Preprints of the Federated Conference on
Computer Science and Information Systems pp. 69–76

Application of Ant-Colony Optimisation to Compute Diversified Entity Summarisation on Semantic Knowledge Graphs

Witold Kosiński[1,2], Tomasz Kuśmierczyk[3], Paweł Rembelski[1], Marcin Sydow [1,3]
[1] Department of Computer Science, Polish-Japanese Institue of Information Technology,
ul. Koszykowa 86, 02-008 Warszawa
[2] Institute of Mechanics and Applied Computer Science, Kazimierz-Wielki University,
ul. Chodkiewicza 30, 85-072 Bydgoszcz
[3] Institutue of Computer Science, IPI PAN, Warszawa, ul. Ordona
Email: wkos@pjwstk.edu.pl, t.kusmierczyk@phd.ipipan.waw.pl, rembelski@pjwstk.edu.pl, msyd@poljap.edu.pl

***Abstract*—We present ant colony optimisation approach, enriched with a novel self-adaptation mechanism, applied to solve DIVERSUM Problem that consits of generating a small diversified entity summarisation in a knowledge graph. The recently proposed DIVERSUM problem is viewed in this paper in a novel way as a NP-hard combinatorial optimisation problem. The presented preliminary experimental results indicate superiority of this approach to the previously proposed solutions to the DIVERSUM problem.**

***Index Terms*—Ant Colony Optimisation, Diversified Entity Summarisation, Max Sum Dispersion, Semantic Knowledge Graphs**

I. Introduction

TRADITIONAL optimisation techniques and optimisation methods are related to different types of problems. Some of them deal with constraint handling by using penalty methods, however, they often get stuck in local optima. Moreover, they usually need knowledge of first/second order derivatives of objective functions and constraints.

Hence one looks for more sophisticated methods, especially in cases when: Search space is discrete, discontinuous, non-convex, etc. or objective functions and constraints are non-differentiable or computationally expensive.

If we look at the branch of computer science, called *computational intelligence* then we find here 3 main divisions: nature inspired algorithms, fuzzy logic systems, neural networks.

In the last quarter of the previous century a dozen or more different optimisation algorithms have been proposed that are nature-based methods. One can list some of them: genetic, or more general – evolutionary algorithms, that are based on the Darwin's theory of evolution, swarm optimisation techniques, that copy the swarm intelligence, and they include: ant colony systems, particle swarm optimisation; and simulated annealing methods, that is based on the process of steel production; generalised extremal optimisation, that is based on point-wise equilibrium phenomenon, [1], in which the evolution happens step-wise in contrast to the Darwin continuous evolution, and many more methods.

In this paper we present application of the Ant Colony Optimisation (ACO) approach (Section II), to solve the DIVERSUM problem, i.e. a recently [2] introduced hard optimisation problem of Diversified Entity Summarisation on Knowledge Graphs (Section III). The DIVERSUM problem consists in selecting a small set S of arcs incident to a node x in a knowledge graph so that S gives a good "summary" of the node x. In this paper we present a diversity-aware objective measure so that the DIVERSUM problem is viewed as a combinatorial optimisation problem that is NP-hard.

Then, we report preliminary experimental results (Section IV) run on real data that illustrate the applicability of the presented ACO optimisation technique on the DIVERSUM problem and show that the peformance is higher compared with some previous solutions to the problem.

Ant Colony Optimisation approach was described in [3] and subsequently in a large number of publications. A more recent examples are [4] and [5] where also its application to a hard computational problem such as TSP was presented.

The problem of Diversified Entity Summarisation on Knowledge Graphs has been recently described and studied in [2]. In this paper, we study this problem from the optimisation perspective as optimising a properly defined diversity-aware objective function (Section III-C).

A. Contributions

The contributions of this paper include:

- novel self-adaptation mechanism for one of the parameters of ant colony optimisation algorithm, presented in Section II-G
- formulating the DIVERSUM problem as a combinatorial optimisation one, via defining a novel objective function (Section III-C)
- application of ant colony optimisation approach to such defined optimisation problem (Section III)
- promising preliminary experimental results on real data (Section IV)

©2013

II. Ant Colony Optimisation Algorithm

Ant System (AS), has been proposed by M. Dorigo in 1992, is a nature-based heuristic which tackles a range class of the real-life optimisation problems by information cooperation (i.e. *pheromone trials* evolution) inside a set $A = \{a_1, a_2, \ldots, a_{\mathfrak{m}}\}$ of $\mathfrak{m} \geq 1$ abstract agents called *ants*.

AS algorithm is an iterative method in which the three main rules are repeated sequentially:

- *Neighbour Choosing Rule* (AS-NCR) – which defines the probability of adding a new element to the solution being constructed,
- *Solution Construction Rule* (AS-SCR) – which stands for a process of constructing a complete solution for an input problem (AS-NCR is used inside),
- *Pheromone Update Rule* (AS-PUR) – which determines how the ants exchanges local information to build a global description of a space being under optimisation process.

One extra rule is often added, namely:

- *Pheromone Evaporation Rule (AS-PER)* – which defines quantity of evaporation (loss) of global information written down in pheromone trials.

Ant System heuristic schema:

while (stop condition is false) **do**
  for every ant a_i **do**
    construct solution with AS-SCR (AS-NCR inside)
      for every ant a_i **do**
      update pheromone trails with AS-PUR
  evaporate pheromone trials with AS-PER

Let us list the main components of the Ant System.

