BOOTSTRAPPING THE RECOGNITION AND ANAPHORIC LINKING OF NAMED ENTITIES IN *DROSOPHILA* ARTICLES

ANDREAS VLACHOS, CAROLINE GASPERIN, IAN LEWIN, TED BRISCOE

Computer Laboratory,
University of Cambridge,
15 JJ Thomson Avenue, CB3 0FD
E-mail: FirstName.LastName@cl.cam.ac.uk

This paper demonstrates how *Drosophila* gene name recognition and anaphoric linking of gene names and their products can be achieved using existing information in FlyBase and the Sequence Ontology. Extending an extant approach to gene name recognition we achieved a F-score of 0.8559, and we report a preliminary experiment using a baseline anaphora resolution algorithm. We also present guidelines for annotation of gene mentions in texts and outline how the resulting system is used to aid FlyBase curation.

1. Introduction

Curated databases are critical in the biomedical sciences as a method of systematizing and making accessible the rapidly expanding scientific literature[1,2]. However, curation is expensive because it requires considerable manual effort on the part of domain experts. In this paper, we describe the development of an adaptive textual information extraction (IE) system using bootstrapping machine learning techniques, designed to function as part of an interactive system to aid curation by supporting thematically-guided navigation of the article being curated in terms of the entities of interest.

Most IE systems for biomedical and other domains have been developed either using supervised machine learning techniques requiring large quantities of annotated data[3] or by manually encoding domain specific rules[4]. Here we describe how we have replicated and extended the approach of

Morgan *et al.*[5] using FlyBase[a] and the Sequence Ontology[6b] to bootstrap an initial unsupervised system with state-of-the-art performance.

The link to extant public-domain resources, such as FlyBase and the Sequence Ontology, both supports the initial automatic adaptation of the system and also provides essential functionality. The association of gene names with FlyBase gene identifiers is a useful extension of classic named entity recognition in the context of FlyBase curation. However, the means by which this link is obtained and by which the initial *Drosophila* gene name recognizer is bootstrapped also relies critically on the availability of such extant resources which, while not developed to support creation of IE systems, contain valuable information which can be exploited to adapt IE technology to the domain. Similarly, the Sequence Ontology encapsulates general genomic knowledge concerning genes, their components, their products, and their products' subclasses and components which can be exploited effectively in the *Drosophila* literature to compute the anaphoric link, whether coreferential or associative, between gene mentions and mentions of proteins, RNA and other gene products.

2. *Drosophila* Gene Name Recognition

In the biomedical domain, there is a paucity of annotated text and none which is focused entirely on the *Drosophila* literature. We extend recent approaches to bootstrapping systems for name recognition, partly by necessity, but also because annotation is expensive and does not constitute a viable long-term approach to the development of IE systems.

2.1. *Reproducing the Morgan et al. experiment*

FlyBase provides a dictionary of all *Drosophila* genes and their synonyms that appear in the extant curated literature together with links to the literature indicating where a specific name is used to refer to a particular gene. Morgan *et al.*[5] exploited this information to create annotated material to train a gene name recognizer. In brief, abstracts were tokenized and then genes names linked to specific abstracts in FlyBase were tagged applying longest-extent pattern matching. The process resulted in a large but noisy corpus which was in turn used to train a hidden Markov model (HMM) recognizer.

[a] www.flybase.net
[b] http://song.sourceforge.net/

We replicated this experiment with an enlarged dataset and different software. We built a list of all articles mentioned in the FlyBase bibliography for which the database also recorded at least one gene as having been mentioned within it. We then retrieved all the abstracts of those articles using the NCBI Entrez Programming Utilities[7]. This gave a total of 16609 abstracts (9.5% more than Morgan *et al.*). The abstracts were tokenized using the RASP toolkit[8c]. Then, following Morgan *et al.*, in each abstract we annotated all the gene name mentions licensed by the associated FlyBase gene name list.

The 16609 abstracts contained approximately 7800 distinct gene names representing 5243 distinct genes out of a total of over 44000K names recorded in FlyBase[d]. Many gene names and synonyms do not appear in the training material. As Morgan *et al.* note, there are gene synonyms that are common English words, such as *to* and *by*, resulting in precision errors in the training data. Sometimes genes mentioned in abstracts are not in the respective FlyBase gene lists of those articles (as only some relevant sections of the article are curated), resulting in recall errors.

