normal terms explained in Section III. Based on KES, the improved coverage objective function is defined similarly using TF-IDF weighting scheme and cosine measure based on the coverage objective function defined in Section 5.

D. Knowledge Base Driven Coverage

In this section, we define the knowledge base driven coverage function $f_{coverage-k}$ using the KES explained earlier. Mean entity vector $\mu^s = [\mu_1^s, \mu_2^s, ..., \mu_m^s]$, is used to compare the similarity between the mean document vector o and the constructed topics. The topics are constructed from the named entities that exist in the knowledge base. We have defined Key Entity Set(KES) and Named Entity Topic(NET). NET if constructed from KES using a Knowledge base NER. The detailed steps are explained in Algorithm 1. Each mean vector of NET for document d is defined by:

$$\mu_k^s = \sum_{i=1}^{n} w_{ik} \cdot y_i, k = 1, 2, ..., m. \tag{7}$$

where $y_i \in 0, 1$ is a binary variable that represents whether a term t_i is in KES. Now the improved knowledge base driven coverage objective function $f_{coverage-k}$ is defined as follows:

$$f_{coverage\text{-}k}(\mathcal{X}) = sim(o, \mu^s) \cdot sim(o, o^s) \cdot \sum_{i=1}^{n} \frac{sim(o, s_i) \cdot x_i}{n} \tag{8}$$

V. Adaptive Multi-objective Optimization for Text Summarization

A. Non-dominated Sorting Genetic Algorithm (NSGA-II)

Developed as an improved version of NSGA, NSGA-II has been one of the best performing multi-objective optimization algorithms since its inception in 2002 [53], having been applied to diverse areas such as engineering, computer science, biology, and economics. It generates a population of individual solutions to obtain a set of approximated Pareto-optimal solutions. The algorithm improves the non-dominated solutions in each iteration till it can achieve the least erroneous set of non-dominated solutions. There are two main operators involved in the foundation of NSGA-II. These two operators are crowding distance and non-dominated sorting. Out of these two operators, crowding distance is used to guarantee a set of diversified solutions in the whole search space. On the other hand, non-dominated sorting is used to select the best solutions from crossover and mutation by disposing of the solutions worse than other population members. These two operators are also responsible for ranking both dominated and non-dominated solutions.

The first step of NSGA-II is the generation of N population members, which are randomly generated based on the given lower and upper variable bounds. Thereafter these N population members are sorted used a non-dominated sorting algorithm. The new child solutions are formed with two genetic operators known as crossover and mutation. After this step, the N parent and child population are coupled together to create a combined population of size 2N. Then N non-dominated solution members are selected from them using the non-dominated sorting algorithm. Thereafter, crowding distance is obtained to discard the low-density solutions which are the solutions that are closed to each other. These steps are continued till the maximum number of generations specified by the user is reached or the termination criteria are met.

Algorithm 2 Adaptive NSGA-II for text summarization.

```
INPUT: $n \geq 0 \vee x \neq 0$
OUTPUT: $y = x^n$
Adaptive Initialization
Generate non-dominated population $P_0$ of size N
Compute the objective function values of the initial population
Adaptive Mutation
Adaptive Crossover
Generate children population $Q_s$ of size N
while $i \leq i_{max}$ do
    Merge parents and children while maintaining ellitism($R_i = P_i \cup Q_i$)
    Fast non-dominated sorting algorithm(non-dominated Pareto fronts $F_1, F_2, ..., F_k$ in $R_i$
    Crowding distance sorting
    Tournament selection(generate next population $P_{i+1}$ from $R_i$)
    Next generation $Q_{i+1}$
    $i \leftarrow i + 1$
```

B. Improved NSGA-II for Text Summarization

The improvements on NSGA-II are done in the context of keeping the length limit constraint and conforming to the constraint while each step is carried out. The improved version can also be generalized and applied to any genetic algorithm with initialization, mutation, or crossover. Since this is the case for every genetic-based algorithms, the method we suggest can be generalized to any genetic algorithm. As evolutionary algorithms generally do not consider the characteristics of each gene during the evolution progresses, each step generates infeasible candidate solutions which not only decreases the performance but also leads to inefficiency. In the text summarization task, the variance of the sentence lengths is quite high compared to the length constraint. For example, the DUC2002 competition had a summary length constraint of 200 words, and the sentence length varied from a couple of words up to 10% $\sim$ 20% of the length limit. We apply modifications in the initialization, crossover, and mutation part of the existing evolutionary multi-objective optimization algorithm to accommodate the 200 words limit constraint by adding or removing sentences from the summary. Algorithm 2 shows the process of proposed improved NSGA-II. The following explains the details of each step.

C. Chromosome Encoding

Each of the candidate summaries is considered as a chromosome since each sentence is encoded using one bit per sentence as explained in Section III. We use binary encoding for the summary representation.

Algorithm 3 Adaptive initialization

```
for each Entry in chromosome do
    for each gene $g_i$ do
        Take a random real value $\gamma$ between 0 and 1
        Take a random integer value $\sigma$ between $\alpha$ and $\beta$
        if $\gamma < 1 - \frac{\sigma}{n}$ then
            $g_i \leftarrow 1$
        else
            $g_i \leftarrow 0$
```

1) Adaptive Initialization: We parameterize initialization with lower bound length variable α and upper bound length variable β. This process generates a random length chromosome approximately between α and β. Since one of the rationales behind the evolutionary algorithm to find optimal value is randomness, the proposed methods also reflect randomness throughout the process.

