Calculating distance measure for clustering in multi-relational settings

Olegas Niakšu
Vilnius University, Institute of Mathematics and Informatics
Akademijos str. 4, LT-08663, Vilnius, Lithuania
E-mail: Olegas.Niaksu@mii.vu.lt

ABSTRACT

The paper deals with a distance based multi-relational clustering application in a real data case study. A novel method for a dissimilarity matrix calculation in multi-relational settings has been proposed and implemented in R language. The proposed method has been tested by analyzing publications related to data mining subject and indexed in the medical index database MedLine. Clustering based on partitioning around medoids was used for the semi-automated identification of the most popular topics among the MedLine publications. The algorithm implements greedy approach and is suitable for small data sets with a limited number of 1:n relational joins.

1 INTRODUCTION

Clustering has been studied for decades in disciplines such as statistics and data mining (DM). Clustering can be defined as a DM task, where objects are being *unsupervisedly* subdivided into groups, in such a way, that objects of each group are more similar to each other than in comparison to the objects in other groups. Logically, the objects similarity measure is of key importance. The main contribution of our research is a novel customized distance measure calculation method, which reflects relational features of the input data. The method was applied for a dissimilarity matrix calculation, which was later used with partitioning clustering approaches.

Typically, existing clustering algorithms are representatives of one of the following clustering method groups: hierarchical methods, partitioning methods (e.g. k-means, pam), density-based methods (e.g. DBSCAN), model-based methods, subspace clustering, fuzzy clustering, etc.

However, the majority of these clustering methods have been created to process data in “a single table” format. Therefore, typically clustering algorithms underperform in multi-relational data.

We have applied distance based clustering with a novel compound distance measure, based on Gower and Ochiai metrics, which was created specifically for the exploratory research of publications related with DM topic from MedLine database [7]. However, the algorithm can be reused for similar multi-label text classification tasks.

Following the study [5], this research also contributes to the topic of defining DM footprint in healthcare domain, its spread, usability and characteristic features.

The remaining of the paper is organized as follows. Section 2 briefly summarizes the approaches for the clustering in multi-relational settings. Section 3 introduces a novel similarity measure calculation approach. Experimental investigation is described in Section 4 and conclusions are presented in Section 5.

1.1 Background

In our experiment, PubMed database was used, as the biggest medical database, having explicit hierarchical semantic tagging system, called MeSH [6].

PubMed is comprised of more than 21 million citations for biomedical literature from MEDLINE, life science journals, and online books. The Medical Subject Headings (MeSH) is a controlled vocabulary, which is used for indexing, cataloging, and searching for biomedical and health-related information and documents.

Each publication in our case-study has been mapped to MeSH Concepts, Descriptors and Semantic Types.

The whole search result data set with available attributes has been exported to XML format, and then transferred to a relational database.

Having MeSH vocabulary and the exported publications dataset in one database schema, allowed us to leverage semantic concept aggregation underlying in MeSH and to group articles on a higher abstraction layer using distance measure described in Section 3.

2 MULTI-RELATIONAL PARTITIONING CLUSTERING FOR 2:N ONE-TO-MANY RELATIONAL ENTITIES

According to Van Laer and De Raedt [9], when upgrading propositional algorithm to the first-order learners type, it is important to retain as much of the original algorithm as possible, and only the key notion should be updated. In case of distance-based approaches, the distance measure or its direct derivative similarity measure is the key notion of choice.

As it was proposed by T. Horwath and S. Wrobel [2], instead of forming an explicit hypothesis in the form of first-order clauses, we can store all available objects, comprising aggregated distance measures. As a next step, we compare each object, with neighboring objects.

In our case study, relational data representation includes *one-to-many* relational joins between the entities *Keyword* and *MeSH Concept*, *Keyword* and *MeSH Descriptor*, and between *MeSH Semantic Type* and *MeSH Concept*.

MeSH definitions of these entities are as follows. *Descriptor* is used to index citations in MEDLINE database, and for cataloging of publications. Most *Descriptors* indicate the subject of an indexed item, such as a journal article. MeSH *Descriptors* are organized in 16 categories, each of them is further divided into subcategories, where descriptors are arrayed hierarchically in twelve hierarchical levels.

A *Descriptor* is broader than a *Concept* and consists of a class of concepts. *Concepts*, in turn, correspond to a class of *Terms* which are synonymous with each other. Thus MeSH has a three-level structure: *Descriptor → Concept → Term*. Every *Term* is assigned to one or more *Semantic Types*, which assign the broadest ontological meaning to a T*erm*. There are only 132 different *Semantic Types* in MESH. In our experiment, we have de-normalized entity-relationship structure in a way that entities *Term* and *Concept* have been merged into entity *Concept*.

Summarizing, MeSH controlled vocabulary allowed us to extract additional semantic information from the keywords assigned to the articles.

