On Building a Quantitative Food-Disease-Gene Network

Hui Yang*  
Dept of Computer Science  
San Francisco State Univ.  
San Francisco, CA, 94132  
huiyang@sfsu.edu

Abhishek Sharma‡  
Dept of Computer Science  
San Francisco State Univ.  
San Francisco, CA, 94132  
asharma1@sfsu.edu

Rajesh Swaminathan ‡  
Dept of Computer Science  
San Francisco State Univ.  
San Francisco, CA, 94132  
rajeshs@sfsu.edu

Vilas Ketkar  
San Francisco State Univ.  
vilask@sfsu.edu

Abstract

Nutritional genomics is a new science that studies the relationship between foods (or nutrients), diseases, and genes. Large amounts of scientific findings have been published in this area, primarily in unstructured text. Moreover, given a pair of entities, different studies can report different findings. It is hence important to obtain a holistic view of the reported relationships. In this article, we describe an information extraction system aiming to reach this goal. The system integrates natural language processing techniques, domain ontology, statistical, and machine learning methods. It consists of four main modules: (1) entity extraction, which recognizes and extracts five types of entities: foods, chemicals (or nutrients), diseases, proteins and genes; (2) relationship extraction, which extracts binary relationships between entities; (3) relationship polarity analysis, which categorizes relationships into three groups: positive, negative, and neutral; and (4) strength analysis, which rates a relationship as weak, medium, or strong. To the best of our knowledge, we are the first to propose to analyze the polarity and strength of a binary relationship. We have evaluated our system using the GENIA corpus and datasets drawn from the MEDLINE database. The first two modules outperform the reported best results with an average F-score of 0.89 and 0.82, respectively; while the last two also achieve promising results with an accuracy of 0.75-0.84 and ~0.90, respectively.

Key words: nutritional genomics, text mining, relationship extraction, relationship polarity, relationship strength

1 INTRODUCTION

Advances in bio-technology and life sciences are leading to an ever-increasing volume of published research data, predominantly in unstructured text (or natural language). At the time of writing, the MEDLINE database consists of 19 million scientific articles with a growth rate of ~400,000 articles per year [8]. This phenomenon becomes even more apparent in nutritional genomics, an emerging new science that studies the relationship between foods (or nutrients), diseases, and genes [16]. For instance, soy products and green tea have been two of the intensively studied foods in this new discipline due to their controversial relationship with cancer. A search to the MEDLINE database on “soy and cancer” renders a total of 1,287 articles, and a search on “green tea and cancer” renders 1,318 articles. Due to the large number of publications every year, it is unrealistic for even the most motivated to manually go through these articles to obtain a full picture of the findings reported to date. This however has become ever more important and necessary due to the following reasons: (1) given a pair of entities, e.g., green tea and cancer, different studies might report different findings with respect to their relationship. For example, Sonn et al. suggest that “green tea is beneficial to the treatment of cancer” [25], whereas Sauvaget et al. conclude that these two are not related [23]. In other words, a relationship can be positive (good), negative (bad), or neutral. We term this as the relationship polarity; and (2) even if different studies agree with each other on the relationship polarity between two entities, they may report it with a different level of decisiveness. As another example, one study suggests that “soy intake … may protect against breast cancer …” [19], while another study indicates “soy intake is believed to be an essential factor for the incidence of hormone-dependent tumors (e.g., breast cancer) …” [28]. Obviously, the latter is more decisive than the former. We term the decisiveness of a relationship as the relationship strength.

In this article, we propose to develop an information extraction system that automatically (1) extracts the binary relationships between foods, diseases and genes; and (2) analyze the polarity and strength of these relationships. The long-term goal is to build food-disease-gene networks that statistically quantify the various relationships reported in nutritional genomics.

To reach this goal, one need to first accurately recognize and extract the terms that describe foods, diseases, and genes. In addition, we will also need to recognize chemicals (nutrients) in foods and proteins. This is because: (1) given a whole food (e.g., soy), scientific research often focuses on understanding how different organic compounds (e.g., genistein in soy) contained within the food impact certain diseases (e.g., breast cancer); and (2) genes and their protein products are often used interchangeably in practice. Past efforts in this area, termed as Named Entity Recognition (NER), have been on genes or proteins. (See the reviews by Cohen and Skusa et al. [2] [24] for a list of works in this area.) The best reported F-scores on NER are generally between 0.75–0.85 [24]. We adopt an approach that utilizes domain ontology, statistics, and syntactic information for this task and achieve an average F-score of 0.89.