Algorithm 4 Adaptive mutation

```
INPUT: Mutation probability $P_m$, zeros ratio $\alpha$, ones ratio $\beta$
for each gene in chromosome do
    Take a random real value $\gamma$ between 0 to 1
    if $\gamma < p$ then
        Take a random real value $\gamma'$ between 0 to 1
        if $g_i$ is 0 then
            if $\frac{\bar{\gamma}}{2} < \alpha$ then
                $g_i \leftarrow 1$
        else
            if $\frac{\bar{\gamma}}{2} < \beta$ then
                $g_i \leftarrow 0$
```

2) Adaptive Mutation: A mutation operator ensures diversity in the population members. It is possible that crossover can not create new solutions. In this case, the mutation is used to perturb the solution and introduce new members to the population. The limitation of random mutation is that it has an adverse effect when the expected solutions are highly skewed. In the DUC2002 dataset, for example, the number of genes(sentences) in a chromosome range from 111 to 614 with an average of 260. Out of this chromosome length, only about 10 genes are to be selected in a candidate solution. Applying unskewed randomness leads to random gene selections, where more genes would be selected than necessary. Although the whole optimization process prevents the infeasible solutions, the power of each generation would be wasted because of the infeasible genes. Our strategy is to not allowing too much divergence from the initial ratio of selected and non-selected genes. We also randomize this process so that as the evolution progresses, some of the genes diverge from the initial ratio.

3) Adaptive Crossover: The crossover operator is responsible for choosing two random members and create two new child solutions. The parent and child solutions can be the same or different.

However, a desired skewness in the solution could become an issue as in initialization and mutation operation during the crossover stage. During a crossover, two chromosomes' subsequences between crossover location parameter s and t are swapped. Because of the initial gene skewness in the original chromosomes, this could lead to an undesirable degree of gene ratio altercation resulting in infeasible child solutions. To overcome this problem, we allow the exchange of the genes in a unit of pair. The selected pair exchange ensures that the number of active genes is maintained after the crossover.

Algorithm 5 Adaptive crossover.

```
INPUT: crossover probability $P_c$
Take random real number $r$ between 0 and 1
for each Entry in chromosome do
    if $r \leq P_c$ then
        Select random integer i
        while $x_i$ is $y_i$ do
            Select random integer j
            while $x_j$ is $y_j$ do
                Crossover ($x_i$, $y_i$)
                Crossover ($x_j$, $y_j$)
```

D. Selection

The selection operator is used to select the better members within the population. Usually, two members are chosen randomly and the best one is chosen. Thereafter, multiple copies of the best solution members could be selected redundantly.

E. Stopping Criterion

The optimum number of generations is correlated with the number of sentences in the document collection. The stopping criterion could be given as the number of consecutive generations with static objective function outcomes. This method allows a different amount of evolution done to each of the topics, therefore we employed a previously utilized constant total generation number as the stopping criterion instead. This allows a more consistent comparison between the outcomes with a fixed number of evolution for all topics.

F. Constraint Handling

The DUC2002 MDS task requires a summary to include 200 words and the DUC2004 MDS task limits the summary to have 665 bytes. However, since the binary encoded candidates can be of any length, the summary can not exactly meet the constraint. This brings difficulty in measuring coverage and diversity. Therefore,we relax the 200 length constraint to allow a limit of $limit + maximumsentencelength - 1$. This allows us to fill in as much of the sentence as possible right before the limitation. For a fair comparison, we deduct the ROUGE score of our methods when comparing with the ones that followed strict constraints. The reduced ROUGE score for our method is as follows:

$$ROUGE_{reduced} = ROUGE * (1 - \frac{overflow}{limit}) \tag{9}$$

This metric reduces the ROUGE score as if the overflow part were not included in the evaluation.

TABLE III. THE DUC DATASETS

Description	DUC2002	DUC2004
Number of Topics	59	50
Number of Documents in each Topic	5∼15	9∼11
Number of Documents in Total	567	503
Data source	TREC	TDT
Summary length	200 terms	665 bytes

TABLE IV. STATISTICS OF THE DUC2002 DATASET

	Doc.	Sent.	Term(Unique)	Preprocessed Term(Unique)
average	9.6	260.0	5401.7(1844.7)	2931.6(1124.4)
max	15.0	614.0	11815.0(3387.0)	6134.0(1936.0)
min	5.0	111.0	2027.0(915.0)	1066.0(561.0)

TABLE V. STATISTICS OF THE DUC2004 DATASET

	Doc.	Sent.	term(unique)	Preprocessed Terms(Unique)
average	10.1	259.4	5342.6(1899.9)	2811.3(1144.3)
max	11.0	531.0	9502.0(3111.0)	4990.0(1940.0)
min	9.0	81.0	2174.0(859.0)	1255.0(526.0)

G. Solution Selection Strategy

However we need to select one summary to compare with other existing methods, and multiple non-dominated solutions are given as a result of multi-objective optimization. Thus, we need a method to select one summary from the candidates. Here, we suggest a solution selection method for multi-objective optimization among the Pareto optimal solutions. As defined in III, the objective functions take the average of the similarities. This implies that the coverage function does not consider the number of sentences included in the candidate summary. We define the solution ranking equation using the sentence length and the coverage value. The strategy is flexible since the solution selection is not done at a certain generation, but is done per topic basis. Final solution for the topic τ is defined as follows:

$$Solution(C_\tau) = \underset{c \in C_\tau}{\operatorname{argmax}} \, f_{coverage_k}(c) \cdot l(c) \tag{10}$$

where C is the candidate summary set over all the generations on topic τ, $l(c)$ is the number of terms in the candidate summary c.