Formally, in our study, first-order instances of Articles A are represented by the predicate *Article* A, and the following ground atoms: Concept - C, Descriptor - D, and Semantic type - S. Let us assume that our case study's dataset's instance example I:

I = A (art1),

with defined background knowledge BK:

C(art1, “Benpen”),
D(art1, “Penicillin G”),
S(“Benpen”, “Antibiotic”).

The vocabulary of this example consist of the predicate A and the background predicates concept C, the descriptor D and the semantic type S, with the following argument types: A(a1: name), C(a1: name, a2: discrete), D(a1: name, a2: discrete), S(a1: name, a2: discrete). The structure of ground atoms repeats a subset of relational data structure. More precisely, the entities *Concept* and *Descriptor* are joined to the entity *Article* through the entity *Keyword*. But, since *Keyword* is in one-to-one relation with *Article,* it was substituted by it.

3 THE SIMILARITY MEASURE IN MULTI-RELATIONAL SETTINGS

In this section we will describe an approach how to combine different similarity measures in a way, suitable to multi-relational structures, in particular considering our use case example.

Very often, in complex data structures, there can be no objectively “best” distance or similarity measure, or at least formal proof would be too expensive. Therefore, there are certain trade-offs when selecting optimum similarity measure. Since the data in our case study does not form Euclidean space, we require more robust distance measure. Gower's general coefficient of similarity [1] is one of the most popular measures of proximity for mixed data types.

Using Gower's general similarity coefficient we can compare values of predicate arguments. Gower’s coefficient of similarity s_i is defined as follows:

$$s_{i,j} = \frac{\sum_k w_k s_{ijk}}{\sum_k w_k} \qquad (1),$$

where: s_{ijk} denotes the contribution provided by the k_{th} variable dependable on its data type, and w_k is assigned weight function. In other words, the similarity measure of the two objects i & j, is a sum of normalized weighted similarities of each object’s variable k (attribute of the entity).

The calculation of s_{ijk} depends on the data type as described below. For nominal variables:

$$s_{ijk} = 1, \text{ iff } x_{ik} = x_{jk}\text{ , and } s_{ijk} = 0, \text{ when } x_{ik} \neq x_{jk} \qquad (2)$$

For numeric variables:

$$s_{ijk} = 1 - \left|x_{ik} - x_{jk}\right| / r_k, \qquad (3),$$

where r_k is a difference between max and min values of k’th variable. As in the case with nominal variables, s_{ijk} equals to 1 when $x_{ik} = x_{jk}$. And s_{ijk} equals to 0, when x_{ik} and x_{jk} represent maximum and minimum values of the variable.

Binary data type in Gower metric can be treated as a nominal data type, where, $s_{ijk} = 1$, iff the compared values equals to 1. Additionally, it shall be stated, that for the cases where all variables are of binary type, another similarity measures might be more preferable, like Jaccard similarity coefficient.

Furthermore, to compare two value lists in the case of comparing objects with one-to-many relations, we propose to use Ochiai (Ochiai-Barkman) coefficient [7]:

$$s_{l1,l2} = \frac{n(l_1 \cap l_2)}{\sqrt{n(l_1) \times n(l_2)}} \qquad (4),$$

where l_1, l_2 – nominal value lists, n(l) – the number of elements in l.

In a relational data structure, the compared objects are represented by a number or relations and relational joins. For each attribute of a relation, denote it as a variable *k*, which is considered to be a part of the selected search space, atomic similarities s_{ijk} have to be calculated using Gower similarity for a specific data type, value lists using Ochiai coefficient extended by Gower similarities for numeric and binary data types. Finally, the overall similarity measure between two objects is calculated as a weighted sum of s_{ijk} according to (1).

A relational data model has always to be treated with care, and certain preprocessing, de-normalization has to be applied. Considering the whole available relational data might be impractical. Hence, only valuable entities and attributes have to be selected. There are different recommendations on the relational feature selection, e.g. as described in works of R.T. Ng and J. Han [4].

The selected entities of the data model shall be analyzed for de-normalization possibility, assuming their relational join type. Entities with one-to-one type joins typically can be easily merged. For the entities connected with one-to-many joins, Ochiai with Gower coefficient for numeric, binary data types shall be used. *Many-to-many* related entities in many cases can be de-normalized to one one-to-many relationship.