A. Neighbour Choosing Rule

Neighbour Choosing Rule is based on the probability of adding to constructed solution a new element x_j, just after x_i is previously added, is equal to

$$p_{i,j} = \frac{(\tau_{i,j})^{\alpha}(\eta_{i,j})^{\beta}}{\sum_{k \in \Theta}(\tau_{i,k})^{\alpha}(\eta_{i,k})^{\beta}}$$

where Θ is a set of indexes of all actually reachable elements and:

- $\tau_{i,j}$ – is pheromone trail value connected with an action of choosing element x_j just after an element x_i is chosen,
- $\eta_{i,j}$ – is heuristic value connected with an action of choosing element x_j just after an element x_i is chosen (this coefficient is defined within an input problem),
- α, β – are parameters that control the relative weight of pheromone trail and heuristic value; they need to be adapted.

One of the results of our paper is to define and implement a self-adaptation algorithm for the parameter α; here we assume $\beta = 0$.

B. Pheromone Update Rule

In Pheromone Update Rule for ant a_k the following action is executed

$$\tau_{i,j} \leftarrow (1-\rho)\,\tau_{i,j} + \Delta_k$$

where Δ_k is a value of fixed function of solution quality, and the following Pheromone Evaporation Rule has been already incorporated

$$\tau_{i,j} \leftarrow (1-\rho)\,\tau_{i,j}$$

where ρ is a fixed evaporation coefficient.

In our paper we are generalise the Ant System in the form of a *Discrete Ant System*. In its heuristic we face with a collective work, performed by a discrete number of objects and with the use of a discrete number of functions (tools), that may possess only a discrete number of different values. In this way the set of information collection by individuals - ants in order to construct individual solutions by the use of global knowledge of $\mathfrak{m}$ ants from the set A, is finite, as well. Now we define the optimisation problem for a simple ant system.

By an *optimisation problem* we understand a given quadruple $(\Sigma, \mathcal{R}, \Delta, \|\cdot\|)$, where:

- $\Sigma = \{x_1, x_2, \ldots, x_{\mathfrak{n}}\}$ – is a finite set of $\mathfrak{n}$ indexed *objects* (symbols),
- $\mathcal{R} \subset \Sigma^*$ – is a finite set of $\mathfrak{r}$ indexed *solutions* (words),
- $\Delta : \Sigma^* \to \{0,1\}$ – is a *solution acceptance function* such, that

$$\Delta(\omega) = \begin{cases} 1, & \text{if } \exists\left(\omega' \in \Sigma^*, \omega'' \in \mathcal{R}\right)\left(\omega \circ \omega' = \omega''\right) \\ 0, & \text{in other case} \end{cases}$$

- $\|\cdot\| : \mathcal{R} \to \mathbb{R}_+ \cup \{0\}$ – is a *solution quality function*, where we deal here with the minimisation problem for the quality function.

Notice that this definition of the optimisation problem is suitable for a wide range of real-world computation problems including NP-hard combinatorial problems. For example in the traveling salesman problem (TSP) the set Σ contains all labels of vertices of the graph under consideration and $\mathcal{R}$ is the set of all permutations of Σ. In a discrete knapsack problem, on the other hand, Σ is the set of all objects to be put into knapsack, while $\mathcal{R}$ is the set of words-solutions representing filling methods of the knapsack, taking into account all combinations and constrains put on the number of objects.

The solution $\omega^* \in \mathcal{R}$ is an *optimal* one if

$$\forall(1 \leq i \leq \mathfrak{r})\,(\|\omega_i\| \geq \|\omega^*\|)$$

Let $\mathcal{R}^* \subset \mathcal{R}$ denote the set of all optimal solutions. Our task in the optimisation problem $(\Sigma, \mathcal{R}, \Delta, \|\cdot\|)$ is to find any optimal solution $\omega^* \in \mathcal{R}^*$.

C. Pheromone Trail: Basic Variant

The value of saturation of pheromone trail left by an ant is upper bounded by $\tau_{max} \in \mathbb{N}_+$. We assume that

$$\mathbb{H} = \{1, 2, \ldots, \tau_{max}\}$$

is a *set of possible values* of saturation of pheromone trail. Then $F \in \mathbb{H}^{\mathfrak{n}}$ is a column vector of size $\mathfrak{n}$, which defines the *saturation level* of pheromone trail, where $F[i]$ is a value connected with an ant's possibility of choosing an object x_i. Next

$$\mathcal{F} = \{F_1, F_2, \ldots, F_{\mathfrak{f}}\}$$

is a finite set of all possible indexed vectors of saturation level. Notice that

$$\mathfrak{f} = (\tau_{max})^{\mathfrak{n}}$$

In our DAS algorithm we introduce a new object the matrix $H \in \mathbb{H}^{\mathfrak{n}\times\mathfrak{n}}$ of size $\mathfrak{n} \times \mathfrak{n}$, which determines the *saturation level* of pheromone trail, where $H[i, j]$ is a value connected with an ant's moving action from an object x_i to an object x_j. Thus