VI. EVALUATION

A. Dataset

The experiments are performed using the DUC datasets in the years 2002 and 2004. The DUC2002 dataset contains 59 topics with 5 to 10 documents included in each topic. The task is to generate a summary with a maximum of 200 words. The DUC2004 dataset contains 50 topics with 10 to 11 documents included in each topic. The goal for the DUC2004 MDS task is to generate a summary with a maximum of 665 bytes. Table III shows the statistics of the DUC2002 and DUC2004 datasets. Table IV and Table V show specific statics of the two dataset. The preprocessed terms are acquired by removing stopwords and stemming the words.

B. Metric

Recall-Oriented Understudy for Gisting Evaluation (ROUGE) metric is used by DUC for text summarization competition. ROUGE metric is used to compare experimental results against the gold standard summaries generated by human annotators. The ROUGE score measures how much the words in the human reference summaries overlap with the machine-generated summaries in terms of the n-gram by counting overlapping words or a word sequence. There are four types of ROUGE scores. ROUGE-N, ROUGE-L, ROUGE-W, ROUGE-S. We use 3 of these measures for our evaluation and comparison with other methods. As for ROUGE-S, we use ROUGE-SU, which is its variation. In this research, ROUGE-1, ROUGE-2, and ROUGE-S(ROUGE-SU) are used for the comparison. ROUGE-N is the most basic method of comparison. It measures N-gram co-occurrence statistics. ROUGE-N is computed as follows:

$$ROUGE - N = \frac{\sum_{S \in sum_{ref}} \sum_{gram_N \in S} Count_m(gram_N)}{\sum_{S \in sum_{ref}} \sum_{gram_N \in S} Count(gram_N)} \tag{11}$$

where N is the length of the consecutive words used as a unit for the comparison. $Count_m(gram_N)$ is the maximum number of N-grams co-occurring in a candidate summary and the set of reference summaries. $Count(gram_N)$ is the total number of N-grams in the reference summarizes.

ROUGE-L is based on the longest common subsequence(LCS) metric. Here a summary sentence is viewed as a sequence of words. The rationale behind ROUGE-L is that the longer the LCS between the two summary sentences is, the more similar they are. We compare the union LCS matches between a reference summary sentence, r_i, and every candidate summary sentence, c_j.

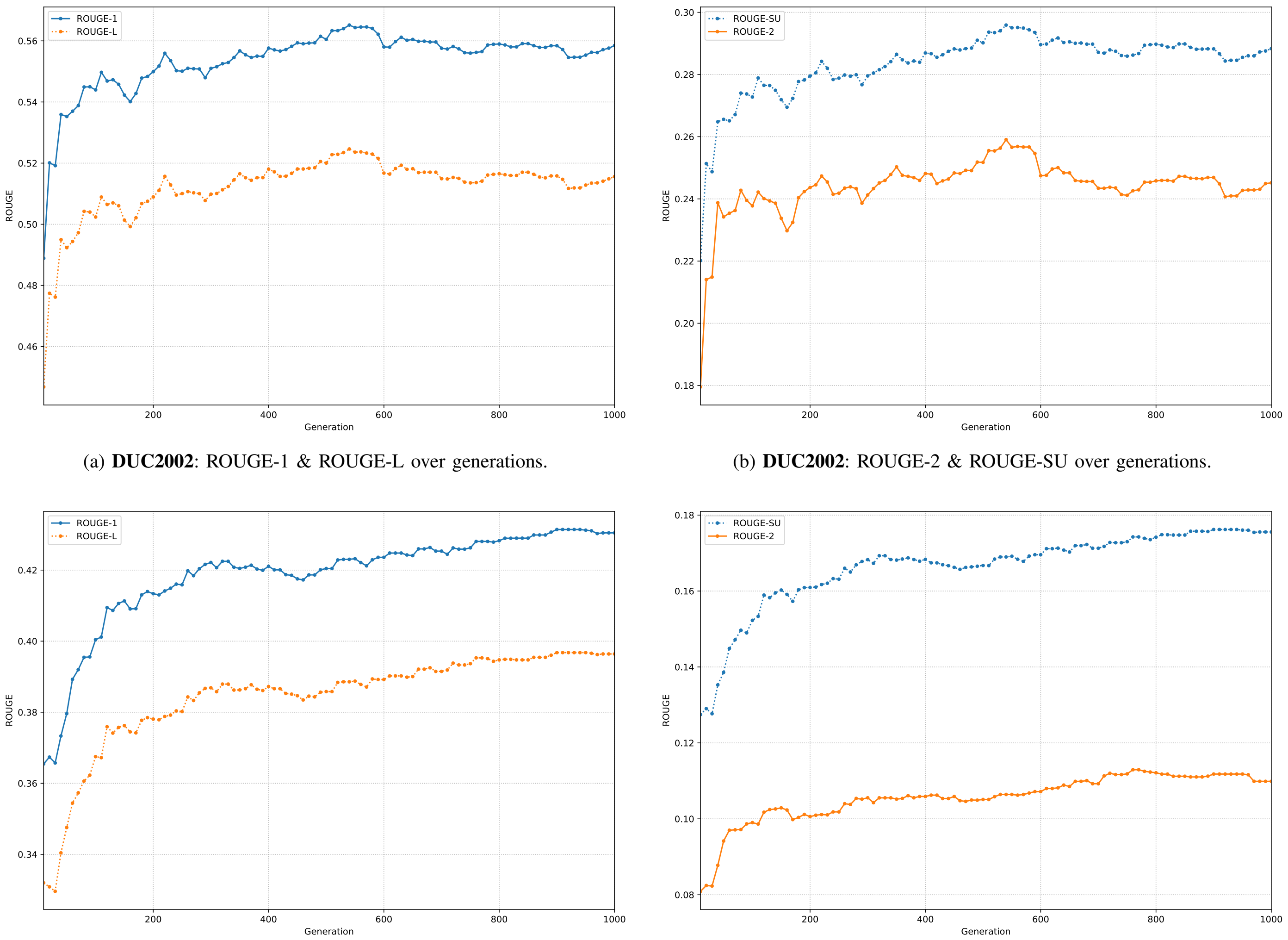
$u = [t_1, t_2, ..., t_m]$ and $v = [t_1, t_2, ..., t_n]$. $z = u \cup v$ with length θ. Let $LCS_{sequence}(x, y) = [t_1, t_2, ..., t_\theta]$ be the LCS word sequence representation between sentence u and sentence v. Given the sentence level LCS $R_{LCS}(R, S)$, reference summary of u sentences containing a total of m words and a candidate summary of v sentences containing a total of n words, ROUGE-L is computed as:

$$R_{LCS}(R, C) = \frac{\sum_{i=1}^{u} LCS_{\cup}(r_i, C)}{|R|} \tag{12}$$

ROUGE-S is a skip-bigram co-occurrence measure. It counts ordered bigrams allowing for arbitrary gaps. ROUGE-SN is used to parameterize maximum skip distance.

$$R_{skip2} = \frac{SKIP2(R, S)}{C(|S|, 2)} \tag{13}$$

where SKIP2(R, S) is the number of skip-bigrams that match between the reference summary R and the candidate summary S. ROUGE-S shows poor evaluation result if the skip-grams are not detected. So when the two sentences consist of the same words but with reversed order, ROUGE-S will assign 0 value to the pair. To overcome this potential problem, ROUGE-SU, an extension of ROUGE-S is used in our experiment. ROUGE-SU extends ROUGE-S by adding unigram as a counting unit in addition to ROUGE-S.



(a) **DUC2002**: ROUGE-1 & ROUGE-L over generations.

(b) **DUC2002**: ROUGE-2 & ROUGE-SU over generations.

(c) **DUC2004**: ROUGE-1 & ROUGE-L over generations.

(d) **DUC2004**: ROUGE-2 & ROUGE-SU over generations.

Fig. 1. Various ROUGE Scores over Generations of Evaluation on the DUC2002 & DUC2004 Datasets

Summarization approaches with non-optimization methods use ROUGE-1, ROUGE-2, ROUGE-L, ROUGE-SU for their evaluation. The ones with optimization methods use ROUGE-2 and ROUGE-L for their evaluation. For the comparison with the two groups, we adopt the list of metrics respectively. The meaning of each metric is quite important. ROUGE-L reflects the most important sequence therefore is more focused on the main contents when comparing the sentences. ROUGE-2 will count every bigram units, which may suffer from unimportant bigrams being a source of an inaccurate comparison.

VII. Result

A. Experiment Setup

The population is set to 50, and generation is set to 1000 for the existing methods as other evolutionary methods [9], [6], [4] did. The population is changed to 52 for the proposed method as NSGA-II contains a tournament selection stage where the population should be set to a multiple of four. This does not give advantage to our method, since as you can see in Fig. 1. The local optimum reaches before 1000th generation, and we apply our solution selection method described in Section V much earlier than the initially defined condition. Mutation probability is set to $5 \cdot \frac{1}{n_i}$ where n_i is number of sentence in topic i. Crossover probability is set to 0.8. Our initialization is done with a random number of sentences between 7 to 11.

TABLE VI. ROUGE-2 Comparison with Evolutionary Optimization Methods on the DUC2002 Dataset

Topic	GO[4]	ABCO[6]	**KE-B**	**KE-K**	**KE-B***	**KE-K***
d061j	0.305	0.365	0.377	0.343	0.427	0.343
d062j	0.200	0.342	0.100	0.115	0.105	0.188
d063j	0.275	0.272	0.257	0.276	0.307	0.276
d064j	0.233	0.308	0.339	0.469	0.339	0.469
d065j	0.182	0.198	0.149	0.104	0.333	0.270
d066j	0.181	0.290	0.263	0.234	0.263	0.234
d067f	0.260	0.356	0.217	0.345	0.330	0.394
d068f	0.496	0.444	0.393	0.360	0.521	0.479
d069f	0.232	0.240	0.147	0.168	0.406	0.178
d070f	0.262	0.305	0.241	0.380	0.245	0.388
Avg	**0.263**	**0.312**	**0.253**	**0.279**	**0.328**	**0.325**

TABLE VII. ROUGE-L Comparison with Evolutionary Optimization Methods on the DUC2002 Dataset

Topic	GO[4]	ABCO[6]	**KE-B**	**KE-K**	**KE-B***	**KE-K***
d061j	0.554	0.590	0.590	0.595	0.629	0.595
d062j	0.481	0.536	0.438	0.448	0.453	0.464
d063j	0.528	0.509	0.550	0.530	0.550	0.530
d064j	0.488	0.495	0.575	0.663	0.579	0.663
d065j	0.457	0.464	0.439	0.439	0.534	0.511
d066j	0.441	0.519	0.558	0.510	0.558	0.524
d067f	0.529	0.580	0.475	0.588	0.559	0.632
d068f	0.626	0.639	0.632	0.646	0.736	0.698
d069f	0.476	0.554	0.510	0.561	0.672	0.571
d070f	0.513	0.515	0.466	0.563	0.479	0.563
Avg	**0.509**	**0.540**	**0.523**	**0.554**	**0.575**	**0.575**