* Corresponding author. ‡: Primary student contributors.
The next task is to extract the binary relationship between two entities. Three main approaches—co-occurrence based, rule based and machine learning—have been adopted for this purpose in the past, again with a focus on relationships involving either proteins or genes [24]. These approaches can typically deliver a reasonable precision range (0.60–0.80), but with a low recall (~0.50). We have employed an approach that utilizes both syntactic and semantic information to extract binary relationships. Facilitated by external sources such as VerbNet [12], WordNet [13], and the UMLS (United Medical Language System) Metathesaurus [10], our approach achieves a balanced F-score of ~0.82.

The last two tasks in the system are to analyze the polarity and strength of a relationship. As mentioned earlier, they are important towards deriving food-gene-disease networks, due to the presence of different and sometimes conflicting findings with regard to the relationship between a certain food and disease (e.g., soy beans and breast cancer). As far as we know, we are the first to propose analyzing the polarity and strength of a binary relationship. Nonetheless, these tasks are related to the opinion mining problem [14], which mostly focuses on analyzing the polarity and strength of reviews on a single product, e.g., a movie. In addition, it often mainly relies on adjectives and adverbs. However, it is insufficient in our case, because most binary relationships are action-based (e.g., “soy intake reduces the risk of cancer”) and scientific authors often use adjectives and adverbs sparingly. Furthermore, the entities involved in a relationship can affect the polarity of a relationship. For instance, the previous example describes a positive relationship, because the object “cancer” is a disease. We have built a 2-phase Support Vector Machine for polarity analysis, and a Support Vector Regression for strength analysis [15].

2 RELATED WORK

The proposed system is related to three areas: named entity recognition, bio-relationship extraction, and opinion mining. We next review the most germane studies. Readers are referred to [1][2][14][24] for more in-depth discussions.

Named Entity Recognition (NER): In the biomedical field, efforts in NER have been primarily on genes and proteins. Three main approaches have been employed for NER: lexicon-based, rule-based and machine learning approaches [2][24]. Lexicon-based approaches rely on well-defined vocabularies and recognize entities based on string match. Rule-based approaches on the other hand use rules or regular expressions to capture the lexical features exhibited by typical gene or protein names. Finally, machine learning approaches such as SVM and Hidden Markov Model have also been applied for NER [17][30].

Relationship extraction: This task has recently become an area of interest, resulting in many studies [2][24]. Four mainstream approaches have been used for relationship extraction: co-occurrence based, NLP (Natural Language Processing) based, machine learning, and link based. The co-occurrence approach infers a relationship between two entities if they frequently collocate with each other. NLP based approaches perform sentence parsing to obtain small syntactic units. Syntactic templates such as noun-verb-noun are then used to extract relationships. Machine learning approaches such as SVM have also been used to extract relationships. Finally, link-based methods extend the co-occurrence approach to infer relationships between entities if they co-occur with a common term [26].

Polarity and strength analysis: These two tasks are related to the opinion mining problem that is often applied to determine whether a product review or opinion is positive or negative [5][14][21][27]. Both supervised and unsupervised approaches have been applied for polarity analysis in opinion mining, whereas supervised approaches such as SVM are commonly used to rate the strength of an opinion [22][27][29]. These approaches in general mainly rely on adjectives and adverbs, which are not applicable to relationship polarity analysis as discussed in Section 1.

Finally, there exist a number of systems that resemble the proposed system. They are however mainly focus on genes and proteins and do not include polarity and strength analysis [6][24].

3 SYSTEM AND ALGORITHMS

Fig. 1 presents a schematic description of the proposed system. It includes four modules: (i) entity recognition, (ii) relationship extraction, (iii) polarity analysis, and (iv) strength analysis. We next discuss these modules in detail.

3.1 Data Acquisition

We utilize an in-house program to acquire scientific publications in Nutritional Genomics from the MEDLINE database [9]. Given a set of key words, this program automatically downloads all the relevant articles from MEDLINE and saves them on a local disk. Both abstracts and full texts (if available) will be downloaded. In this work, we only use the abstracts.

3.2 Entity Recognition and Extraction (ERE)

We take two main steps to extract terms that describe
foods, chemicals, diseases, genes, and proteins: (1) recognizing entities that can be mapped to established lexicons using both syntactic information, and heuristics; and (2) identifying co-references of the recognized entities and their abbreviated forms if any.