The recognizer used in our experiments is the open source toolkit LingPipe[e]. The named entity recognition module is a 1st-order HMM model using Witten-Bell smoothing. For each token $t[n]$ and possible label $l[n]$, the following joint probability is computed, conditioned on the previous two tokens and the previous label:

$$P(t[n], l[n] | l[n-1], t[n-1], t[n-2]) \quad (1)$$

Unknown tokens are analyzed using a morphologically-based classifier, which we modified slightly to adapt it to the domain. The approach is highly lexical and conservative compared to others (e.g. Crim *et al.*[9]) which deploy more abstract and general features to achieve greater domain-independence. Lingpipe achieves high precision by only generalizing to unseen names in lexical contexts which are clearly indicative of gene names in the training data.

We tested the performance of the trained recognizer on the test data used in Morgan *et al.*[5]. The data consists of 86 abstracts doubly-annotated

[c]http://www.cogs.susx.ac.uk/lab/nlp/rasp/

[d]Exact figures depend on how much normalization (e.g. homogenizing punctuation, Greek letters, capitalization and whitespace) one applies to the names before counting them.

[e]http://www.alias-i.com/lingpipe/

by a biologist curator (Colosimo) and a computational linguist (Morgan). The performance of LingPipe on each annotation (Recall/Precision/F-score) was 0.8086/0.7485/0.7774 and 0.8423/0.8483/0.8453, respectively. To calculate these figures we used the evaluation script used for the BioNLP2004[f] shared task. Morgan *et al.*, evaluating on the the first set of annotations, reported 0.71/0.78/0.75. Our performance appears better, especially in terms of recall.

2.2. *NER Annotation guidelines*

There is a large difference in performance (0.0679 in F-score) between the two annotations of the dataset due to difficulties in applying the annotation scheme. According to the guidelines used in Morgan et al.[5], gene names are tagged not only when they refer to genes, but also when they are part of mentions of proteins or transcripts, as in *the zygotic Toll protein*. Only *Drosophila* genes are tagged, excluding reporter genes, genes that are not part of the natural *Drosophila* genome, families, particular alleles or protein complexes. However, *Drosophila* genes can be synonymous with foreign genes (e.g. *Hsp90*), family names are often synonymous with specific names (e.g. *CSP*), and foreign and reporter genes are often not flagged as such in text. Additionally, mutant genes, which are not part of the natural genome, are usually referred to using the name of the original gene, leading to inconsistencies in annotation in cases like *dunce mutations* or *eye PKCI700D mutant*.[g]

We developed revised guidelines, partially based on those developed for ACE[10]. We did not exclude foreign genes, reporter genes and families, as they are often of interest to curators and users of FlyBase. Like Morgan *et al.*, gene names (<*gn*>) are tagged not only when they refer to genes but also when they are found in pre-nominal modifier positions. Following ACE, we annotate the surrounding noun phrase (NP). The NP is tagged either as a *gene-mention* (<*gm*>) or as *other-mention* (<*om*>), depending on whether it refers to a gene or not (see 1) and 2) in Figure 1). In cases of alleles, mutants or protein complexes, the gene name is tagged and the remaining tokens of the NP are tagged *other-mention* (see 5) in Figure 1).

[f] http://research.nii.ac.jp/ collier/workshops/JNLPBA04st.htm

[g] Overall the biologist Colosimo's annotations are more accurate given the annotation guidelines used. He avoids tagging reporter genes synonymous with specific ones (e.g. *Gal4*), mutants, or gene families (e.g. *Hedgehog Hh*), resulting in fewer genes tagged (909) than by Morgan (989).

(1) <*gm*>the <*gn*>dunce</*gn*> gene</*gm*>
(2) <*om*>the <*gn*>dunce</*gn*> mutations</*om*>
(3) <*gm*>the human <*gn*>IL-2</*gn*> gene</*gm*>
(4) <*om*>the unrearranged <*gn*>TcR delta</*gn*> gene expression</*om*>
(5) <*om*><*gn*>eye</*gn*> PKCI700D mutant</*om*>

Figure 1. Annotated examples

As also reported by Dingare *et al.*[11], the data used in the BioNLP[3] and BioCreative[12] evaluations contained many cases in which modifiers of gene names and nouns modified by gene names were variably annotated. Using the guidelines suggested in this paper, the annotation of such cases becomes clearer. In 3) and 4) in Figure 1, the guidelines are applied to cases with inconsistent annotation reported by Dingare et al.[11]. 2) was inconsistently annotated by Colosimo and Morgan. Annotation of NPs is also relevant to recovery of anaphoric links (see § 3) and aids annotation of gene names within coordinated NPs.

LingPipe's performance using our guidelines to resolve differences between Morgan and Colosimo was Recall/Precision/F-score 0.8081/0.8493/0.8282. Further results below will be reported on both our re-annotation (called "merged") and the gold standard used in Morgan *et al.* (called "morgan").