TABLE VIII. ROUGE scores from the DUC2002 Dataset

Methods	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-SU
KNET-EMO-K*	0.5680	0.2921	0.5328	0.3079
KNET-EMO-B*	0.5612	0.2853	0.5264	0.3020
KNET-EMO-K	0.5385	0.2409	0.4978	0.2810
KNET-EMO-B	0.5284	0.2295	0.4854	0.2731
WFS	0.4994	0.2582	0.4893	0.2874
OCDsum-SaDE	0.4990	0.2548	0.4708	0.2855
DUC	0.4987	0.2523	0.4680	0.2841
MCKP	0.4938	0.2511	0.4694	0.2855
WCS	0.4933	0.2484	0.4628	0.2789
BSTM	0.4881	0.2457	0.4552	0.2702
FGB	0.4851	0.2410	0.4508	0.2686
LexRank	0.4796	0.2295	0.4433	0.2620
Centroid	0.4538	0.1918	0.4324	0.2363
NMF	0.4459	0.1628	0.4151	0.2169
LSA	0.4308	0.1502	0.4051	0.2023
Random	0.3878	0.1196	0.3771	0.1852
CRSum-SF	0.3890	0.1028	·	·

TABLE IX. ROUGE Scores from the DUC2004 Dataset

Methods	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-SU
KNET-EMO-K*	0.4281	0.1250	0.3967	0.1891
KNET-EMO-K	0.4094	0.1105	0.3760	0.1746
KNET-EMO-B*	0.4080	0.1046	0.3786	0.1662
KNET-EMO-B	0.3958	0.0985	0.3643	0.1626
WFS	0.3933	0.1121	0.3960	0.1354
OCDsum-SaDE	0.3954	0.0969	0.3927	0.1367
CRSum-SF	0.3953	0.1060	·	·
DUCbest	0.3822	0.0922	0.3869	0.1323
MCKP	0.3864	0.0924	0.3892	0.1333
WCS	0.3987	0.0961	0.3893	0.1353
BSTM	0.3907	0.0901	0.3880	0.1322
FGB	0.3872	0.0812	0.3842	0.1296
LexRank	0.3784	0.0857	0.3753	0.1310
Centroid	0.3673	0.0738	0.3618	0.1251
NMF	0.3675	0.0726	0.3675	0.1292
LSA	0.3415	0.0654	0.3497	0.1195
Random	0.3227	0.0639	0.3488	0.1197

We compare the proposed model with two different groups of methods. One group of methods is the ones that can be compared using the DUC2002 and DUC2004 datasets and the other group is the ones with evolutionary optimization algorithms where only topics from 61 to 70 in the DUC2002 dataset are considered in the literature [6], [37]. Each of the methods for the comparison is shown in Table I. Some of the methods are taken from the list in Alguliev et. al. [9], and we extend the list with other methods not listed in the paper.

B. Evolutionary Multi-objective Optimization Results

We first verify whether the multi-objective optimization formulation of the summarization problem is valid. Fig. 1 shows the ROUGE results over generations. We can observe that the proposed algorithm generally produces an improved result as the generation passes. Note that the improvement is not a monotonic increase. Before reaching generation 200, all four graphs show a valley, a decrease of ROUGE score compared to the previous generations. One of the reasons for this phenomenon is that there is a length limitation. To find a solution that maximizes the objective function, the genes are continuously changed through mutation and crossover. Since the objective functions are defined to consider the average similarity of the sentences in a candidate solution, certain generations may contain the ones with a smaller number of sentences which may be given a low ROUGE score as the summary does not fully utilize the 200 word limit. Considering a little fluctuation as an inevitable variance throughout the evolution process, we can see the evolutions are generally improving towards higher ROUGE scores.

Another characteristic to discuss in Fig. 1 is that for the DUC2002 dataset, the peak is reached quite faster than the results of the DUC2004 dataset. Our further experiment using 200 populations and 2000 generations showed that the performance improves as the generation continues. The graphs resemble a fractal structure, where the local pattern is similar to the global pattern. But since we need to find the best solution among the candidate solutions, we propose a solution selection strategy to select solutions from various generations for each topic as explained in Section V.

C. Comparison with Multi-objective Methods

Table VI and Table VII show the ROUGE score comparison between the proposed Knowledge-based Named Entity Topic Construction using Evolutionary Multi-objective Optimization(KNET-EMO, shown as KE in tables) and its variations along with two existing multi-objective optimization methods. KNET-EMO-B(KE-B) is the base method without using the knowledge-based topic construction approach and only using the improved NSGA-II. KNET-EMO-K(KE-K) is the proposed method that reflects all of our main contributions. (*) are the ROUGE scores of the solutions manually selected for their summarization qualities, showing that the multi-objective optimization has successfully generated high performing results.

From the average scores in Table VI and Table VII, ROUGE-L shows superior performance compared to ROUGE-2. This implies the proposed method successfully found the sentences that contain the core contents of the original documents in a more relaxed manner. Low ROUGE-2 and high ROUGE-L suggest that it is robust when we need to generate summaries from the real-world text. This is because it can generate the summaries regardless of the exact sequence of the terms while allowing other terms to appear in between, which is a more human way of selecting the sentences.

A high ROUGE-2 score can be manually achieved by taking the sentences from the answer set and select the ones from them because this will lead to the same sequence which leads to high ROUGE-2. But high ROUGE-L cannot be generated in this manner. This can be only achieved by selecting the sentences that reflect the original document, but not the same sequence of terms in the answer set. The generally observed pattern of high ROUGE-2 and low ROUGE-L scores in other research is more adapted or even overfitted to the given task alone. The proposed method showed the opposite result, indicating it can be generalized to other text corpora, unlike the existing methods.