$$\mathcal{H} = \{H_1, H_2, \ldots, H_{\mathfrak{h}}\}$$

is a finite set of all possible indexed matrices of saturation level. Similarly

$$\mathfrak{h} = (\tau_{max})^{\mathfrak{n}^2}$$

D. Pheromone Trail – Probabilistic

Now let $\widehat{\mathbb{H}}$ be a set of discrete probabilistic values over set $\mathbb{H}$ of all possible values of saturation of pheromone trail, namely

$$\widehat{\mathbb{H}} = \left\{ \frac{a}{b} : a \in \{1, 2, \ldots, \tau_{max}\}, b \in \{1, 2, \ldots, \mathfrak{n} \cdot \tau_{max}\} \right\}.$$

where $0 < \frac{a}{b} \leq 1$. Then, we define the following *reduction function* Ω for the column vectors of set $\mathcal{F}$ where $\mathcal{F} \times \mathcal{R} \times \mathbb{R}_+ \to \widehat{\mathbb{H}}^{\mathfrak{n}}$ and $\Omega(F, \omega, \alpha) = \widehat{F}$ is a stochastic column vector, that according to the rule

$$\widehat{F}[i] = \begin{cases} \dfrac{F[i]^{\alpha}}{\sum\limits_{\{j:\Delta(\omega x_j)=1\}} F[j]^{\alpha}} & \text{if} \quad \Delta(\omega x_i) = 1 \\ 0 & \text{in other case} \end{cases}, \tag{1}$$

for α being a coefficient which describes an individual behavior of a single ant.

E. Neighbour Choosing Rule

The action of Neighbour Choosing Rule is governed by the nondeterministic function, called here $NCR : \mathcal{F} \times \Sigma^* \to \Sigma$ which at given $F \in \mathcal{F}$ and actually constructed word $\sigma \in \Sigma^*$ assume values x_i with the probability given by Eq.(1). It can be formulated as follows:

Remark 1. For any column vector $F \in \mathcal{F}$ and any word $\sigma \in \Sigma^*$ and arbitrary symbol $x \in \Sigma$ the following takes place

$$Pr(NCR(F, \sigma) = x) \in (0, 1), \text{ if } \Delta(\omega x) = 1, \text{and} \tag{2}$$

$$Pr(NCR(F, \sigma) = x) = 0 \text{ in other cases}.$$

This is the basis of the next nondeterministic evolution mechanism in DAS, related to the solution construction rule *SCR*, where $SCR : \mathcal{F} \times \mathcal{H} \to \mathcal{R}$. This rule is the composition of a sequence of independent events generated by a multiple application of the mechanism NCR, unless an internal stop condition is meet. In the reality after the initiation of the algorithm and the formation of the first element of the word-solution, according to the column vector F and

$$\omega \leftarrow NCR(F, \varepsilon), \tag{3}$$

at some i-th stage of the iteration process we obtain a word $\omega = x_{l_1}, x_{l_2}, \ldots, x_{l_i}$, say, then as the result of the l_i-th column vector of the matrix H, denoted here by $H[l_i, \cdot]$, we determine the next element

$$x_{l_{i+1}} \leftarrow NCR(H[l_i, \cdot], \omega) \tag{4}$$

with the probability given by the property of the rule NCR. Then we add the next symbol xl_{i+1} at the end of the word ω, i.e.

$$\omega \leftarrow \omega x_{l_{i+1}}. \tag{5}$$

From this description we can formulate the following remark concerning the probabilistic nature of SCR.

Remark 2. For any column vector $F \in \mathcal{F}$, an arbitrary matrix $H \in \mathcal{H}$ and a word-solution $\omega \in \mathcal{R}$ of the form $\omega = x_{l_1}, x_{l_2}, \ldots, x_{l_r}$, the following is true:

$$Pr(SCR(F, H) = \omega) =$$

$$= Pr(NCR(F, \varepsilon) = x_{l_1}) \cdot \prod_{i=1}^{r-1} Pr(NCR(H[l_i, \cdot], x_{l_1}, x_{l_2}, \ldots, x_{l_i}), \tag{6}$$

which means, by Remark 1, that

$$Pr(SCR(F, H) = \omega) \in (0, 1). \tag{7}$$

The last statement has a fundamental meaning for our nondeterministic algorithm: each word-solution $\omega \in \mathcal{R}$, and by this each optimal word-solution from $\mathcal{R}^* \subset \mathcal{R}$ may be obtained with a positive probability, as a result of the application of the SCR rule in an arbitrary pheromone structure F and H. To these, rather static, mechanisms NCR and SCR we are adding a dynamic one that is related with the actualisation of the pheromone intensity on the trails, governed by the rule PUR. In this way we are introducing a dynamic exchange of information.

The PUR mechanism has two components: short range and long range, and is deterministic. They act on elements of the sets $\mathcal{F}$ and $\mathcal{H}$. We are adding (increasing) some amount of pheromone, by the application of two operators

$$\text{inc}_{\mathcal{F}} : \mathcal{F} \times \mathcal{R} \to \mathcal{F}, \& \ \text{inc}_{\mathcal{H}} : \mathcal{H} \times \mathcal{R} \to \mathcal{H}, \tag{8}$$

and we are decreasing the amount of pheromone by the application of two next operators

$$\text{dec}_{\mathcal{H}} : \mathcal{H} \times \mathcal{R} \to \mathcal{H}, \& \ \text{dec}_{\mathcal{H}} : \mathcal{H} \times \mathcal{R} \to \mathcal{H}. \tag{9}$$

If $\omega = x_{l_1}, x_{l_2}, ..., x_{l_r}$ is a word-solution, then the action of $\mathrm{inc}_{\mathcal{F}}$ on the vector F is

$$F[l_1] \leftarrow min(\tau_{max}, F[l_1]+1)$$

and the action of $\mathrm{dec}_{\mathcal{F}}$ is

$$F[l_1] \leftarrow max(1, F[l_1]-1),$$

on the vector element with the index l_1, being the index of the first element of the solution ω.

