“BisoNet” Generation using Textual Data

Marc Segond and Christian Borgelt

European Center for Soft Computing
Calle Gonzalo Gutiérrez Quirós s/n, E-33600 Mieres (Asturias), Spain
{marc.segond,christian.borgelt}@softcomputing.es

Abstract. According to Koestler, the notion of a *bisociation* denotes a connection between pieces of information from habitually separated domains or categories. In this paper, we consider a methodology to find such bisociations using a network representation of knowledge, which is called a *BisoNet*, because it promises to contain bisociations. In a first step, we consider how to create BisoNets from several textual databases taken from different domains using simple text-mining techniques. To achieve this, we introduce a procedure to link nodes of a BisoNet and to endow such links with weights, which is based on a new measure for comparing text frequency vectors. In a second step, we try to rediscover known bisociations, which were originally found by a human domain expert, namely indirect relations between migraine and magnesium as they are hidden in medical research articles published before 1987. We observe that these bisociations are easily rediscovered by simply following the strongest links. Future work includes extending our methods to non-textual data, improving the similarity measure, and applying more sophisticated graph mining methods.

1 Introduction

The concept of association is at the heart of many of today’s powerful ICT technologies such as information retrieval and data mining. These technologies typically employ “association by similarity or co-occurrence” in order to discover new information that is relevant to the evidence already known to a user.

However, domains that are characterized by the need to develop innovative solutions require a form of creative information discovery from increasingly complex, heterogeneous and geographically distributed information sources. These domains, including design and engineering (drugs, materials, processes, devices), areas involving art (fashion and entertainment), and scientific discovery disciplines, require a different ICT paradigm that can help users to uncover, select, re-shuffle, and combine diverse contents to synthesize new features and properties leading to creative solutions. People working in these areas employ creative thinking to connect seemingly unrelated information, for example, by using metaphors or analogical reasoning. These modes of thinking allow the mixing of conceptual categories and contexts, which are normally separated. The functional basis for these modes is a mechanism called *bisociation* (see [1]).

According to Arthur Koestler, who coined this term, *bisociation* means to join unrelated, and often even conflicting, information in a new way. It means being "double minded" or able to think on more than one plane of thought simultaneously. Similarly, Frank Barron [2] says that the ability to tolerate chaos or seemingly opposite information is characteristic of creative individuals.

Several famous scientific discoveries are good examples of bisociations, for instance Isaac Newton's theory of gravitation and James C. Maxwell's theory of electromagnetic waves. Before Newton, a clear distinction was made between *sub-lunar* (below the moon) and *super-lunar physics* (above the moon), since it was commonly believed that these two spheres where governed by entirely different sets of physical laws. Newton's insight that the trajectories of planets and comets can be interpreted in the same way as the course of a falling body joined these habitually separated domains. Maxwell, by realizing that light is an electromagnetic wave, joined the domains of optics and electromagnetism, which, at his time, were also treated as unrelated areas of physical phenomena.

Although the concept of bisociation is frequently discussed in cognitive science, psychology and related areas (see, for example, [1–3]), there does not seem to exist a serious attempt at trying to formalize and computerize this concept. In terms of ICT implementations, much more widely researched areas include association rule learning (for instance, [4]), analogical reasoning (for example, [5, 6]), metaphoric reasoning (for example, [7]), and related areas such as case-based reasoning (for instance, [8]) and hybrid approaches (for example, [9]).

In order to fill this gap in current research efforts, the BISON project[1] was created. This project focuses on a knowledge representation approach with the help of networks of named entities, in which bisociations may be revealed by link discovery and graph mining methods, but also by computer-aided interactive navigation. In this paper we report first results obtained in this project.

The rest of this paper is structured as follows: in Section 2 we provide a definition of the core notion of a *bisociation*, which guides our considerations. Based on this definition, we justify why a network representation—a so-called *BisoNet*—is a proper basis for computer-aided bisociation discovery. Methods for generating BisoNets from heterogeneous data sources are discussed in Section 3, including procedures for selecting the named entities that form its nodes and principles for linking them based on the information extracted from the data sources. In particular, we present a new measure for the strength of a link between concepts that are derived from textual data. Such link weights are important in order to assess the strength of indirect connections like bisociations.