Another factor for this result is the difference in the sentence parsing method. If the sentence parsing is more consistent with human judgment, it will not have a negative influence on the performance. But if the sentence parsing is not synchronous with human judgment it can have a negative influence on the bigram measure.

The best scores, KNET-EMO-B*(the base method) and KNET-EMO-K*(the knowledge-based method) are both 0.575 from Table VII, but the solution selected scores KNET-EMO-B and KNET-EMO-K differ. We can understand it as a knowledge-based coverage measure combined with our solution selection method is more suitable in finding the solutions regarding human judgment.

D. Comparison with General Methods

Table VIII and Table IX show the comparison to the generally known text summarization methods. We compare the results using ROUGE-1, ROUGE-2, ROUGE-L, and ROUGE-SU as previously explained. Since we compare against evolutionary algorithms, we defined the tolerance threshold and used the value for the summary length limit as in other evolutionary optimization methods. This is to avoid cutting off the sentence at the limit point but include the ones when the constraint limit is violated. Otherwise, we cannot fill the content within the length limit. Here for the comparison with the general methods, we normalize our results with the ratio between the limit and the tolerance threshold. This is to deduct the ROUGE score that the proposed method may have gained by having the tolerance threshold.

According to the tolerance threshold defined in Section V, the average of the maximum length sentences from each topic is 49 bytes for the DUC2002 and 57.78 bytes for the DUC2004. For DUC2002 we need to convert these 49 bytes into the word count. Since the average word length of the DUC2002 corpus is 5.5 and we need to add a 1 byte for a space for each word, the average word tolerance for the DUC2002 dataset is $\frac{49}{6.5} = 7.54$. This means that our method accepted the candidate summaries that do not violate more than or equal to the average of 7.54 words for the DUC2002 datasets and 57.78 bytes for the DUC2004 datasets. For a fair comparison, we assume that the solutions in the proposed method fully utilize this advantage and normalize it with the ratio of the tolerance from the total length. So for DUC2002, $\frac{49}{200} = 3.8\%$ is deducted from the ROUGE score of our method. For DUC2004, 57.78/665 = 8.7% is deducted from the ROUGE score of our method. So we compete using 96.2% of the ROUGE scores from the proposed method for the DUC2002 dataset against the methods listed in the result Table VIII, and 91.3% of the ROUGE scores from the proposed method for the DUC2004 dataset against the other methods in result Table IX.

Table VIII and Table IX both shows that the proposed method outperforms existing methods on both the DUC2002 and DUC2004 datasets. The values in the (*) is the score of the manually selected best solution from the candidate summary pool. They show that although our selection method can find nice solutions, they are not the best among the candidate solutions found from the evolution. This confirms that our problem formulation of text summarization as a multi-objective optimization is validated as the candidates contain solutions with very high scores. Note that KNET-EMO-K outperforms the best candidate(KNET-EMO-B*) using the base method. This confirms that the proposed knowledge-based named entity topic construction method outperforms the base method using only the evolutionary multi-objective optimization.

We have observed from the experiment results that after reaching the ROUGE performance peak, as the evolution continues, the ROUGE score decreases after certain generations as explained previously. The objective function is an approximation of the ideal objective as human summarizers use heuristics to generate summaries. Although human summarizers aim to generate summaries that are to a certain extent optimized, it needs not to be as optimal as the result of the automatic summarizers. Thus, finding an optimal stopping point is one of the future research topics to work on.

VIII. Conclusion

In this research, the authors proposed a multi-objective optimization approach for the automatic text summarization. On top of the framework based on an evolutionary multi-objective optimization, a novel method to utilize a knowledge base is proposed. Combining knowledge base utilization method and the improvement algorithm on the evolutionary multi-objective optimization steps, the evaluation results have shown that the proposed method not only out-performs previous research on text summarization but also shows better performance compared to the recent research on evolutionary multi-objective optimization techniques. For future works, many-objective optimization formulation of the text summarization is considered where the framework can take more objective functions to pursue more precise optimization. As discussed previously, there are further research opportunities with regards to early stopping method so that we can find the level of optimization the human summarizers achieve. Finally, an extension of the knowledge base methods are to be studied for a better solution selection method given multiple Pareto optimal candidate solutions.

References

[1] R. Baeza-Yates, B. Ribeiro-Neto *et al.*, *Modern information retrieval.* ACM press New York, 1999, vol. 463.

[2] T. Berners-Lee, J. Hendler, and O. Lassila, "The semantic web," *Scientific american*, vol. 284, no. 5, pp. 34–43, 2001.

[3] M. Maybury, *Advances in automatic text summarization.* MIT press, 1999.

[4] H. H. Saleh, N. J. Kadhim, and B. A. Attea, "A Genetic Based Optimization Model for Extractive Multi-Document Text Summarization," *Iraqi Journal of Science*, vol. 56, no. 2B, pp. 1489–1498, 2015, 00003.

[5] C. Jung, R. Datta, and A. Segev, “Multi-document Summarization Using Evolutionary Multi-objective Optimization,” in *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, ser. GECCO ’17. New York, NY, USA: ACM, 2017, pp. 31–32, 00002 event-place: Berlin, Germany.

[6] J. M. Sanchez-Gomez, M. A. Vega-Rodríguez, and C. J. Pérez, “Extractive multi-document text summarization using a multi-objective artificial bee colony optimization approach,” *Knowledge-Based Systems*, vol. 159, pp. 1–8, Nov. 2018, 00008.