The application of two other operators $\mathrm{dec}_{\mathcal{H}}$ and $\mathrm{dec}_{\mathcal{H}}$ is governed by two formula:

$$H[l_i, l_{i+1}] \leftarrow min(\tau_{max}, H[l_i, l_{i+1}]+1), \&$$
$$H[l_i, l_{i+1}] \leftarrow (max(1, H[l_i, l_{i+1}-1]-1)$$

respectively, for all elements $H[l_i, l_{i+1}]$ of the matrix H, where $i = 1, 2, ..., r$ is the subsequent index of the symbols of the solution ω.

F. Theoretic model of DAS

Now we describe the theoretical model of DAS for the single ant, because of its simplicity. The extension for a set of ants is rather of technical nature, because of the sequential character of the algorithm in a loop.

By a state of an ant at time t we understand a quadruple

$$(F_{(t)}, H_{(t)}, \omega_{(t)}, \omega^*_{(t)})$$

where all objects, but the last one, have been already introduced and the index (t) means that it is their actual values at time t. The last object is the historical value of the best word-solution obtained till the moment t.

From the previous considerations we may form the following obvious proposition.

Proposition 1.

A. A state of the ant at time $t+1$ depends on the state of the ant at time t, only.

B. The set of all possible states of the ant is finite.

Now we may introduce an algebraic model of an DAS algorithm evolution for a single ant, where:

- $\widehat{U}_{(t)} \in [0,1]^{\mathfrak{s}}$ – is a column stochastic vector of size $\mathfrak{s}$ such, that $\widehat{U}_{(t)}[i]$ determines a value of probability of a chance that an ant state in moment t is s_i,
- $\widehat{T} \in [0,1]^{\mathfrak{s}\times\mathfrak{s}}$– is a column stochastic matrix of size $\mathfrak{s}\times\mathfrak{s}$ such, that $\widehat{T}[i,j]$ determines a value of probability of a chance that an ant changes its state from s_i to s_j.

Therefore

$$\widehat{U}_{(t)}\widehat{T} = \widehat{U}_{(t+1)} \tag{10}$$

gives a probabilistic evolution of an ant state between moments t and $t+1$. In general if $\widehat{U}_{(0)}$ is an initial distribution of probability of an ant start state, then

$$\widehat{U}_{(i)} = \widehat{U}_{(0)}\widehat{T}^i \tag{11}$$

describes ant state at moments $i = 1, 2, 3, \ldots$. Hence we formulate the next proposition.

Proposition 2.

Evolution process of a single ant in our theoretical model of DAS is a Markovian chain.

The method how to fill the matrix $\widehat{T}$ is described in [6], [7]. The question of the convergence has been solved positively as well, by the application of the convergence theorem formulated for the case of evolutionary algorithms in [8]. In fact we may formulate the result on the so-called pointwise convergence of the DAS, by enlarging the space Σ^* and adding a super-state composed of all states in which the last elements are undistinguishable by the solution quality function.

G. Self-adaptation of α parameter

For the purpose of numerical experiments presented in the article, we introduced a new mechanism for self-adaptation of α parameter occurring in the equation determining the probability of obtaining by a single ant a solution $\omega \in \mathcal{R}$ (see the equation at the beginning of Section II-E and Eq.1). The mechanism is based on the notion of radius variation of α parameter which is further denoted by γ. For practical reasons (arithmetic capabilities of computer's CPU) we aim to reduce γ possible values by $0 \le \gamma < max$, where *max* is some fixed constant. Then, value of α parameter at time t is in range from 1.0 to $1.0+\gamma$ (strengthening the pheromone trace) or from $\frac{1.0}{1.0+\gamma}$ to 1.0 (reducing the pheromone trace).

Initially (time $t = 0$) condition $\gamma = 0.0$ is satisfied, thus α parameter is equal to 1.0 (neutral state). In each subsequent iteration $t > 0$ we increment γ radius by $\frac{1}{N}$, where N is the size of the ant nest. Next α^* is value of α coefficient which was previously used while currently best result ω^* was found. Further α self-adaptation mechanism is in accordance with the following rules:

- if $|\alpha - \alpha^*| \le \frac{1}{N}$ holds, then α value is reset randomly with uniform probability in the range from 1.0 to $1.0+\gamma$ or form $\frac{1.0}{1.0+\gamma}$ to 1.0,
- if $|\alpha - \alpha^*| > \frac{1}{N}$ holds, then α value is reset to $\frac{\alpha+\alpha^*}{2}$ (bisection scheme).

Finally, if solution ω found at time t is better than solution so far the best ω^*, γ is again set at 0.0.

The above mechanism of self-adaptation of α parameter can be limited in the space of discrete values. Thus, all of the properties of Discrete Ant Algorithm introduced before, including most significant pointwise convergence, are sustained.