Afterwards, in Section 4 we report results on a benchmark data set (consisting of titles and abstracts of medical research articles), in which a human domain expert already discovered hidden bisociations. By showing that with our system we can create a plausible BisoNet from this data source, in which we can rediscover these bisociations, we provide evidence that the computer-aided search for bisociations is a highly promising technology.

Finally, in Section 5 we draw conclusions from our discussion.

[1] See `http://www.bisonet.eu/` for more information on this EU FP7 funded project.

2 Bisociation and BisoNets

Since the core notion of our efforts is *bisociation*, we start by trying to provide a sufficiently clear definition, which can guide us in our attempts to create a system able to support a user in finding bisociations. A first definition within the BISON project[2] characterizes *bisociation* as follows:

> A *bisociation* is a link L that connects two domains D_1 and D_2 that are unconnected given a specific context or view V by which the domains are defined. The link L is defined by a connection between two concepts c_1 and c_2 of the respective domains.

Although the focus on a connection between two habitually (that is, in the context a user is working in) separated domains is understandable, this definition seems somewhat too narrow. Linking two concepts from the same domain, which are unconnected within the domain, but become connected by employing indirect relations that pass through another domain, may just as well be seen as bisociations. The principle should rather be that the connection is not fully contained in one domain (which would merely be an association), but needs access to a separate domain. Taking this into account, we generalize the definition:

> A *bisociation* is a link L between two concepts c_1 and c_2, which are unconnected given a specific context or view V. The concepts c_1 and c_2 may be unconnected, because they reside in different domains D_1 and D_2 (which are seen as unrelated in the view V), or because they reside in the same domain D_1, in which they are unconnected, and their relation is revealed only through a *bridging concept* c_3 residing in some other domain D_2 (which is not considered in the view V).

In both of these characterizations we define domains formally as sets of concepts. Note that a *bridging concept* c_3 is usually also required if the two concepts c_1 and c_2 reside in different domains, since direct connections between them, even if they cross the border between two domains, can be expected to be known and thus will not be interesting or relevant for a user.

Starting from the above characterization of *bisociation*, a network representation, called a *BisoNet*, of the available knowledge suggests itself: each concept (or, more generally, any named entity) gives rise to a node. Concepts that are associated (according to the classical paradigm of similarity or co-occurrence) are connected by an edge. Bisociations are then indirect connections (technically paths) between concepts, which cross the border between two domains.

Note that this fits both forms of bisociations outlined above. If the concepts c_1 and c_2 reside in different domains, the boundary between these two domains necessarily has to be crossed. If they reside in the same domain, one first has to leave this domain and then come back in order to find a bisociation.

[2] See `http://www.inf.uni-konstanz.de/bisonwiki/index.php5`, which, however, is not publicly accessible at this time.

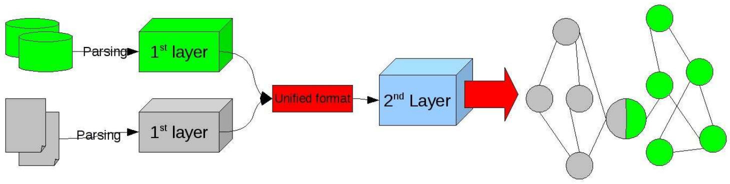

Fig. 1. Illustration of the structure of the BisoNet generator.

3 BisoNet Generation

A system for generating BisoNets requires three ingredients: (1) A component to access the original, usually heterogeneous data sources. In order to cope with different data formats, we suggest, in Section 3.1, a two-layer architecture. (2) A method for choosing the named entities that are to form the nodes of the BisoNet. Here we rely on standard keyword extraction techniques, as discussed in Section 3.2. (3) A procedure for linking the nodes of a BisoNet and for endowing them with weights that indicate the association strength. For this we suggest, in Section 3.3, a new association measure for keywords.

3.1 Data Access and Pre-Processing

As explained above, a BisoNet is a network that promises to contain bisociations. In order to generate such networks, we first have to consider two things: we must be able to read different and heterogeneous data sources, and we have to be able to merge the information derived from them in one BisoNet. Data sources can be databases (relational or of any other type), text collections, raw text, or any data that provide information about a domain. Due to the wide variety of formats a data source can have, the choice we made here is not to provide an interface of maximal flexibility that can be made to read any data source type, but to structure our creation framework into two separate layers.