**Step 1. Entity recognition:** The UMLS Metathesaurus—the largest vocabulary that contains biomedical and health-related concepts, common terms, and the relationships among them—is used to facilitate the ERE task. Specifically, we use the companion software program MetaMap [2] to map terms in an abstract to those in the Metathesaurus. Given a phrase such as “gastric cancer”, MetaMap often returns multiple mappings. As shown in Fig. 2, a total of 5 mappings are returned, where each row identifies one mapping. The first two numbers are the mapping and evaluation scores in [0,1000] (a perfect score), and the last field identifies the semantic type (e.g., disease) of the term given a mapping.

1000 1000|C0024623|Malignant neoplasm of stomach|Gastric Cancer|neop|1000 1000|C0069797|Stomach Carcinoma|Gastric Cancer|neop|888 861|C003855|Stomach|Gastric|bpoc|Cancer Genes|Cancer|inv|888 861|C003855|Stomach|Gastric|bpoc|Malignant Neoplasms|Cancer|neop|888 861|C003855|Stomach|Gastric|bpoc|Specialty Typ cancer|Cancer|bmod|

**Fig. 2** The MetaMap results for the term “Gastric Cancer”, which is likely of the following semantic types: neoplastic process (neop), invertebrate (inv), and biomedical Occupation or Discipline (bmod).

Among all the returned mappings, which one is the most likely? To answer this, we employ two heuristics: (1) Let us denote the highest mapping score as $S_h$. We first select all the mappings whose mapping score is in $[S_h-\delta, S_h]$, where $\delta$ is a user-defined parameter and set to 50 in our evaluation; and (2) we then examine these selected mappings and select the most frequently mentioned semantic type to label the term of interest. Take the case in Fig. 2 as an example and let $\delta=150$. All five mapping will be selected, among which the semantic type “neop” (neoplastic process) is the most frequent (3 times). The term “gastric cancer” is therefore labeled as “neop”, which is a disease. The rationale behind these heuristics is that the mappings returned by MetaMap with the highest score are not necessarily the best mapping due to the stochastic nature of the algorithm. It is possible that one entity might be associated with multiple semantic types using the above approach. Note that we have manually grouped the 135 UMLS semantic types into the five types of interest after removing irrelevant types (e.g., bird).

**Improving food term extraction:** MetaMap can identify most food-describing phrases. However, it sometimes fails to correctly identify common whole foods, “kiwi” and “grape seeds”. This can be attributed to fact that UMLS is largely biomedical. To address this issue, we employ the USDA (United States Department of Agriculture) food database [7] as well. Given a sentence, we first build its parse tree using the Penn TreeBank parser [8]. We then focus on the noun phrases in the sentence. For each multi-word noun phrase, we consider all of its subsequences and obtain a mapping score in [0, 1] for each subsequence.

This score is determined by the following factors: (1) the relative length and location of a subsequence in the noun phrase: the longer and the closer to the end of the phrase, the better; (2) the location of the match in an USDA food entry: the nearer to the start, the better; and (3) the number of words matched with a USDA food entry: the more, the better. If the highest mapping score of a noun phrase is greater than a threshold (e.g., 0.5), we label it as food.

We next employ the UMLS-based and USDA-based approaches to finalize the list of food entities. A term will be labeled as food if it is labeled by either or both methods. The UMLS-based label is chosen should a conflict arise. This ensemble method proves to be effective and achieves an F-score of 0.945, compared to 0.85 and 0.64 if only UMLS or USDA database is used.

**Step 2. Abbreviations and co-references:** The use of abbreviations is a common practice in scientific writing, for instance, SERM for “selective estrogen receptor modulator”. Co-reference is another commonly employed linguistic feature where multiple phrases refer to the same referent. For instance, the pronoun “they” in the sentence “Soy isoflavones have numerous biological properties that suggest that they may protect against colorectal cancer.” is a co-reference of “soy isoflavones”. These two features however cannot be effectively handled by the previous step because (1) co-references often span multiple sentences and (2) there are no set rules for abbreviations and they often vary from individual to individual. As a result, abbreviation databases [3] cannot identify many abbreviations during our evaluation.

