Extracting Protein-Protein Interaction Sentences by Applying Rough Set Data Analysis

Filip Ginter[1,2], Tapio Pahikkala[1,2], Sampo Pyysalo[1,2], Jorma Boberg[1,2], Jouni Järvinen[1,2], and Tapio Salakoski[1,2]

[1] Turku Centre for Computer Science (TUCS)
[2] Department of Information Technology, University of Turku
Lemminkäisenkatu 14 A, FIN-20520 Turku, Finland

Abstract. In this paper, we introduce a way to apply rough set data analysis to the problem of extracting protein-protein interaction sentences in biomedical literature. Our approach builds on decision rules of protein names, interaction words, and their mutual positions in sentences. In order to broaden the set of potential interaction words, we develop a morphological model which generates spelling and inflection variants of the interaction words. We evaluate the performance of the proposed method on a hand-tagged dataset of 1894 sentences and show a precision-recall break-even performance of 79,8% by using leave-one-out cross-validation.

1 Protein-Protein Interactions

The amount of published knowledge in the biomedical domain is overwhelming, and due to the introduction of high-throughput methods it grows at an unprecedented rate. While there are a number of efforts to identify and store information in databases [1, 2], most knowledge remains available only in unstructured text form in scientific articles and their abstracts. The premier bibliographic database in the field, PubMed[1], contains over 14 million citations and more than 7 million abstracts. The amount of data makes manual information extraction a formidable task and is enough to deluge computationally expensive automatic information extraction systems such as those based on sentence parsing [3].

A number of projects have focused on the development of methods for extracting protein-protein interactions (see e.g. [4] for a recent survey). While most of the previous work has concentrated on extraction on the level of publication abstracts, we present a method for extracting sentences actually describing protein-protein interactions. The proposed method is highly efficient and it is additionally capable of explaining the decisions that have been made.

2 Information Systems and Decision Rules

This section is devoted to some basic concepts of rough set data analysis. We adopt here the notation used by Pawlak in [5]. An *information system* is a pair $\mathcal{S} = (U, A)$,

[1] `http://www.ncbi.nlm.nih.gov/PubMed/`

S. Tsumoto et al. (Eds.): RSCTC 2004, LNAI 3066, pp. 780–785, 2004.

where U is a finite nonempty sets of *objects* called the *universe* and A is a finite sets of *attributes*. With every attribute $a \in A$, we associate a set V_a of its *values*. Furthermore, each attribute $a \in A$ is a mapping $a: U \to V_a$.

Let $\mathcal{S} = (U, A)$ be an information system. With every subset $B \subseteq A$, we may associate a language For(B). Formulas of For(B) are built up of attribute-value pairs (a, v), where $a \in B$ and $v \in V_a$, by means of logical connectives $\vee$, $\wedge$, and $\neg$ in a canonical way.

For any $\Phi \in$ For(B), we denote by $\|\Phi\|_{\mathcal{S}}$ the set of all objects $x \in U$ satisfying Φ in $\mathcal{S}$, and refer it to as the *meaning* of Φ in $\mathcal{S}$. The meaning $\|\Phi\|_{\mathcal{S}}$ of Φ in $\mathcal{S}$ is defined inductively as follows:

$$\begin{aligned}
\|(a, v)\|_{\mathcal{S}} &= \{x \in U \mid a(x) = v\};\\
\|\Phi \vee \Psi\|_{\mathcal{S}} &= \|\Phi\|_{\mathcal{S}} \cup \|\Psi\|_{\mathcal{S}};\\
\|\Phi \wedge \Psi\|_{\mathcal{S}} &= \|\Phi\|_{\mathcal{S}} \cap \|\Psi\|_{\mathcal{S}};\\
\|\neg\Phi\|_{\mathcal{S}} &= (\|\Phi\|_{\mathcal{S}})^{\mathtt{C}}.
\end{aligned}$$

If we distinguish in an information system $\mathcal{S} = (U, A)$ two subsets C and D of A, called *condition* and *decisions attributes*, respectively, then the system will be called a *decision table*. We assume that $C \cap D = \emptyset$ and $C \cup D = A$. A *decision rule* in $\mathcal{S}$ is an expression $\Phi \to \Psi$, where $\Phi \in$ For(C) and $\Psi \in$ For(D), and C and D are the condition and the decision attributes of $\mathcal{S}$, respectively. With every decision rule $\Phi \to \Psi$, we associate a *certainty factor*

$$\text{cer}_{\mathcal{S}}(\Phi \to \Psi) = \frac{\text{card}(\|\Phi \wedge \Psi\|_{\mathcal{S}})}{\text{card}(\|\Phi\|_{\mathcal{S}})}.$$