III. Application to Entity Summarisation

In this section, we demonstrate the application of the described optimisation technique on an interesting NP-hard optimisation problem concerning entity summarisation on semantic knowledge graphs that was recently proposed in [2].

A. Semantic Knowledge Graphs

A semantic knowledge graph (henceforth denoted as KG) is a quite novel format for represented semantic data. Knowledge graphs can be automatically or semi-automatically constructed in a process of "knowledge harvesting" from large corpora of text documents e.g. from the WWW, with use of advanced open-domain information extraction technology. There are

existing large datasets in such formats, e.g. DBpedia [9] or YAGO [10]

Basicly, KG constists of: fact graph and ontology.

The fact graph is a directed multi-graph where

- each node represents some entity (e.g. musician, actor, politician etc.) from some domain (e.g. art, movies, politics, etc.), e.g. "Fryderyk Chopin"
- each directed arc represents some "fact" concerning the entities being its ends, e.g. "Fryderyk Chopin is composer". Such facts are commonly represented as so called RDF-triples, that consist of subject, predicate and object, that is often denoted in the relational form: `predicate(subject, object)` (e.g. `is(Fryderyk Chopin, composer)`.

The ontology represents type hierarchy of the entities, i.e. each entity is connected to type node(s) in the ontology by special arcs like "type", etc.

B. Entity Summarisation

[2] studies a problem of entity summarisation in KG, i.e. given an entity (to be summarised) x, knowledge graph G and (small) $k \in N^+$ (a limit on number of facts in the summary) to select a set S of facts concerning the entity x that make a concise "summary" of x. In graph terms the problem consists in selecting a small set of "representative" arcs incident with x in the graph.

Since a node in a large KG (like DBpedia or YAGO) can easily have degree of 100 or higher, and the typical value of k is around 10, it naturally leads to a hard optimisation problem of how to select "the best" facts to the summary.

C. Novel Diversity-aware quality measure of entity summary

A good summary should select the most *important* facts concerning the entity. Furthermore, as observed and experimentally confirmed in [2], a desired summary would be *diversified* i.e. contain facts that concern various *aspects* of the entity being summarised.

This would be stated as a combinatorial optimisation problem via a properly defined bi-criteria objective function obj that takes into account 1) importance of facts, 2) mutual "dissimilarity" between facts.

More precisely, the problem can be defined as: out of the (given) set D of all facts concerning the entity, select a subset S of (up to) k facts so that the following measure is maximised:

$$obj(S) = (k-1)\sum_{d\in S} imp(d) + 2\delta \sum_{d_1 \neq d_2 \in S} diss(d_1, d_2)$$

where $imp : D \to Q^+$ is a "importance" weight of a fact and $diss : D^2 \to Q^+$ is represents pairwise "dissimilarity" of two facts and δ is a parameter to balance between the two criteria of "importance" and "dissimilarity" (to be tuned experimentally). We assume that imp is an increasing function of importance and $diss$ is an increasing function of dissimilarity of facts.

E.g. given entity "Albert Einstein" and two facts concerning this entity: d_1= `hasWonPrize(Albert Einstein, Nobel Prize)` and d_2 = `hasWonPrize(Albert Einstein, Mateucci Medal)` it seems reasonable that d_1 is more important than d_2. In addition, given a fact d_3=`hasChild(Albert Einstein, Evelyn Einstein)` it seems reasonable that $diss(d_1, d_2)$ is lower than $diss(d_1, d_3)$, etc.[1].

The imp and $diss$ functions can be computed based on structural and statistical properties of the underlying knowledge graph. In this paper we assume that the values of the functions are computed externally and we do not focus on this issue.[2]

The factors of $(k-1)$ and 2 are used in the formula to reflect the fact that there are $k(k-1)/2$ possible pairs of a k-element set.

The problem of optimising the obj function defined above is NP-hard (i.e. it can be reduced from MaxSumDispersion NP-hard optimisation problem). If the $diss$ function was a metric, one would adapt the existing 2-approximation algorithm for the MaxSumDispersion problem to this problem.

In the next section, we experimentally apply the ACO optimisation method described in Section II to solve hard optimisation problem defined here. This is additionally justified by the fact that $diss$ function is not necessarily a metric what excludes the applicability of some known approximation algorithms for the MaxSumDispersion problem.

TABLE I
ENTITIES SELECTED FOR EVALUATION.

Entity	Number of facts	Number of predicates
Albert Einstein	32	10
John Wayne	135	12
Denzel Washington	34	7
Robert Mitchum	61	5

IV. EXPERIMENTS

To study practical properties of the described approach we used YAGO2 semantic knowledge database [10] as the underlying knowledge graph. We run the experiments on a sample of entities representing some known people. Some of them, that represent actors, were earlier used for evaluation in [2] and some (e.g. "Albert Einstein") are new. Table I presents selected entities with number of facts assigned. Also number of unique predicates per entity is shown in third column. One can see that the number of facts varies much between entities but for each of them is much higher than 12. This implies that applying a brute force approach of considering all possible 12-element subsets of incident arcs (facts) would be of prohibitive time complexity.

For each fact of an entity we calculated its importance weight and for each pair of facts we calculated dissimilarity. Weights and dissimilarities were later used in $obj(S)$.