[7] Y. Gong and X. Liu, “Generic text summarization using relevance measure and latent semantic analysis,” in *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2001, pp. 19–25, 00861.

[8] H. Takamura and M. Okumura, “Text summarization model based on maximum coverage problem and its variant,” in *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, 2009, pp. 781–789, 00144.

[9] R. M. Alguliev, R. M. Aliguliyev, and N. R. Isazade, “Multiple documents summarization based on evolutionary optimization algorithm,” *Expert Systems with Applications*, vol. 40, no. 5, pp. 1675–1689, 2013, 00025.

[10] H. P. Luhn, “The automatic creation of literature abstracts,” *IBM Journal of research and development*, vol. 2, no. 2, pp. 159–165, 1958, 02891.

[11] K. R. McKeown, J. L. Klavans, V. Hatzivassiloglou, R. Barzilay, and E. Eskin, “Towards multidocument summarization by reformulation: Progress and prospects,” in *Proceedings of the Sixteenth National Conference on Artificial Intelligence and the Eleventh Innovative Applications of Artificial Intelligence Conference Innovative Applications of Artificial Intelligence*, ser. AAAI ’99/IAAI ’99. Menlo Park, CA, USA: American Association for Artificial Intelligence, 1999, pp. 453–460.

[12] J. Goldstein, V. Mittal, J. Carbonell, and M. Kantrowitz, “Multi-document Summarization by Sentence Extraction,” in *Proceedings of the 2000 NAACL-ANLPWorkshop on Automatic Summarization - Volume 4*, ser. NAACL-ANLP-AutoSum ’00. Stroudsburg, PA, USA: Association for Computational Linguistics, 2000, pp. 40–48.

[13] K. Ganesan, C. Zhai, and J. Han, “Opinosis: A graph based approach to abstractive summarization of highly redundant opinions,” in *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, 2010, pp. 340–348, 00336.

[14] L. Bing, P. Li, Y. Liao, W. Lam, W. Guo, and R. J. Passonneau, “Abstractive multi-document summarization via phrase selection and merging,” *arXiv preprint arXiv:1506.01597*, 2015, 00026.

[15] S. Harabagiu and F. Lacatusu, “Using topic themes for multi-document summarization,” *ACM Transactions on Information Systems*, vol. 28, no. 3, pp. 1–47, Jun. 2010.

[16] K. Sarkar, “Syntactic trimming of extracted sentences for improving extractive multi-document summarization,” *Journal of Computing*, vol. 2, no. 7, pp. 177–184, 2010, 00013.

[17] J. Carbonell and J. Goldstein, “The use of MMR, diversity-based reranking for reordering documents and producing summaries,” in *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 1998, pp. 335–336.

[18] H. Zha, “Generic summarization and keyphrase extraction using mutual reinforcement principle and sentence clustering,” in *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2002, pp. 113–120, 00286.

[19] D. R. Radev, H. Jing, M. Styś, and D. Tam, “Centroid-based summarization of multiple documents,” *Information Processing & Management*, vol. 40, no. 6, pp. 919–938, 2004.

[20] G. Erkan and D. R. Radev, “Lexrank: Graph-based lexical centrality as salience in text summarization,” *Journal of artificial intelligence research*, vol. 22, pp. 457–479, 2004, 02065.

[21] J. Otterbacher, G. Erkan, and D. R. Radev, “Biased LexRank: Passage retrieval using random walks with question-based priors,” *Information Processing & Management*, vol. 45, no. 1, pp. 42–54, 2009, 00063.

[22] M. Al-Dhelaan, “StarSum: A Simple Star Graph for Multi-document Summarization,” in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR ’15. New York, NY, USA: ACM, 2015, pp. 715–718, 00003.

[23] F. Wei, W. Li, Q. Lu, and Y. He, “A document-sensitive graph model for multi-document summarization,” *Knowledge and information systems*, vol. 22, no. 2, pp. 245–259, 2010, 00053.

[24] X. Wan and J. Yang, “Multi-document summarization using cluster-based link analysis,” in *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2008, pp. 299–306, 00217.

[25] J.-Y. Yeh, H.-R. Ke, and W.-P. Yang, “iSpreadRank: Ranking sentences for extraction-based summarization using feature weight propagation in the sentence similarity network,” *Expert Systems with Applications*, vol. 35, no. 3, pp. 1451–1462, 2008, 00034.

[26] S. Chen, M. Huang, and Z. Lu, “Summarizing Documents by Measuring the Importance of a Subset of Vertices Within a Graph,” in *Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology - Volume 01*, ser. WI-IAT ’09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 269–272, 00005.

[27] C. Shen and T. Li, “Multi-document summarization via the minimum dominating set,” in *Proceedings of the 23rd International Conference on Computational Linguistics*, ser. COLING ’10. Stroudsburg, PA, USA: Association for Computational Linguistics, 2010, pp. 984–992. [Online]. Available: http://dl.acm.org/citation.cfm?id=1873781.1873892

[28] X. Han, T. Lv, Q. Jiang, X. Wang, and C. Wang, “Text summarization using sentence-level semantic graph model,” in *2016 4th International Conference on Cloud Computing and Intelligence Systems (CCIS)*. IEEE, 2016, pp. 171–176.

[29] R. Barzilay and K. R. McKeown, “Sentence fusion for multidocument news summarization,” *Computational Linguistics*, vol. 31, no. 3, pp. 297–328, 2005, 00265.

[30] A. R. Deshpande and L. Lobo, “Text summarization using Clustering technique,” *International Journal of Engineering Trends and Technology*, vol. 4, no. 8, 2013, 00007.