We will also use a *coverage factor* of the decision rule $\Phi \to \Psi$ defined by

$$\text{cov}_{\mathcal{S}}(\Phi \to \Psi) = \frac{\text{card}(\|\Phi \wedge \Psi\|_{\mathcal{S}})}{\text{card}(\|\Psi\|_{\mathcal{S}})}.$$

The certainty factor is the frequency of objects having the property Ψ among the objects which have the property Φ, and the coverage factor is the frequency of the objects having the property Φ in the set of objects which have the property Ψ.

3 Decision Rules Based on Interaction Words

The set of interaction words is "rough" (or "fuzzy") in a sense that some instances of a certain word describe a protein-protein interaction, but other instances do not. *Potential interaction words* are the words that have at least once acted as an interaction word in some sentence describing an interaction.

We assume that protein-protein interaction sentences always contain at least two *proteins* that are interacting, and an interaction word describing the type of the interaction. Therefore, in this paper we concentrate only on sentences which contain at least two protein names. For each potential interaction word appearing in such a sentence, we consider three possibilities how the word can be related to the proteins of the sentence:

Table 1. A simple decision table

U	WORD	POSITION	INTERACTION
x_1	activity	after	no
x_2	link	after	no
x_3	repair	after	no
x_4	required	after	no
x_5	binds	middle	yes
x_6	form	middle	yes
x_7	link	middle	yes
x_8	regulate	middle	yes

a potential interaction word may appear *before* all the proteins, in the *middle* of some proteins, or *after* all the proteins. For example, let us consider the following sentences; the first can be found in [6] and the second in [7].

1. "All our data are consistent with models in which `RAD17`, `RAD24` and `MEC3` are coordinately *required* for the *activity* of one or more DNA *repair* pathways that *link* DNA damage to cell cycle arrest."
2. "The head domain of `talin` thus *binds* to `integrins` to *form* a *link* to the `actin` cytoskeleton and can thus *regulate* `integrin` function."

The first sentence contains three proteins `MEC3`, `RAD17`, and `RAD24`, and four words *activity*, *link*, *repair*, and *required* that are potential interaction words, but this sentence does not describe an interaction. The second sentence contains the proteins `actin`, `integrins`, and `talin`, and the potential interaction words *binds*, *form*, *link*, and *regulate*. This sentence describes several protein-protein interactions.

Let us now consider more formally the structure of the decision table $\mathcal{S} = (U, A)$. The set of objects U is such that each object in U corresponds to one instance of one potential interaction word. The attribute set of $\mathcal{S}$ contains the condition attributes WORD and POSITION and the decision attribute INTERACTION. The value set of the attribute WORD contains all known potential interaction words and the value set of POSITION contains the values "before", "middle", and "after". The decision attribute INTERACTION may have the values "yes" or "no". The above sentences can be represented by a decision table given in Table 1. Note that the word *link* has now two roles: *link* describes an interaction in the second sentence, but not in the first one.

Next we will describe how we build up the decision table $\mathcal{S}$. Suppose that we have a training set of sentences, labeled as either interaction or non-interaction sentences. It is also essential that the protein names are tagged in the sentences. Note that there exist several methods for automatic identification of protein names in biomedical text, many of which are evaluated in [4]. Furthermore, the training sentences that are labeled as interaction sentences must be tagged to recognize the words that actually describe protein-protein interactions – these words form the set of potential interaction words V_{WORD}. After obtaining the set V_{WORD}, the set of training sentences must be re-tagged to further identify all potential interaction words in them (even though these potential interaction words do not necessarily describe interactions in these sentences). After the tagging phase the decision table $\mathcal{S}$ can be formed as described above.