We identify abbreviations based on the observation that they are frequently specified within parentheses right after the long form. We have designed an algorithm to identify the following forms of abbreviations: (1) acronyms, e.g., SERM for “selective estrogen receptor modulator”; (2) those that correspond to a non-contiguous subsequence of the long form, e.g., “nucleophosmin-anaplastic lymphoma kinase” as NPM-ALK; and (3) the abbreviations that are constructed using a combination of initials and a whole word, e.g., “estrogen receptor beta” as ERBeta.

We use the OpenNLP toolkit [8] for co-reference recognition. It takes the entire abstract as input and identifies all the co-references. We then go over these co-references and retain those whose referent is an entity of interest, abbreviated or not.

3.3 Binary Relationship Extraction

We focus on extracting binary relationships that involve one or more previously identified entities. We observe that people often employ verbs or verb phrases to describe a relationship, for instance, the underlined verb phrase in “Soy food consumption may reduce the risk of fracture”. However, only certain verbs can be used to describe a relationship [19]. If identified, these verbs can be used to indicate the presence of a relationship. The question is: how to identify these verbs in a realistic yet effective way?

To identify the list of relationship-describing verbs, we start with the 54 relationships identified in the UMLS
Metathesaurus [10], each of which includes a verb such as "increase" and "associate". We extract these verbs and form the initial verb list. For each of these verbs, we then use VerbNet [11] to identify its semantic class and add all the verbs in this class to the verb list. We further expand this verb list by obtaining all the synonyms of each verb using WordNet [12]. These synonyms are added to the verb list too if they are not already present. The result is a comprehensive list of relationship-describing verbs.

We next use this verb list to recognize sentences that contain a relationship. Given an abstract, we only consider the sentences that contain at least one labeled entity, as the other sentences are unlikely to describe a relationship. For each entity-bearing sentence, we use its parse tree to identify the verb phrase(s). We then check whether the main verb (e.g., "reduce" in "may reduce the risk of") is in the list of relationship-describing verbs identified earlier. If yes, we say the sentence is relationship bearing.

For each relationship-bearing sentence, we need to further analyze it to identify the two involved parties. A natural solution is to identify the subject and object in the sentence, and consider them as the two parties. To this end, we utilize the Stanford NLP Parser [11]. This parser recognizes 55 grammatically dependent pairs (or relationships), such as nominal subject and direct object. We then use these dependency pairs to infer the subject and object in a sentence. Note that the subject or object is not necessarily an entity of interest.

Finally, for each relationship-bearing sentence, the program locates its relationship-describing verb phrase(s). It then extracts the noun phrases that are located nearest to the main verb on both sides, provided that these noun phrases are entities of interest, the subject term, or the object term. A 3-tuple (noun phrase, verb phrase, noun phrase) is then constructed to represent the binary relationship. Notice the difference between our method and traditional template-based methods, which use a template such as "noun-verb-noun" to identify binary relationships between two nouns. Our method does not require such a template. As seen in the evaluation, it achieves a balanced precision and recall.

### 3.4 Relationship Polarity and Strength Analysis

As discussed in Section 1, the polarity of a relationship can be positive, negative or neutral. In addition, a positive or negative relationship can also exhibit different levels of strength, i.e., the decisiveness of a relationship. In this work, we project the strength to three levels: 1-weak, 2-medium and 3-strong. See Tab. 1 for some examples.

<table>
<thead>
<tr>
<th>Example Sentences</th>
<th>Polarity</th>
<th>Strength</th>
</tr>
</thead>
<tbody>
<tr>
<td>Soy food consumption may reduce the risk of fracture ...</td>
<td>Positive</td>
<td>1-weak</td>
</tr>
<tr>
<td>Eating soybean pastes was associated with the increased risk of gastric cancer ...</td>
<td>Negative</td>
<td>2-medium</td>
</tr>
<tr>
<td>coffee might be associated with equal production ...</td>
<td>Neutral</td>
<td>n/a</td>
</tr>
</tbody>
</table>

Tab. 1 Polarity and strength of exemplar sentences.