[31] K. Sarkar, “Sentence clustering-based summarization of multiple text documents,” *International Journal of Computing Science and Communication Technologies*, vol. 2, no. 1, pp. 325–335, 2009, 00035.

[32] A. Kogilavani and P. Balasubramani, “Clustering and feature specific sentence extraction based summarization of multiple documents,” *International Journal of Computer Science Information Technology*, vol. 2, no. 4, pp. 99–111, 2010, 00022.

[33] J.-H. Lee, S. Park, C.-M. Ahn, and D. Kim, “Automatic generic document summarization based on non-negative matrix factorization,” *Information Processing & Management*, vol. 45, no. 1, pp. 20–34, 2009, 00097.

[34] D. Wang, S. Zhu, T. Li, and Y. Gong, “Multi-document Summarization Using Sentence-based Topic Models,” in *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, ser. ACLShort ’09. Stroudsburg, PA, USA: Association for Computational Linguistics, 2009, pp. 297–300, 00158 event-place: Suntec, Singapore.

[35] D. Wang, S. Zhu, T. Li, Y. Chi, and Y. Gong, “Integrating document clustering and multidocument summarization,” *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 5, no. 3, p. 14, 2011, 00069.

[36] D. Wang and T. Li, “Weighted consensus multi-document summarization,” *Information Processing & Management*, vol. 48, no. 3, pp. 513–523, 2012, 00040.

[37] D. H. H. Saleh, “Genetic Based Optimization Models for Enhancing Multi-Document Text Summarization,” 2016, 00000.

[38] P. Ren, Z. Chen, Z. Ren, F. Wei, J. Ma, and M. de Rijke, “Leveraging contextual sentence relations for extractive summarization using a neural attention model,” in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 2017, pp. 95–104, 00040.

[39] L. Hennig and D. A. I. Labor, “Topic-based Multi-Document Summarization with Probabilistic Latent Semantic Analysis.” in *RANLP*, 2009, pp. 144–149, 00065.

[40] A. Nenkova and R. Passonneau, “Evaluating content selection in summarization: The pyramid method,” in *Proceedings of the human language technology conference of the north american chapter of the association for computational linguistics: Hlt-naacl 2004*, 2004, 00473.

[41] C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in *Text Summarization Branches Out*. Barcelona, Spain: Association for Computational Linguistics, Jul. 2004, pp. 74–81.

[42] L. Hennig, E. W. De Luca, and S. Albayrak, "Learning summary content units with topic modeling," in *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*. Association for Computational Linguistics, 2010, pp. 391–399, 00005.

[43] S. Lee, S. Belkasim, and Y. Zhang, "Multi-document text summarization using topic model and fuzzy logic," in *International Workshop on Machine Learning and Data Mining in Pattern Recognition*. Springer, 2013, pp. 159–168, 00009.

[44] N. K. Nagwani, "Summarizing large text collection using topic modeling and clustering based on MapReduce framework," *Journal of Big Data*, vol. 2, no. 1, p. 1, 2015, 00007.

[45] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17. Red Hook, NY, USA: Curran Associates Inc., 2017, p. 6000–6010.

[46] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186. [Online]. Available: https://aclanthology.org/N19-1423

[47] Y. Liu, "Fine-tune BERT for Extractive Summarization," *arXiv:1903.10318 [cs]*, Mar. 2019, 00000 arXiv: 1903.10318.

[48] M. Yousefi-Azar and L. Hamey, "Text summarization using unsupervised deep learning," *Expert Systems with Applications*, vol. 68, pp. 93–105, 2017, 00000.

[49] A. Abuobieda, N. Salim, M. S. Binwahlan, and A. H. Osman, "Differential evolution cluster-based text summarization methods," in *2013 INTERNATIONAL CONFERENCE ON COMPUTING, ELECTRICAL AND ELECTRONIC ENGINEERING (ICCEEE)*. IEEE, 2013, pp. 244–248.

[50] R. L. Cilibrasi and P. M. B. Vitanyi, "The Google Similarity Distance," *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, no. 3, pp. 370–383, Mar. 2007, 01525.

[51] L. Huang, Y. He, F. Wei, and W. Li, "Modeling Document Summarization as Multi-objective Optimization," in *2010 Third International Symposium on Intelligent Information Technology and Security Informatics*, Apr. 2010, pp. 382–386, 00024.

[52] K. Sekaran, P. Chandana, J. R. V. Jeny, M. N. Meqdad, and S. Kadry, "Design of optimal search engine using text summarization through artificial intelligence techniques," *Telkomnika*, vol. 18, no. 3, pp. 1268–1274, 2020.

[53] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *IEEE transactions on evolutionary computation*, vol. 6, no. 2, pp. 182–197, 2002.

[54] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, "DBpedia: A Nucleus for a Web of Open Data," in *The Semantic Web*, ser. Lecture Notes in Computer Science, K. Aberer, K.-S. Choi, N. Noy, D. Allemang, K.-I. Lee, L. Nixon, J. Golbeck, P. Mika, D. Maynard, R. Mizoguchi, G. Schreiber, and P. Cudré-Mauroux, Eds. Springer Berlin Heidelberg, 2007, pp. 722–735, 00000.

[55] F. M. Suchanek, G. Kasneci, and G. Weikum, "Yago: a core of semantic knowledge," in *Proceedings of the 16th international conference on World Wide Web*. ACM, 2007, pp. 697–706.

[56] P. N. Mendes, M. Jakob, A. García-Silva, and C. Bizer, "DBpedia spotlight: shedding light on the web of documents," in *Proceedings of the 7th international conference on semantic systems*. ACM, 2011, pp. 1–8, 01019.