The first layer directly accesses the data source and therefore has to be newly developed for or at least adapted to the specific format of the data source. The second layer is the actual BisoNet generation part. It takes its information from the first layer, always in the same format, and therefore can generate a BisoNet from any data source, as far as it is parsed and exported in the form provided by the first layer (see Figure 1 for a sketch).

The way data should be provided to the second layer is fairly simple, because in this paper we confine our considerations to textual data. As a consequence, the second layer creates nodes from data that are passed as records containing textual fields. These textual fields can contain, for now, either words or authors names. This procedure and data format is well adapted to textual databases or text collections, but is meant to evolve in future development in order to be able to take other types of data sources into account. However, since most of the data sources that we have used so far were textual data sources, this protocol seems simple and efficient. Future extensions could consist in including raw data fields (for example, to handle images), and will then require an adaptation of the second layer to be able to create nodes from other objects than textual data.

The second layer builds a BisoNet by extracting keywords using standard text mining techniques such as stop word removal and stemming (see [10]). The extracted keywords are weighted by their TFIDF (Text Frequency - Inverse Document Frequency) value (see [11]), thus allowing us to apply a (user-defined) threshold in order to filter the most important keywords, as will be detailed in Section 3.2. Links between nodes are created according to the presence of co-occurrences of the corresponding keywords in the same documents, and are weighted using a similarity measure adapted to the specific requirements of our case, which will be presented in Section 3.3. In the case that author lists are provided with each text string, extracted keywords are also linked to the related authors. These links are weighted according to the number of times a keyword occurs in a given author's work.

3.2 Creating nodes

In our BisoNets nodes represent concepts. As we only talk about textual databases, we made the choice to characterize concepts by keywords that are extracted from the textual records taken from the data sources. In the second layer of our framework, each textual record j is processed with stop word removal algorithm. Then the text frequency values are computed for each remaining term i as shown in Equation 1, where $n_{i,j}$ is the number of occurrences of the considered term in textual record j and $\sum_k n_{k,j}$ is the sum of number of occurrences of all terms in textual record j.

$$\mathrm{tf}_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \tag{1}$$

Naturally, this procedure of keyword extraction is limited in its power to capture the contents of the text fields. The reason is that we are ignoring synonyms (which should be handled by one node rather than two or more), hyper- and hyponyms, pronouns (which may refer to a relevant keyword and thus may have to be counted for the occurrence of this keyword) etc. However, such linguistic properties are very difficult to take into account and need sophisticated tools (like thesauri etc.). Since such advanced text mining is not the main goal of our work (which rather focuses on BisoNet creation), keeping the processing simple seemed a feasible option. Nevertheless, advanced implementations may require such advanced processing, because ignoring, for example, synonyms and pronouns can distorts the statistics underlying, for instance, the term frequency value: ignoring pronouns that refer to a keyword, or not merging two synonyms makes the TF lower than it should actually be.

After all records have been processed, the inverse document frequency of each keyword i is computed according to Equation 2, where $|D|$ is the total number of records in the database and $|\{d \in D \mid t_i \in d\}|$ is the number of records in which the term t_i appears.

$$\text{idf}_i = \log \frac{|D|}{|\{d \in D \mid t_i \in d\}|} \tag{2}$$

Each node is then weighted with its corresponding average TFIDF value (summing and normalizing all the TF values for a node and then multiplying by the IDF value). This TFIDF approach is a very well known approach in text mining that is easy to implement and makes one able to easily apply a threshold, thus selecting only the most important nodes (keywords). A node then contains, as an attribute, a list of all the TF values for each document of the BisoNet in which its associated keyword appears. This allows us to compute the similarity measure presented in Section 3.3 in order to create links.

According to the definition of a bisociation presented in Section 2, two concepts have to be linked by other concepts that are not in their proper domain (so-called *bridging concepts*). This leads us to introduce the notion of domains, into which the nodes are grouped, so that we can determine when borders between domains are crossed. In order to be able to classify nodes according to their membership in different domains, it is important that they keep, also as an attribute, the domains the data sources belong to, from which they have been extracted. Since the same keyword can occur in several data sources, taken from different domains, one has to be able (for example, for graph mining and link discovery purposes) to know whether a certain keyword has to be considered from a certain domain's point of view. The nodes therefore keep this information as vector of domains their associated keyword belongs to.