In our case study a compound object *Article*, has value list variables (vectors) *Concepts* and *Descriptors.* And variable *Concept* is in fact is a part of a predicate pointing to the variable *Semantic type.*

Furthermore, applying the generic Gower similarity coefficient to the predicates C, D, and S, we have constructed the following compound similarity measures to compare two instances of article A:

$$sim_{A1,A2} = \frac{w_c simC + w_d simD + w_s simS}{\mathrm{w_c + w_d + w_s}},$$

where

$$\mathrm{simC} = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m} s_{ij}(concept_i(A_1), concept_j(A_2))}{\sqrt{m \times n}},$$

$$\mathrm{simD} = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m} s_{ij}(descriptor_i(A_1), descriptor_j(A_2))}{\sqrt{m \times n}},$$

$$\mathrm{distS} = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m} s_{ij}(semantictype_i(A_1), semantictype_j(A_2))}{\sqrt{m \times n}}.$$

Since the similarity measures $\mathrm{s_{ij}}$(concepts), $\mathrm{s_{ij}}$(descriptors), and $\mathrm{s_{ij}}$(semantictypes) measure the similarity among nominal values, only formulas (2) and (4) have been used in our case study.

In essence, *simC*, *simD*, and *simS* calculate similarity of the value lists (accordingly *Concepts, Descriptors*, and *Semantic Types*) which are relationally joined to the central entity *Article*. Other known approach for this task is described by Horwath, Wrobel et al. [2], where the authors proposed to calculate influence function, the cost of which equals to the effort of the lists equalization. However, our proposed, simple match calculation requires less computational effort and is reasonable for the lists with non-repeating values.

Another important aspect is the determination of weights for the overall similarity measure (1) calculation. Some authors, e.g. T. Horwath and S. Wrobel propose a simplified approach, by not using weights at all. This simplification in many cases may be unadjusted, because of the uneven nature of the data. In our experiment, two approaches have been used: the statistical one, where weights are proportional to the number of tuples of the relevant entities; and expert based, where weights have been experimentally adjusted and normalized by the domain expert.

In the first case weights have been calculated as follows:

$$w_c = \frac{n_c}{n_c+n_d+n_s}, w_d = \frac{n_d}{n_c+n_d+n_s}, w_s = \frac{n_s}{n_c+n_d+n_s} \quad (5)$$

The described weight distribution is reasonable in the cases, when we want to level the importance of the each list value variable according to the relative number of tuples in each entity.

In other examples, having more diverse set of variables, this statistical approach might be appended or changed by the domain expert knowledge and empirical experiments. If that is the case, for the calculation efficiency, it is important to store all s_{ijk}values, for further experiments with different $\mathrm{w_k}$ values. In opposite case, if only the resulting $\mathrm{s_{ij}}$ are preserved, when in order to change the weights, the whole similarity matrix shall be recalculated from a scratch.

According to our experiment results, the described similarity measure derives stable values, meaning that small changes on a term do not cause big changes in distance values. The experiments with real data have shown that in some cases it is even too stable and lack some responsiveness to the data changes. However, this is easily solvable by fine-tuning weight parameters $\mathrm{w_c}$, $\mathrm{w_d}$, $\mathrm{w_s}$. First of all, in order to automatically extend the distance measure to varying arities, we assigned initial weight values proportionally to the sizes of Concept, Descriptor and Semantic Type nominal value lists, as shown in (5). Later we underwent a series of trials with subjective $\mathrm{w_c}$, $\mathrm{w_d}$, $\mathrm{w_s}$ values, based on subjective domain expertise.

Finally, the dissimilarity value was calculated as follows:

$$dissim_{A1,A2} = 1 - \mathrm{sim}_{A1,A2} \quad (6)$$

The algorithm, calculating full dissimilarity matrix for the set of articles, has been implemented in R. R libraries “cluster” and “fpc” were used, for the different partitioning around medoids (PAM) implementations [3]. Due to a large search space, extended by joined relations, the algorithm requires a vast computational power. Multiple iterations of distances between each object and its selected related compound entities have resulted in the algorithm complexity of $O(n^2 L_c^2 L_d^2 L_s^2)$, where $\mathrm{L_c}$ – the length of list of *Concepts*, $\mathrm{L_d}$ – the length of list of *Descriptors*, $\mathrm{L_s}$ – the length of list of *Semantic types*.

Moreover, it is well scalable, and our further step will be parallelization of this algorithm.

The results of PAM clustering application with the described similarity measure for exploratory analysis of publications indexed by PubMed are presented in the next sections.

4 EXPERIMENTAL INVESTIGATION

The dissimilarity matrix calculation algorithm has been implemented in R language, and the resulted matrix of dissimilarity measures has been used with PAM clustering, implemented in R. Totally 2.284.453 similarity values have been calculated. After a few iterations of code optimization, overall achieved performance of an average size data set for 100 similarity values was in the range of 40-60 seconds on one core of Intel i7 CPU. The parallelization gave a huge effect, since each distance measure is independent and thus can be calculated in parallel. However, data exchange between nodes required by the parallelization had a negative impact and had reduced the positive effect of the parallelization. As a further research step, the algorithm recoding with its further parallelization in mind is planned.