A weight of a fact is given by normalised weight of an object (ending of an edge). We applied undirected random

[1]All the mentioned examples are real and taken from the YAGO dataset

[2]The detailed description of how to compute imp and $diss$ functions and other concepts concerning diversified entity summarisation measure mentioned above are out of scope of this publication and deserves a separate publiction that is currently under preparation.

walk (with probability of getting back to the entity set to 0.15) over graph of facts. From this we obtained global weights of entities. In the next step we took facts (with associated endings = objects) associated to a studied entity and scaled weights in a way to fit in range $[0,1]$. At the end, the most important fact assigned to the entity has weight 1.0 and the least important $\varepsilon(\approx 0)$.

A dissimilarity of two facts associated with an entity was calculated using information about types of the entity. Using information about neighbourhood in hierarchy of types we measure how often both facts occur in similar entities. By similar we understand these of close types. Applied method returns dissimilarities always in range $[0,1]$.

A. Preliminary Results

Figure 1 presents sample results ($k=7$) of our method for $\delta = 0.0$ (diversity oblivious variant) and $\delta = 10.0$ (diversity aware variant) in comparison to the results of diversity aware algorithm from [2]. Although we show full results only for single entity ("Denzel Washington") outcomes for other entities have similar properties. Facts shown on different images were placed to resemble their similarity. One should note that since the time of writing [2] the YAGO2 database has been substantially updated and that could slightly influence final results. However we can see that in Diversum image and in image for $\delta = 10.0$ some facts are equal e.g. `directed(D.Wash., The Great Debaters)`, `influences(D.Wash., Noah Sife)`. Correspondence between other facts can also be found e.g. a fact `actedin(D.Wash., DJ Vu)` was replaced with fact `actedin(D.Wash., The preacher's wife)`, information about birth date and children were replaced with place of birth, wife and gender. Similar correspondence can be found between facts `hasWonPrize(D.Wash., Academy Award for Best Actor)` and `hasWonPrize(D.Wash., Tony Award)`.

Additional remarks can be stated after comparison of images for $\delta = 0.0$ and $\delta = 10.0$. One can see that for $\delta = 0.0$ facts connected to acting career dominate summarisation. Information about influence, wife and birth place was replaced with further information about movies that Denzel Washington acted in.

To measure overall quality of obtained summarisations we used Wikipedia info-boxes. Our method is based on this used in [2]. Let $S(e)$ (where $|S(e)| = k$) denote set of facts in summarisation of an entity e and $W(e)$ denote set of facts that can be found in info-box on corresponding Wikipedia pages. Let $f_1 \rhd f_2$ mean that f_1 is equal or more specific fact than f_2 and f_2 can be inferred from f_1. For example: the fact that someone is an actor can be inferred from the fact that he played in some specific movie. *Recall'* is then defined as:

$$Recall'(e) = \frac{|\{f : f \in W(e) \wedge \exists_{f' \in S(e)} f' \rhd f\}|}{|W(e)|}$$

Tables II and III present comparison of our method to results of Diversum and Precis [2]. *Recall'* measure for $obj(S)$ with $\delta = 0/1/10$ and for Diversum and Precis is shown. Table II contains results for $k = 7$ and Table III for $k = 12$. One can

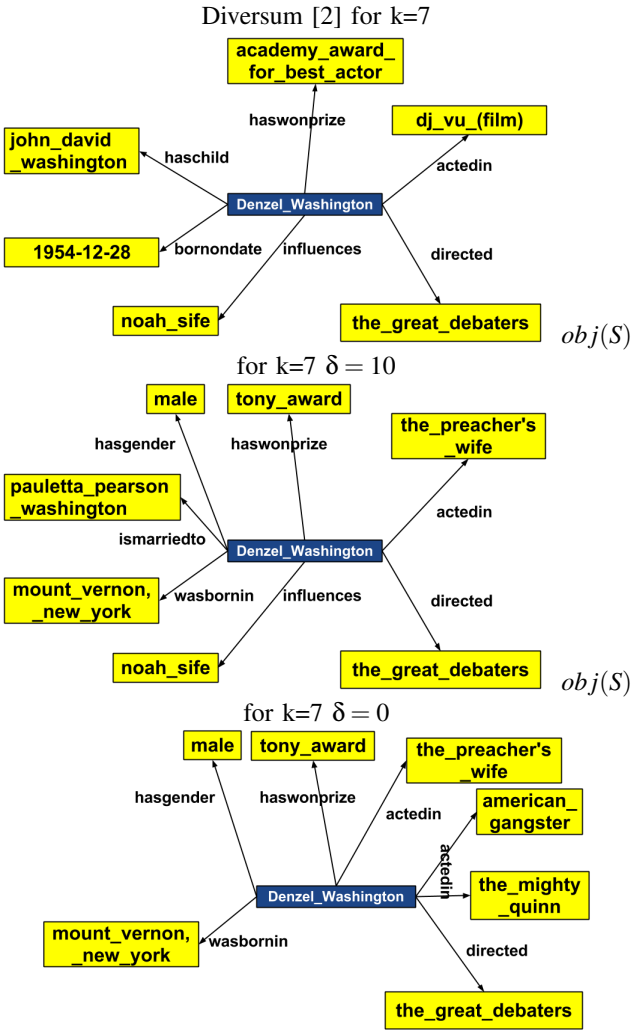

Fig. 1. Comparison of results for "Denzel Washington".

TABLE II
COMPARISON OF *Recall'* FOR $k = 7$.