This can be interesting, for example, to mine or navigate the BisoNet, keeping in mind that a user may be looking for ideas related to a certain keyword belonging to a domain A. The results of a search for bisociations might also belong to domain A, because it is the domain of interest of the user. However, these results should be reached following paths using keywords from other domains, that is to say bisociations. This procedure provides related keywords of interest for the user, as they belong to its research domain, but they might be also original and new connections as they are the result of a bisociation process.

3.3 Linking nodes

As explained in Section 3.2, nodes are associated with a keyword and a set of documents in which this keyword occurs with a certain term frequency (TF). Practically, this is represented using a vector of real values containing, for each document, the term frequency of the node's keyword. In order to determine whether a link should be created between two nodes or not, and if there is to be a link, to assign it a weight, we have to use a similarity measure to compare two nodes (that is to say: the two vectors of TF values).

One basic metric that directly suggests itself is an adaptation of the Jaccard index (see [12]), shown in Equation 3, to this case. Then $|A \cap B|$ represents the number of elements at the same index that both have a positive value in the two vectors and $|A \cup B|$ the total number of elements in the two vectors.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \tag{3}$$

This index makes one able to compare two nodes according to the number of similar elements it contains, but does not take into account the importance of the text frequency values. It can also be interpreted as a probability, namely the probability that both elements are positive, given that at least one is positive.

In the Jaccard measure, as applied above, we would consider only whether a vector element is zero or positive and thus neglect the actual value (if it is positive). However, considering two elements at the same index i in two vectors, one way of taking their values into account would be to use their absolute difference (that is, in our case, the absolute difference of the TF values for two terms, but the same document). With this approach, it is easy to compare two vectors (of TF values) by simply summing these values and dividing by the total number of values (or the total number of elements that are positive in at least one vector).

However, this procedure does not properly take into account that both values have to be strictly positive, because a vanishing TF value means that the two keywords do not co-occur in the corresponding document. In addition, we have to keep in mind that having two elements, both of which have a TF value of 0.2, should be less important than having two elements with a TF value of 0.9. In the first case, the keywords associated with the two nodes we are comparing appear only rarely in the document with index i. On the other hand, in the latter case these keywords appear very frequently in this document, which means that they are strongly linked according to this document.

A possibility of taking the TF values itself (and not only their difference) into account is to use the product of the two TF values as a coefficient to the (absolute) difference between the TF values. This takes care of the fact that the two TF values have to be positive, and that the similarity value should be the greater, the larger the TF values are (and, of course, the smaller their absolute difference is). However, in our case, we also want to take into account that it is better to have two similar TF values of 0.35 (which means that the two keywords both appear rather infrequently in the document) than to have TF values of 0.3 and 0.7 (which means the first keywords appears quite rarely, while the other quite frequently).

In order to adapt the product to this consideration, we use the expression in Equation 4, in which k can be adjusted according to the importance one is willing to give to low TF values.

$$\sqrt[k]{\mathrm{tf}_i^A \cdot \mathrm{tf}_i^B} \cdot (1 - |\,\mathrm{tf}_i^A - \mathrm{tf}_i^B\,|), \quad \mathrm{tf}_i^A, \mathrm{tf}_i^B \in [0,1] \tag{4}$$

Still another thing that we have to take into account in our case is that the same difference between tf_i^A and tf_i^B can have a different impact depending on whether tf_i^A and tf_i^B are large or small. To tackle this issue, we combine Equation 4 with the use of the arctan function, which allows us to end up with a similarity measure shown in Equation 5. This form has the advantage that it takes into account that two TF values for the same index have to be positive, that the similarity should be the greater, the larger the TF values are, and that the same difference between tf_i^A and tf_i^B should have a different impact according to the values of tf_i^A and tf_i^B.

$$\sqrt[k]{\mathrm{tf}_i^A \cdot \mathrm{tf}_i^B} \cdot \left(1 - \frac{|\arctan(\mathrm{tf}_i^A) - \arctan(\mathrm{tf}_i^B)|}{\arctan(1)}\right), \quad \mathrm{tf}_i^A, \mathrm{tf}_i^B \in [0,1] \qquad (5)$$

Links in our BisoNets are weighted using the similarity measure shown in Equation 5. This approach allows us to use several different kinds of graph mining algorithms, such as simply thresholding the values to select a subset of the edges, or more complex ones, like calculating, for example, shortest paths.