For the evaluation of the overall clustering quality, cluster's *silhouette* value has been used. The *silhouette* value depicts the quality of each object's cluster. Cluster's *silhouette* value is derived in the following way. Let a(i) be the average dissimilarity between object i and all other objects of the cluster A, to which it belongs. For another cluster C_1, let d(i,C_1) equals to average dissimilarity of i to all objects of cluster C1. Then, let calculate d(i,C) for all the remaining clusters $C_{2..n}$ and assign the smallest of these d(i,C) to d_min(i). The *silhouette* value of an object i is defined as follows:

$$silh_i = \frac{d\_min(i) - a(i)}{\max\{a(i), d\_\min(i)\}} \qquad (7)$$

And the cluster's *silhouette* value is an average *silhouette* value of all its members. Values near 1 mean that the object i is assigned to a correct cluster. In contrast, values close to -1 mean that it is likely that an object is assigned to a wrong cluster. And the *silhouette* value around 0, means that the object i can be equally assigned to the selected or the nearest cluster.

In our case, trying different number of clusters, the maximum achieved *silhouette* values were in the range: 0.20 - 0.30. Objectively, that means the overall clustering result is unsatisfactory, and shows that the found clusters are poorly describing the data set.

However, considering a non-trivial task of scientific publications semantic grouping, the whole exercise was not fruitless, and gave us some interesting insights.

Regretfully, there is no point of reference or golden standard to compare our results with. Therefore, comparison to other possible clustering methods is planned for further research step.

The application of clustering with the described similarity measure on relational data of MedLine and MeSH has shown that there are no large and very popular topics, and the research within DM application in healthcare area is extremelly diverse.

Also our research results have revealed a couple of clusters with a higher research interest. Among them we can mention the following relatively more popular research areas: DM applications within protein structure analysis, specific patient profile search, text mining of medical text, public health legislation documents mining, commerce practices (fraud detection), disease diagnostics, survival prediction, natural language processing information retrieval, image data analysis.

5 CONCLUSION

A compound dissimilarity measure calculation algorithm for multi-relational data structures has been created, implemented and tested with a real world data clustering task. The proposed dissimilarity measure aggregates Gower similarity coefficient and Ochiai-Barkman coefficient and is applicable for different relational data models.

However, the presented approach has not been formally tested yet and requires further experiments and formal evaluation. Initial comparison tests have been made by using the same use case data converted to a propositional form and applying k-means, PAM, and CLARA clustering algorithms. Still it has resulted in another set of low quality clusters, with less interesting practical information.

Hence, the next planned research activity will include approbation with classified multi-relational data sets and comparison to another clustering methods.

The main known shortcoming of the implemented algorithm is its overall performance, due to the applied greedy approach. Though this approach was suitable for our case study, in other cases large data clustering algorithms CLARA or CLARANS [4] might be used instead of PAM.

Conclusions on DM research within healthcare domain

The practical case-study results presented in 4th section are not homogeneous and hence are not generalizable. Instead, we provide a few atomic conclusions of the analyzed case study clustering task:

- Oncology diseases are on the top of the mined disease list. Cardiovascular diseases are only on the third place after nervous system diseases.
- Interestingly, too little attention is currently paid to chronic diseases, which are believed to be the biggest challenge of modern healthcare systems because of the aging population.
- There is an outstanding number of articles in the field of genetics, which reconfirms that DM provides powerful arsenal of techniques for high volume data analysis.

References

[1] Gower, J. C. 1971. A general coefficient of similarity and some of its properties. *Biometrics*, 857-871.

[2] Horwath, T., Wrobel, S., Bohnebeck, U. 2001. Relational Instance-Based Learning with Lists and Terms. Kluwer Academic Publishers. Machine Learning, 43, 53–80

[3] Kaufman, L. and Rousseeuw, P.J. 1987. Clustering by means of Medoids, in Statistical Data Analysis Based on the L_1-Norm and Related Methods, edited by Y. Dodge, North-Holland, 405–416.

[4] Ng, R. T., Han, J. 2002. CLARANS: A Method for Clustering Objects for Spatial Data Mining. IEEE Transactions on Knowledge and Data Engineering, Vol. 14, No. 5.

[5] Niaksu, O., Kurasova, O. 2011. Data Mining Applications in Healthcare: Research vs Practice.

[6] National Library of Medicine – MeSH. http://www.nlm.nih.gov/mesh/meshhome.html

[7] Ochiail, A. 1957. Zoogeographic studies on the soleoid fishes found in Japan and its neighboring regions. Bull Jnp Soc Sci Fish 22:526-530.

[8] PubMed - database of references and abstracts on life sciences and biomedical topics. http://www.ncbi.nlm.nih.gov/pubmed/

[9] Van Laer V. and De Raedt L. 2001. How to Upgrade Propositional Learners to First Order Logic: A Case Study. Relational Data Mining. Springer-Verlag., p.235–261.