We end this section by presenting how we actually make the decisions. Let us assume that we have a decision table $\mathcal{S}$ obtained by examining some training set of sentences. Let us then consider a sentence Σ, not belonging to the set of training examples, which contains at least two protein names. We can represent each instance of a potential interaction word in Σ by a formula $\Phi \in \mathrm{For}(C)$, where C is the set of condition attributes of $\mathcal{S}$, that is, $C = \{\textsc{Word}, \textsc{Position}\}$. For instance, if Σ contains the potential interaction word *bind* in the middle of some protein names of Σ, we may represent this situation by the formula $(\textsc{Word}, \mathrm{bind}) \wedge (\textsc{Position}, \mathrm{middle})$. We also expand all these formulas Φ to a decision rule of the form $\Phi \rightarrow (\textsc{Interaction}, \mathrm{yes})$. Since each instance of a potential interaction word induces one such rule, we may attach a set $R(\Sigma)$ of decision rules to the sentences Σ. Further, for each decision rule $\Phi \rightarrow \Psi \in R(\Sigma)$, we may now compute the certainty factor $\mathrm{cer}_{\mathcal{S}}(\Phi \rightarrow \Psi)$ and the coverage factor $\mathrm{cov}_{\mathcal{S}}(\Phi \rightarrow \Psi)$ determined by $\mathcal{S}$. In our experiments we assumed that $R(\Sigma)$ can possibly be a multiset, because a potential interaction word can appear twice in a similar position in the same sentence.

Because there are usually several rules that can be applied for a sentence Σ, the problem is now how to combine the information given by the certainty and the coverage factors of the rules in $R(\Sigma)$. Of course, there are several ways to do this. Suppose that Σ contains k instances of potential interaction words; this means that $R(\Sigma)$ contains k rules, that is, $R(\Sigma) = \{\Phi_i \rightarrow \Psi_i \mid 1 \leq i \leq k\}$. We introduce the following aggregate value of the rules related to the sentence Σ:

$$\mathrm{agg}_{\mathcal{S}}(\Sigma) = \sum_{i=1}^{k} \mathrm{cer}_{\mathcal{S}}(\Phi_i \rightarrow \Psi_i) \cdot \mathrm{cov}_{\mathcal{S}}(\Phi_i \rightarrow \Psi_i).$$

This value can be interpreted so that the greater $\mathrm{agg}_{\mathcal{S}}(\Sigma)$ is, the more likely the sentence Σ describes an interaction.

Let us also define for the sake of generality that if Σ is a sentence that does not contain at least two protein names or $R(\Sigma) = \emptyset$, then $\mathrm{agg}_{\mathcal{S}}(\Sigma) = 0$.

4 Morphological Model of Interaction Words

The set of potential interaction words $V_{\textsc{Word}}$ needs to be as broad as possible. We start with a set consisting of words that describe a protein-protein interaction in some training sentence labeled as an interaction sentence – in our experiments, which are described in Section 5, the set $V_{\textsc{Word}}$ originally contained 354 unique words. Initially the set $V_{\textsc{Word}}$ contains only the inflected forms occurring in the corpus. Thus, for example, the set $V_{\textsc{Word}}$ might contain the word *activated*, but not the words *activates, activate*, or *activating*. A straightforward way to extend the set is to consider all inflected forms of the words, as opposed to only the forms in which they actually appear in the corpus. It is further important to consider spelling variations like *localize/localise*, *labeled/labelled*, *coactivate/co-activate*.

We develop a two-level morphological model, a formalism introduced by Koskenniemi [8], which we use to generate the various inflections, spelling variants, and derivations of the words in the set $V_{\textsc{Word}}$ obtaining a set of 158 verb stems, 41 noun stems, and 14 adjective or adverb stems. From these 213 stems, our model generates 1304 forms, considerably extending the set $V_{\textsc{Word}}$.

5 Experimental Setting

We compiled a corpus of 1894 sentences manually annotated as containing an interaction (1114 sentences) or not containing an interaction (780 sentences). Additionally the proteins were identified in all sentences, and in the interaction sentences also the interaction words were marked up.

We used the leave-one-out crossvalidation to evaluate the proposed method. We excluded each sentence Σ in turn from the corpus and obtained from the remaining sentences the decision table $\mathcal{S}$ so that first we formed the set V_{WORD} consisting of interaction words of interaction sentences and then expanded V_{WORD} by applying the morphological model of Section 4. Then we re-tagged the sentences by marking each word from the expanded V_{WORD} as a potential interaction word, and finally we formed the actual table $\mathcal{S}$ from these tagged sentences. When computing the $\text{agg}_{\mathcal{S}}(\Sigma)$-value for the excluded sentence Σ, we first searched all the words of V_{WORD} of $\mathcal{S}$ which can be found in Σ. Then we constructed $R(\Sigma)$ and for each rule $\Phi \to \Psi \in R(\Sigma)$ we computed the certainty and the coverage factors that were used to compute $\text{agg}_{\mathcal{S}}(\Sigma)$.