Entity	δ=0	δ=1	δ=10	Div.	Prec.
J. Wayne	0.31	0.38	0.31	0.29	0.06
D. Wash.	0.50	0.50	0.67	0.60	0.20
R. Mitch.	0.30	0.50	0.50	0.50	0.25
mean:	0.37	0.46	0.49	0.46	0.17

see that our approach performs much better than Precis which seems to be the weakest one. The second worst approach is $obj(S)$ with $\delta = 0$ what shows how important diversity is. Diversum algorithm is much better and its results are comparable to our method with $\delta = 1$. When δ is set to 10 and diversity is high quality is the best according to the measure. Overall results are promising but need to be verified in further experiments on more entities.

Authors of *Recall'* assumed that info-boxes have size of about 7-12. This is not necessary true in our case e.g. for "Albert Einstein" there is 54 facts. Therefore we suggest

TABLE III
COMPARISON OF *Recall'* FOR $k = 12$.

Entity	δ=0	δ=1	δ=10	Div.	Prec.
J. Wayne	0.46	0.54	0.54	0.42	0.59
D. Wash.	0.50	0.67	0.67	0.60	0.20
R. Mitch.	0.30	0.50	0.50	0.50	0.25
mean:	0.42	0.57	0.57	0.51	0.35

TABLE IV
DIFFERENT MEASURES FOR "D. WASHINGTON" $k = 7$.

δ	*Recall'*	*Recall*	*Precision*	F_1
0	0.50	0.43	0.71	0.54
1	0.50	0.43	0.57	0.49
10	0.67	0.57	0.57	0.57

modified measures:

$$Recall(e) = \frac{|\{f : f \in W(e) \wedge \exists_{f' \in S(e)} f' \rhd f\}|}{|S(e)|}$$

$$Precision(e) = \frac{|\{f' : f' \in S(e) \wedge \exists_{f \in W(e)} f' \rhd f\}|}{|S(e)|}$$

Precision says how many facts out of returned k infer facts from info-box. *Recall* measures how many facts in info-box can be derived from summarisation or in different words: what fraction of facts we covered out of k possible to cover. As long as single fact from summarisation can infer one or zero facts in info-box, what is a specific property of considered data, *Recall* is limited to 1.0.

One should note that Wikipedia info-boxes contain some types of facts that are not included in YAGO2. Therefore we decided to update sets $S(e)$ in a way to keep only this facts that can be connected to facts in knowledge base according to relation $\rhd$.

Sample comparison of different measures for entity "Denzel Washington" can be found in Table IV. For this entity there is only 6 facts from Wikipedia info-box included into $S(e)$ therefore $Recall < Recall'$. Recall that parameter δ controls diversity of results. For $\delta = 0.0$ only weights are taken into account and dissimilarities are omitted. For $\delta = 1.0$ we expect results balanced in weight and diversity and for $\delta = 10.0$ we expect highly diversified results. One can note that *Precision* is very high for $\delta = 0$. The reason is that from every fact "acted in movie" we can infer that "D. Washington" is an actor. From practical point of view it would be enough to include single fact of such type. This situation points out vulnerability of this measure.

Tables V and VI presents average values of four entities from Table I of measures for $k = 7$ and $k = 12$. An analysis shows that both *Recall* and *Recall'* increase when δ increases. It means that for higher values of δ more facts from info-boxes is covered. *Precision* is the highest for $\delta = 0$. The reason of this behaviour was explained earlier (in context of "Denzel Washington" entity). The highest values of F_1 were obtained for $\delta = 10$ and at the end this value seem to be the best option.

TABLE V
AVERAGE (FOUR ENTITIES) MEASURES FOR $k = 7$.

δ	*Recall'*	*Recall*	*Precision*	F_1
0	0.30	0.50	0.75	0.59
1	0.36	0.61	0.68	0.64
10	0.39	0.61	0.68	0.64

TABLE VI
AVERAGE (FOUR ENTITIES) MEASURES FOR $k = 12$.

δ	*Recall'*	*Recall*	*Precision*	F_1
0	0.34	0.36	0.77	0.46
1	0.46	0.46	0.73	0.55
10	0.46	0.46	0.73	0.55

To study deeper the influence of a parameter δ in $obj(S)$ we performed three experiments for different values (0.0, 1.0, 10.0) of this parameter. Tables VII and VIII compare selected facts for different values of parameter δ. Each column says what fraction of facts is common for results with two different values of δ e.g. second column ("1 vs. 0") says how many facts is retained when $\delta = 0.0$ is changed to $\delta = 1.0$. The Table VII presents results for $k = 7$ and Table VIII for $k = 12$. Additionally Figure 2 visually compares second columns from tables.

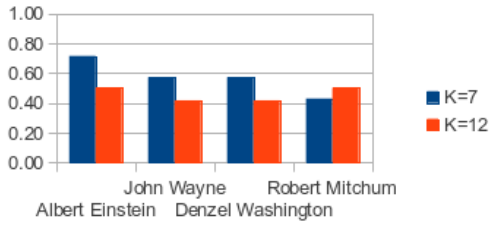

Fig. 2. Fraction of facts shared for $\delta = 0.0$ and $\delta = 10.0$.

TABLE VII
COMMON FACTS FOR DIFFERENT VALUES OF δ (K=7).