4 The Swanson Benchmark

Having shown how BisoNets can be built from textual data sources, we present a benchmark application in this section. The idea is to provide a proof of principle, that this approach of creating a BisoNet can help a user to discover bisociations.

Swanson's approach [13] to literature-based discovery of hidden relations between concepts A and C via intermediate B-terms is the following: if there is no known direct relation A-C, but there are published relations A-B and B-C one can hypothesize that there is a plausible, novel, yet unpublished indirect relation A-C. In this case the B-terms take the role of *bridging concepts*. In his paper [13], Swanson investigated plausible connections between migraine (A) and magnesium (C), based on the titles of papers published before 1987. He found eleven indirect relations (via bridging concepts B) suggesting that magnesium deficiency may be causing migraine.

We tried our approach on the Swansons data source which consists of 8000 paper titles, taken from the PubMed database, published before 1987 and talking about either migraine or magnesium, to see if it was possible to find again these relations between migraine and magnesium. In order to generate a BisoNet, we implemented a parser for text files containing the data from PubMed able to export them in the format understandable by the second layer of our framework. Then, this second layer performed the keywords extraction, using these keywords as nodes and linking these nodes the way described in Section 3.

For testing purpose, we gave the name "domain A" to the data coming from the file concerning migraine, and "domain B" to the data coming from the one concerning magnesium as the purpose here is to discover links crossing domains. Whereas the data of the benchmark was make selecting papers talking about magnesium and paper talking about migraine, we could think about another Swanson-like benchmark selecting papers from the "diseases" domain and others from the "molecules" domain in order to have a larger investigation field.

Given this BisoNet, using a simple threshold filtering the less important nodes and links makes us able to discover indirect relations between magnesium and

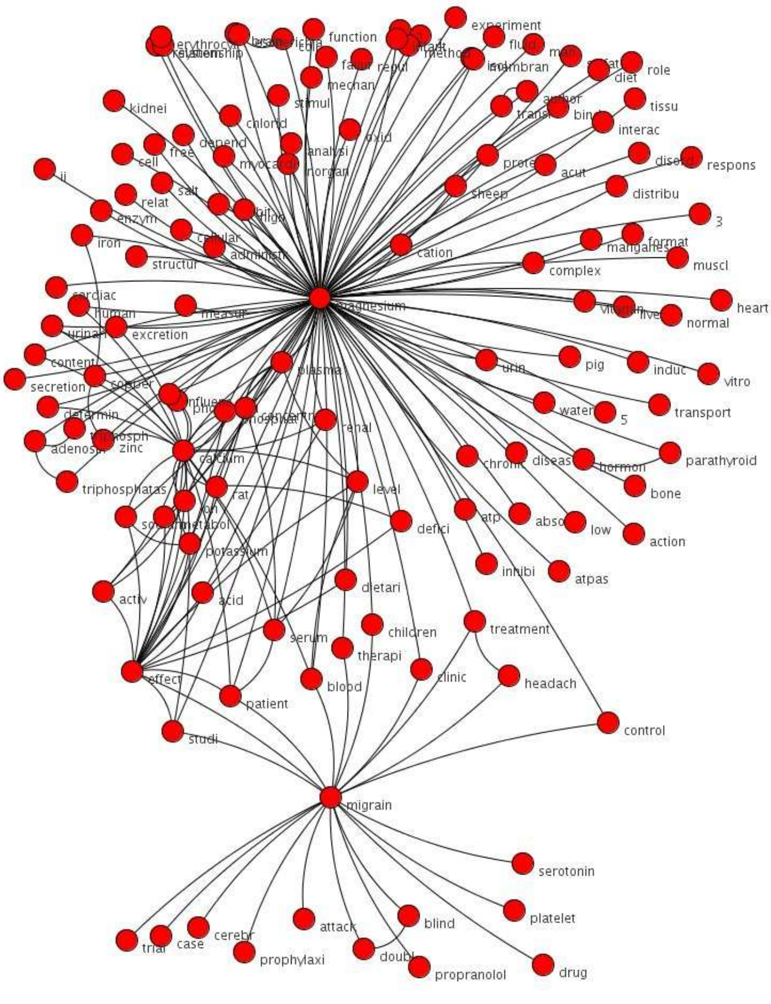

Fig. 2. An example of a BisoNet generated from the Swanson benchmark data sources.