2.3. *Inspecting errors and improving performance*

We tried to identify the main sources of errors and ameliorate them taking account of the specific HMM model utilized. Our first step was to perform an individual evaluation on seen and unseen tokens. This evaluation didn't take into account multi-token genes, because there were many cases where the one boundary of such multi-token cases was incorrect. Therefore, our system was not penalized for partially recognized genes and received/lost a point for each gene token recognized/missed. This token-wise definition of Recall/Precision is used only when reporting results on seen or unseen tokens. In all other cases, the standard definitions are used.

Evaluating on seen tokens on the "merged" dataset, we achieved 0.8272/0.9022/0.8631 Recall/Precision/F-score, which suggests that there are many gene names that are missed, even though they exist in the training data. For example, the gene *gurken* appears 97 times in the training data, of which 90 times it is tagged correctly as a gene on the basis of FlyBase

links. However, the few false negatives in the training data cause LingPipe to fail to tag *gurken* as a gene name during testing. We experimented with a non-conservative version of Morgan *et al.*'s procedure in which all gene names recorded in FlyBase were annotated as such in all of the training data. However, this resulted in many false positives in the training data and overall worse performance.

In general, gene names appearing in the abstracts are mentioned in FlyBase gene lists. So, we post-processed the training data by reannotating tokens as genes when this token was annotated as a gene in the overall training data more than a certain percentage of the time. By doing this though, we risk changing common English words correctly tagged as ordinary words to genes, since some genes have common English words as synonyms. With the percentage set at 80% we obtained Recall/Precision/F-scores of 0.8567/0.8551/0.8559 on the "merged" dataset and of 0.8614/0.7565/0.8056 on "morgan".

On unseen tokens, compared to Morgan *et al.* our performance is significantly higher (F-scores of 0.619 on "merged" and 0.5365 on "morgan" compared to their 0.33). However, LingPipe is rather conservative in classifying unseen tokens as genes (Recall/Precision was 0.4642/0.9285 on "merged").

As with the seen tokens, we tried to improve recall, as it is important for curation to have a system that is able to recognize unseen gene names. For each token classified, we estimated the entropy of the distribution of Equation 1 computed by LingPipe, which gave us an indication of how (un)certain the classifier was of its decision. We observed that many of the recall errors occurred in cases in which the HMM model classified a token with entropy close to 1, i.e. with high uncertainty. We post-processed the output of the classifier by re-annotating as genes unseen tokens that were classified as ordinary words with entropy higher than a specified threshold. As before, the lower this threshold was set, the higher the recall at the expense of precision. By setting this threshold at 0.6 and evaluating on the "merged" dataset, we improved the performance on unseen tokens to 0.7058 F-score. However, this resulted in more partially recognized genes, which slightly reduced the performance when evaluating using the standard definition of the metrics (from 0.8559 to 0.8545). Also, only 49 out of 16779 tokens in the test set were not seen in the training data. In order to demonstrate the value of this method, we performed an experiment using only 20% of the available training data, which resulted in 1040 unseen tokens in the test set. In this case, using the uncertainty of the classifier in the way described earlier, the performance on unseen tokens rose

from 0.4424 to 0.6111 in F-score, while the overall performance using the standard definition of F-score rose from 0.5847 to 0.6487, evaluating on "merged".

2.4. *Reference resolution*

Our recognizer identifies strings that are names of genes. Reference resolution requires determining the FlyBase identifier of a gene name. Frequently, *Drosophila* gene names are not unique identifiers. For ambiguity resolution, a quite effective and simple strategy (around 89% accuracy) is to associate names with the entities that those names most frequently denote using FlyBase's lists of gene names occurring in articles. For orthographic variants, FlyBase's gene synonym lists are a good resource for calculating commonly occurring types of variation (e.g. prefix by *D.* or *Dm.*) which can be applied to previously unseen name strings. Some exploration of variants of these strategies is undertaken in Ma[13].

3. Biomedical anaphora resolution

In FlyBase curation, the "gene" is an organizing concept around which other information is recorded. In order to extract all the information about a gene in a text it is necessary to identify all textual entities (like pronouns, definite descriptions and proper names) that are anaphorically linked to that gene or coreferential with it. These entities may refer to proteins, RNA, alleles, mutants and other gene "products" rather than the gene itself, and may therefore be associative rather than coreferential anaphoric links. Here we report work on resolving anaphoric definite descriptions (DDs; phrases beginning with the definite article *the* e.g. *the faf gene*) and proper nouns (PNs), since in biomedical texts there are fewer cases where pronouns are used.