A straightforward way to construct a classifier is to use a suitable threshold for $\text{agg}_{\mathcal{S}}(\Sigma)$ when deciding whether an unseen sentence is an interaction sentence. However, the selection of such a threshold is a non-trivial and vague task. Therefore, we proceed in a different manner to find out the capability of our method. Namely, we may order the sentences in a descending order with respect to their $\text{agg}_{\mathcal{S}}(\Sigma)$-values to obtain a list L of sentences ordered in such a manner that if a sentence Σ_1 appears in L before a sentence Σ_2, then Σ_1 is more likely an interaction sentence than Σ_2. We noticed in our experiments that if we take the first 50 sentences from L, 49 of them are interaction sentences. Similarly, when selecting 100, 150, and 200 leading sentences of L, we obtain 95, 140, and 189 interaction sentences, respectively.

We also compute the precision-recall curve from this ordered list L of sentences in such a way that we move sentences one by one from the beginning of L to the set of sentences classified as interaction sentences, and the sentences remaining in L form the set of sentences classified as non-interaction sentences. At each step we compute the corresponding precision and recall values based on the current classification. The precision-recall curve is presented in Figure 1. The value of the precision-recall break-even point, the point at which precision and recall values are equal, is 79.8%.

Conclusions

In this paper, we introduced a way to apply rough set data analysis to the protein-protein interaction extraction problem. The method was based on decision rules of proteins, interaction words, and their mutual locations in sentences. We also presented a way to combine information given by the certainty and coverage factors of the rules related to a certain sentence. Our method is efficient and it also has the advantage that it is capable of explaining the decisions that have been made.

Acknowledgment

This work was supported by Tekes, the National Technology Agency of Finland.

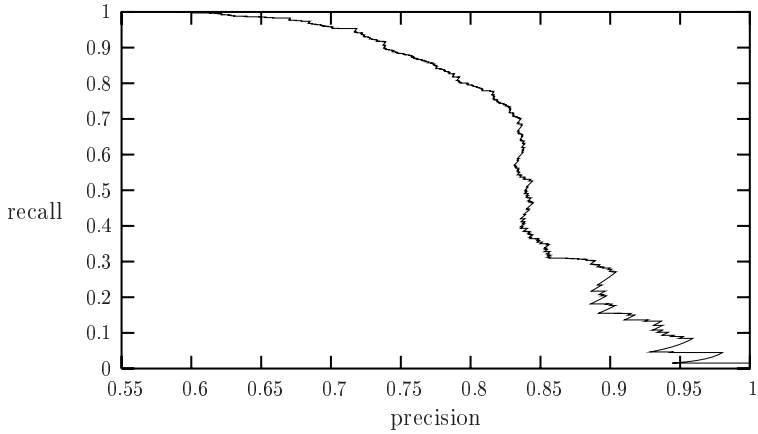

Fig. 1. The precision-recall curve

References

1. Bader, G., Donaldson, I., Wolting, C., Ouellette, B., Pawson, T., Hogue, C.: BIND – the biomolecular interaction network database. Nucleic Acids Research **29** (2001) 242–245
2. Xenarios, I., Rice, D., Salwinski, L., Baron, M., Marcotte, E., Eisenberg, D.: DIP: The database of interacting proteins. Nucleic Acids Research **28** (2000) 289–291
3. Temkin, J., Gilder, M.: Extraction of protein interaction information from unstructured text using a context-free grammar. Bioinformatics **19** (2003) 2046–2053
4. Bunescu, R., Ge, R., Kate, R., Marcotte, E.M., Mooney, R., Ramani, A.K., Wong, Y.W.: Comparative experiments on learning information extractors for proteins and their interactions. Artificial Intelligence in Medicine. Special Issue on Summarization and Information Extraction from Medical Documents (2004) (to appear)
5. Pawlak, Z.: Rough sets, decision algorithms and Bayes' theorem. European Journal of Operational Research **136** (2002) 181–189
6. Lydall, D., Weiner, T.: G2/M checkpoint genes of saccharomyces cerevisiae: further evidence for roles in DNA replication and/or repair. Molecular and General Genetic **256** (1997) 638–651
7. Calderwood, D., Zent, R., Grant, R., Rees, D., Hynes, R., Ginsberg, M.: The talin head domain binds to integrin beta subunit cytoplasmic tails and regulates integrin activation. The Journal of Biological Chemistry **274** (1999) 28071–28074
8. Koskenniemi, K.: Two-level model for morphological analysis. In Bundy, A., ed.: Proceedings of the Eighth International Joint Conference on Artificial Intelligence, 8-12 August 1983, Karlsruhe, West Germany, William Kaufmann, Inc. (1983) 683–685