We treat polarity analysis as a classification problem and the strength analysis as a prediction problem. Specifically, we have constructed a training dataset by manually labeling the polarity and strength of the binary relationships as extracted above. We then use this training set to learn a SVM classifier for polarity classification and a Support Vector Regression (SVR) model for strength prediction [15]. The input feature vectors to both SVM and SVR are identified as follows; (1) we first remove all the non-content words (e.g., the) from the training dataset; (2) we then identify all the unigrams (i.e., single words) in the training set. Denote the set of unigrams as $S_{uni}$; (3) we next augment $S_{uni}$ using both VerbNet and WordNet in a similar fashion as we construct the list of relationship-describing verbs for relationship extraction (Section 3.3). The difference here is we also consider non-verb words, whose synonyms from WordNet will be included. The rationale of this step is that it is often expensive to create training examples. Moreover, even with a large number of training examples available, they might still be biased. This step results in an augmented set of unigrams. Denote it as $S_{uni}'$, and (4) finally, each training example is represented as a binary feature vector of $|S_{uni}'|$ dimensions. The $i^{th}$ dimension is set to 1 if the $i^{th}$ unigram in $S_{uni}'$ appears in the training example, 0 otherwise.

We next apply the SVM-{light} program [15] to these vectors to build an SVM for polarity classification and an SVR for strength analysis. Since SVM is a binary classifier we build a 2-phase classifier for polarity classification. For instance, the first phase separates neutral from positive and negative, and the second phase separates positive from negative. In addition, at each phase, one can select among a variety of kernel functions, such as linear and radial bias functions. We have evaluated different combinations and empirically selected the following strategy to build the 2-phase SVM classifier for polarity classification: Phase 1 uses a linear kernel to build an SVM that separates neutral from positive and negative; and phase 2 uses a polynomial kernel that separates positive from negative.

### 4 Evaluation

We evaluate the four modules in the proposed system using datasets drawn from the MEDLINE database. In addition, we also evaluate the entity recognition module using the GENIA corpus [18]. We next first describe the datasets and then report the evaluation results.

#### 4.1 Evaluation Datasets

We download 1000 articles from the MEDLINE database with the keywords "soy" and "cancer" using our in-house program (Section 3.1). We then randomly select 50 abstracts and manually annotate them by a team of four individuals. Each person independently annotates these abstracts to identify: (1) the five types of entities, i.e., chemicals, foods, diseases, genes and proteins, including co-references and abbreviations; (2) all the binary relationships; and (3) the polarity and strength of each relationship. Below is an annotated sentence: "\textit{chemical} Soy isoflavones/ \textit{relationship} 2+ are likely to be protective of/ \textit{disease} colon cancer\textit{}/", where "2+" indicates a positive relationship with a medium strength.
The four individuals then review each other’s annotation and resolve their differences as a team. This dataset is referred to as Dataset 1. We also manually annotate the relationships with polarity and strengths using another 50 abstracts. We refer to this dataset as Dataset 2.

### 4.2 Entity Extraction Evaluation

We evaluate the entity extraction module using three measurements: precision (p), recall (r) and the F-score (2pr/p+r). Tab. 2 presents the evaluation results of the entity extraction module using Dataset 1. We observe that food entities achieve the best results. This is due to the usage of both the UMLS and USDA food databases. The recognition of co-references and abbreviations increases the combined F-score by 1.5%. Overall, our ER module achieves a balanced precision and recall. The combined F-score of 0.894 is better than the reported best F-scores of 0.75–0.85 [24]. This demonstrates the effectiveness of our algorithm, especially given that we need to extract five types of bio-entities, instead of just genes or proteins.

![Image](image_url)

Tab. 2 Evaluation results of the ERE module using Dataset 1.

<table>
<thead>
<tr>
<th>Entities</th>
<th>Precision</th>
<th>Recall</th>
<th>F-Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Food</td>
<td>0.9657</td>
<td>0.9306</td>
<td>0.945</td>
</tr>
<tr>
<td>Chemical</td>
<td>0.7718</td>
<td>0.8753</td>
<td>0.82</td>
</tr>
<tr>
<td>Disease</td>
<td>0.8148</td>
<td>0.9000</td>
<td>0.855</td>
</tr>
<tr>
<td>Protein-Gene</td>
<td>0.8796</td>
<td>0.8749</td>
<td>0.878</td>
</tr>
<tr>
<td>Combined</td>
<td>0.898</td>
<td>0.8986</td>
<td>0.894</td>
</tr>
</tbody>
</table>

Tab. 3 Evaluation comparison with other leading methods for ER.