The first step towards resolving anaphora is selecting the anaphoric expressions to be resolved and their possible antecedents. We first parse the text using RASP, then select all noun phrases (NPs) in the text and filter them to find the ones referring to relevant entities using information from the gene name recognizer and the Sequence Ontology (SO).

3.1. *Biotype information: semantic tagging*

If a NP is headed by a gene name according to the recognizer (i.e. its rightmost element is a gene name), then it refers to a gene. Otherwise, we

use information in the SO to search for the *biotype* of the head noun which stands in one of the four following possible relations to a gene: "part-of", "type-of", "subproduct" or "is-a".

SO relates entities by the following relations: derives_from, member_of, part_of, is_a, among others[h]. For instance, we extract the unique path of concepts and relations which leads from gene to protein, shown in Figure 2:

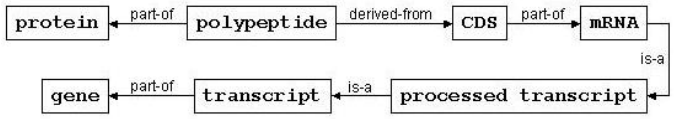

Figure 2. SO path from gene to protein.

Besides the facts directly expressed in this path, we also assume the following:

(1) whatever is-a transcript is also part-of a gene
(2) whatever is part-of a transcript is also part-of a gene
(3) mRNA is part-of a gene
(4) whatever is part-of a mRNA is also part-of a gene
(5) CDS is part-of a gene
(6) polypeptide is a sub-product (derived-from) of a gene
(7) whatever is part-of a polypeptide is also a sub-product of a gene
(8) protein is a sub-product of a gene

We then use these assumptions to add new derivable facts to the original path. For example, an *exon* is a part of a transcript according to SO, therefore, by the 2nd assumption, we add the fact that an *exon* is a part of a gene. We also extract information about gene types that is included in the ontology as an entry called "gene class". Using the derived information, we would tag *the third exon* with "part-of-gene". NPs that remain untagged after this search are tagged as "other-bio" if any head modifier is a gene name. These biotyped NPs are then considered for anaphora resolution.

[h]The member-of relation is considered a type of the part-of relation, so we do not make this distinction and consider both as part-of relations.

3.2. *Anaphora resolution*

Our baseline unsupervised system for anaphora resolution that we present here makes use of lexical, syntactic, semantic and positional information to identify the antecedent of an anaphoric expression. The lexical information consists of the words themselves. The syntactic information consists of NP detection and the distinction between head and premodifiers (extracted from RASP output). The distance (in words) between the anaphoric expression and its possible antecedent is taken into account as positional information. The semantic information comes from the gene recognizer and the SO-based tagging described above. Thus, the system is bootstrapped from a variety of extant resources without any domain-specific tuning.

As anaphoric expressions to be resolved we take all PNs and DDs among the filtered NPs. To link anaphoric expressions to their antecedents we look at three aspects of the corresponding NPs: the head noun, the premodifiers of the head noun, and the biotype.

The pseudo-code to find the antecedent for the DDs and PNs is given below:

- Input: a set A with all the anaphoric expressions (DDs and PNs); a set C with all the possible antecedents (all NPs with biotype information)
- For each anaphoric expression A_i:
  - Let antecedent 1 be the closest preceding NP C_j such that head(C_j)=head(A_i) and biotype(C_j)=biotype(A_i)
  - Let antecedent 2 be the closest preceding NP C_j such that head(C_j)$\neq$ head(A_i) and biotype(C_j)$\neq$ biotype(A_i), but head(C_j)=premodifier(A_i), or premodifier(C_j)=head(A_i), or premodifier(C_j)=premodifier(A_i)
  - Take the closest candidate as antecedent, if 1 and/or 2 are found; if none is found, the DD/PN is treated as non-anaphoric
- Output: The resolved anaphoric expressions in A are linked to their antecedents.

For example, in the passage "Dosage compensation, which ensures that the expression of *X-linked genes(C_j)* is equal in males and females ... the hypertranscription of *the X-chromosomal genes(A_j)* in males", the candidate C_j meets the conditions for antecedent 1 to be linked to the anaphoric expression A_j. In "... the role of *the roX genes(C_k)* in this process ... which

MSL proteins interact with *the roX RNAs(*A_k*)*", C_k meets the conditions for antecedent 2 to A_k.