Entity / δ	1 vs. 0	10 vs. 0	10 vs. 1
A. Einstein	0.71	0.71	1.00
J. Wayne	0.57	0.57	0.57
D. Wash.	0.57	0.57	0.71
R. Mitchum	0.43	0.43	0.57
mean:	0.57	0.57	0.71
std:	0.12	0.12	0.20

TABLE VIII
COMMON FACTS FOR DIFFERENT VALUES OF δ (K=12).

Entity / δ	1 vs. 0	10 vs. 0	10 vs. 1
A. Einstein	0.50	0.50	0.92
J. Wayne	0.50	0.42	0.75
D. Wash.	0.50	0.42	0.92
R. Mitchum	0.50	0.50	1.00
mean:	0.50	0.46	0.90
std:	0.00	0.05	0.10

An analysis of Tables VII and VIII and Figure 2 leads to several remarks. At first, when δ is changed from 0.0 to 1.0 or 10.0 about half of facts is retained. It holds both for $k = 7$ and $k = 12$. However facts selected for $\delta = 1.0$ and $\delta = 10.0$ differs much e.g. about 30% facts differs for $k = 7$ and about 10% for $k = 12$. The difference in behaviour can be explained by small differences between top facts.

What is more, one can see that there are entities that are resistant to changes of δ. Figure 3 presents comparison of facts shared between situations $\delta = 1.0$ vs. $\delta = 10.0$ for $k = 7$ and $k = 12$. One can see that for "Albert Einstein" all facts are retained. By checking Table VII one can also see that for this entity adding information on dissimilarity ($\delta \neq 0$) changed results much less than for the others.

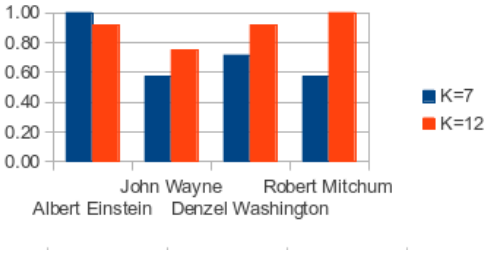

Fig. 3. Fraction of facts shared for $\delta = 1.0$ and $\delta = 10.0$.

V. CONCLUSIONS

Results obtained at current level of advance of our project are very promising. In all cases our results were not worse than these described in [2]. Most importantly, the preliminary results indicate appropriate tuning of the δ parameter in the newly proposed solution quality measure makes it possible to beat the performance of the previously proposed diversity-aware algorithm (DIVERSUM) presented in [2]. Anyways, further experiments and evaluations need to be done to confirm this, including user-based evaluations.

Also the issue of tuning the δ parameter should be separately studied as well as theoretical properties of the proposed measure.

In future work it would be also interesting to experimentaly compare the performance of the presented optimisation technique, especially the influence of the novel self-adaptation mechanism, with some other existing sub-optimal approaches, e.g. approximation algorithms for the Max Sum Dispersion problem and other NP-hard optimisation problems.

ACKNOWLEDGEMENTS

The work was supported by the Polish National Science Centre (NCN) grant N N516 481940 (DIVERSUM project) and partially by grant N N519 5788038 and by research fellowship within "Information technologies: research and their interdisciplinary applications" agreement POKL.04.01.01-00-051/10-00.

REFERENCES

[1] N. Eldredge and S. J. Gould, "Punctuated equilibria: An alternative to phyletic gradualism," in *Models in Paleobiology*, T. J. M. Schopf, Ed. San Francisco: Freeman Cooper, 1972, pp. 82–115.

[2] M. Sydow, M. Pikuła, and R. Schenkel, "The notion of diversity in graphical entity summarisation on semantic knowledge graphs," *Journal of Intelligent Information Systems*, pp. 1–41, 2013. [Online]. Available: http://dx.doi.org/10.1007/s10844-013-0239-6

[3] M. Dorigo, "Optimization, learning and natural algorithms (in Italian)," Ph.D. dissertation, Dipartimento di Elettronica, Politecnico di Milano, Milan, Italy, 1992.

[4] M. Dorigo, M. Birattari, and T. Stutzle, "Ant colony optimization," *Computational Intelligence Magazine, IEEE*, vol. 1, no. 4, pp. 28–39, 2006.

[5] M. Dorigo and L. Gambardella, "Ant colony system: a cooperative learning approach to the traveling salesman problem," *Evolutionary Computation, IEEE Transactions on*, vol. 1, no. 1, pp. 53–66, 1997.

[6] P. Rembelski, "Theoretical model of sas ant colony optimisation algorithm (*In Polish*)," pp. 37–42, 2010, proc. of XII International PhD Workshop OWD 2010.

[7] W. Kosiński and S. Kotowski, "Limit properties of evolutionary algorithms," *In-Tech*, pp. 1–28, 2008.

[8] J. Socała, W. Kosiński, and S. Kotowski, "On asymptotic behaviour of a simple genetic algorithm (*In Polish*)," *Matematyka Stosowana*, vol. 6/47, pp. 70–86, 2005.

[9] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, "Dbpedia: A nucleus for a web of open data," *The Semantic Web*, vol. 4825, pp. 722–735, 2007. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-76298-0-52

[10] 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*, ser. WWW '07. New York, NY, USA: ACM, 2007, pp. 697–706. [Online]. Available: http://doi.acm.org/10.1145/1242572.1242667