migraine, relations that use keywords, belonging to both domains A and B, such as “deficit”, “headache”, “therapy” or “treatment” to link the two concepts we are talking about. This can be easily seen just looking at Figure 2.

5 Conclusion and further work

In this article, we provided a definition of the notion of a bisociation, as understood by Koestler, which is the key notion of the BISON project. Building on this defintion, we then defined the concept of a BisoNet, which is a network bringing together data sources from different domains, and therefore may help a user to discover bisociations. We presented a way we create nodes using simple text-mining techniques, and a procedure to generate links between nodes, which is based on comparing text frequency vectors using a new similarity measure we introduced.

We then perform a benchmark in order to rediscover bisociations between magnesium and migraine that have been discovered by Swanson using articles published before 1987. We see that bisociations between these two terms are easily discovered using the generated BisoNet, thus indicating that BisoNets are a promising technology for such search.

In summary, we venture to say that this work can be easily applied to any kind of textual data source in order to mine data looking for bisociations, thanks to the two layers architecture implementation. In addition, we are working on generalizing these techniques to non-textual data sources, introducing different types of attributes for the nodes, and therefore, other types of similarity measures in order to link the heterogeneous set of nodes. Further work also consists in performing other benchmarks and applying graph mining algorithms in order to confirm the quality of the so generated BisoNets.

The work presented here was supported by the European Commission under the 7th Framework Programme FP7-ICT-2007-C FET-Open, contract no. BISON-211898.

References

1. Koestler, A.: The act of creation. London Hutchinson (1964)
2. F.Barron: Putting creativity to work. In: The nature of creativity. Cambridge Univ. Press (1988)
3. Cormac, E.M.: A cognitive theory of metaphor. MIT Pess (1985)
4. R. Agrawal, T. Imielinski, A.S.: Mining association rules between sets of items in large databases. In: Proceedings of the ACM SIGMOD internation conference on management of data. (1993) 207–216
5. D.J. Chalmers, R.M. French, D.H.: High-level perception, representation and analogy: a critique of artificial intelligence methodology. Journal of Experimental and Theoretical Artificial Intelligence **4** (1992) 185–211
6. B. Falkenhainer, K.D. Forbus, D.G.: The structure mapping engine: algorithm and examples. Artificial Intelligence **41** (1989) 1–63

7. Barnden, J.: An implemented system for metaphor-based reasoning - with special application to reasoning about agents. In: Lecture Notes in Computer Science. Volume 1562. (1999) 143–153
8. Aamodt, A., Plaza, E.: Case-based reasoning: foundational issues, methodological variations and system approaches. Artificial Intelligence Communications **7**(1) (1994) 39–59
9. Cardoso, A., Costa, E., Machado, P., Pereira, F., Gomes, P.: An architecture for hybrid creative reasoning. In: Soft Computing in Case Based Reasoning. Springer Berlin/Heidelberg (2000)
10. van Rijsbergen, C., Robertson, S., Porter, M.: New models in probabilistic information retrieval. In: British Library Research and Development Report. Number 5587. London British Library (1980)
11. Salton, G., Gill, M.M.: Introduction to modern information retrieval. McGraw-Hill (1983)
12. Jaccard, P.: Étude comparative de la distribution florale dans une portion des alpes et du jura. Bulletin de la Société Vaudoise des Sciences Naturelles **37** (1901) 547–579
13. Don R. Swanson, Neil R. Smalheiser, V.I.T.: Ranking indirect connections in literature-based discovery: The role of medical subject headings. Journal of the American Society for Information Science and Technology (JASIST) **57**(11) (September 2006)