3.3. *Experimental Results - Related Work*

We have annotated two articles from PubMed central containing 334 sentences and 7641 tokens in total. 334 anaphoric expressions (90 DDs and 244 PNs) with the relevant biotypes were found and their antecedents were manually annotated when they were functioning anaphorically. When we tested the anaphora resolution algorithm on this annotated data using the manually corrected syntactic and biotype information, the algorithm achieves Recall/Precision/F-score of 0.62/0.64/0.63. However, on the same text using automatic parsing and biotype tagging, performance drops to 0.37/0.43/0.40, primarily because of errors in identifying NPs and extracting their head nouns.

Most previous work on anaphora resolution in (biomedical) texts has used supervised machine learning techniques and different knowledge sources for biotype classification. For instance, Yang *et al.*[14] assigns biotypes using a named entity recognizer trained on the GENIA corpus together with other features as part of a supervised approach; Castano *et al.*[15] uses the UMLS (Unified Medical Language System)[i] to type DDs in MEDLINE abstracts and describes an unsupervised approach. The SO is more focussed on the functional genomics domain and therefore more appropriate for FlyBase curation.

4. Conclusions and Future Work

The two modules described are integrated into an interactive environment for FlyBase curators to help them in the task of literature curation. The environment allows navigation by anaphorically-linked entities and links the current paper with information derived from FlyBase and the SO.

The gene recognizer achieves state-of-the-art performance via bootstrapping but may be further improved by training on full articles with a greater variety of lexical contexts and by the use of additional feature types. Anaphora resolution requires improvement. We plan to use the baseline system to generate noisy training data for a statistical anaphora resolution module. Both components will be incrementally improved using

[i]http://www.nlm.nih.gov/research/umls/

active training with curators correcting a small number of highlighted low confidence cases in each presented article.

Acknowledgments

This work is part of the BBSRC-funded FlySlip[j] project. We would like to thank Alexander Morgan for making the annotated test data available to us and for advice on replication of the experiment reported in Morgan *et al.*[5], Chihiro Yamada for his expert help with annotation of *Drosophila* articles, and Bob Carpenter for help with LingPipe. Caroline Gasperin is funded by a CAPES award from the Brazilian government.

References

1. L. Hirschman, J. C. Park, J. Tsujii, L. Wong, and C. H. Wu. Accomplishments and challenges in literature data mining for biology. *Bioinformatics*, 18(12):1553–1561, 2002.
2. H. Liu and C. Friedman. Mining terminological knowledge in large biomedical corpora. In *Pacific Symposium on Biocomputing*, pages 415–426, 2003.
3. J. Kim, T. Ohta, Y. Tsuruoka, Y. Tateisi, and N. Collier, editors. *Proceedings of JNLPBA, Geneva, Switzerland*, August 28–29 2004.
4. R. Gaizauskas, G. Demetriou, P. J. Artymiuk, and P. Willet. Protein structures and information extraction from biological texts: The "PASTA" system. *BioInformatics*, 19(1):135–143, 2003.
5. A. A. Morgan, L. Hirschman, M. Colosimo, A. S. Yeh, and J. B. Colombe. Gene name identification and normalization using a model organism database. *J. of Biomedical Informatics*, 37(6):396–410, 2004.
6. Karen Eilbeck and Suzanna E. Lewis. Sequence ontology annotation guide. *Comparative and Functional Genomics*, 5:642–647, 2004.
7. Eric Sayers and David Wheeler. *Building Customized Data Pipelines Using the Entrez Programming Utilities (eUtils)*. NCBI.
8. E. J. Briscoe and J. Carroll. Robust accurate statistical annotation of general text. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation*, pages 1499–1504, 2002.
9. J. Crim, R. McDonald, and F. Pereira. Automatically annotating documents with normalized gene lists, 2004.
10. Annotation guidelines for entity detection and tracking (EDT).
11. S. Dingare, J. Finkel, M. Nissim, C. Manning, and C. Grover. A system for identifying named entities in biomedical text: How results from two evaluations reflect on both the system and the evaluations. In *The 2004 BioLink meeting at ISMB*, 2004.

[j] http://www.cl.cam.ac.uk/users/av308/Project_Index/Project_Index.html

12. Christian Blaschke, Lynette Hirschman, and Alexander Yeh, editors. *Proceedings of the BioCreative Workshop*, Granada, March 2004.
13. Ning Ma. Using author trails to disambiguate entity references. Master's thesis, University of Cambridge, Computer Laboratory, 2005.
14. X. Yang, J. Su, G. Zhou, and C. L. Tan. An NP-cluster based approach to coreference resolution. Geneva, Switzerland, August 2004.
15. J. Castano, J. Zhang, and J. Pustejovsky. Anaphora resolution in biomedical literature. In *Proceedings of the International Symposium on Reference Resolution for NLP*, 2002.