We also evaluate the ER module using the GENIA V3.02 corpus [18]. This corpus consists of 2000 abstracts from MEDLINE extracted using the terms “human”, “blood cells” and “transcription factors”. Following a similar protocol in the JNLPBA entity recognition contest [17], we simplify the 36 entities types in GENIA to Proteins, Genes, and Chemicals. We then run our ER module over 500 abstracts randomly selected from the corpus. Tab. 3 lists the evaluation results of our algorithm and the top three performers in the JNLPBA contest. Note that not all these algorithms use supervised methods (e.g. SVM), are trained over 2000 abstracts and tested over 404 abstracts. In contrast, our algorithm is unsupervised and does not require training time. Tab. 3 shows that our method is comparable with these programs in precision, but with a significantly higher recall. This further demonstrates the efficacy and generality of our approach.

### 4.3 Relationship Extraction Evaluation

We evaluate our relationship extraction module using both Dataset 1 and Dataset 2. We perform approximate match of the programatically extracted relationship-describing phrases with the manually annotated ones and compute the precision, recall and F-score. Tab. 4 reports the evaluation results over both datasets. For each dataset, we also conduct a comparative study to observe whether our approach is location-sensitive by extracting relationships from the concluding sentences in an abstract and the full abstracts, respectively. From Tab. 4, we see that our approach is in general not biased and achieves a balanced precision (0.82–0.86) and recall (0.76–0.85). This is better than existing methods, which often deliver a reasonable precision (0.60–0.80), but with a low recall (~0.50) [24]. Two possible reasons we notice can contribute to the missed or misidentified relationships: (1) the manually annotated datasets are not perfect; and (2) the list of verbs included in the UMLS semantic relationships might be too restricted. We are currently investigating different approaches to further improve this module.

![Image](image_url)

Tab. 4 Evaluation results of the relationship extraction module.

### 4.4 Evaluation of Polarity Analysis

We first use Dataset 1 to evaluate the accuracy of different 2-phase SVM classifiers one can build by varying (a) the class to be separated in the first phase, and (b) the kernel function used at each phase. We have three options for factor (a), i.e., positive, negative or neutral, and four options for factor (b) at both phases, i.e., linear kernel (L), polynomial kernel (P), radial bias function (RB) and sigmoid kernel (S). This gives us a total of 3*4*4=48 options to build a 2-phase SVM. We split Dataset 1 into training and test data at a ratio of 80:20. We then compute the classification accuracy of each SVM. For comparison, we also implement a Naïve Bayesian classifier. Fig. 3 depicts the accuracies where the first phase separates neutral from positive and negative relationships. The x-axis indicates the kernel function used in the first phase and different colors are used to identify the kernel function used in the second phase. The accuracy from the Naïve Bayesian classifier is also included. We also evaluate the accuracies of SVMs that separate positive or negative from the rest in the first phase. They in general have a lower accuracy than that in Fig. 3. We omit them here due to space limit. From this figure, we observe that the linear-polynomial option delivers the best accuracy of 0.75, compared to 0.61 from the Naïve Bayesian classifier.

![Image](image_url)

Fig. 3 Accuracy of polarity analysis (neutral vs (pos. vs neg.))
currently exploring other strategies to improve the accuracy. For instance, we expect adding semantic information (e.g., the object is a disease entity) and bigrams to the feature vector will improve the accuracy.

4.5 Evaluation of Strength Analysis

To evaluate the performance of the strength analysis module, we train the SVR using 80% of examples in Dataset 1 and test it on the remaining 20%. We do not separate positive from negative relationships. Fig. 4 reports the accuracy based on 4 different kernels and that from the Naïve Bayesian classifier. The linear kernel renders the best accuracy of 0.90, compared to 0.78 from the Naïve Bayesian classifier. This result is promising, although the testing set is relatively small.

Fig. 4 Accuracy of strength analysis

5 CONCLUSIONS AND FUTURE WORK

We have presented a systematic approach to analyze articles in nutritional genomics towards building a food-disease-gene network. We propose an unsupervised method that uses domain lexicons, syntactic information, and heuristics to identify 5 types of bio-entities. A verb-centered approach facilitated by semantic role analysis is then used to identify binary relationships between entities. Finally, we propose to study the polarity and strength of a relationship, an important issue that has not been of much study in the field.

We are currently developing algorithms to integrate the extracted relationships by comparing the two involved parties. This will allow us to quantify a given relationship, for instance, to arrive at a result like “30 out of 40 articles have reported a positive relationship between soy intake and breast cancer”. In addition, we are implementing a visualization module so that users can interactively and intuitively view the identified relationships. Finally, we are annotating a larger dataset for further evaluation.

6 REFERENCES