One model per entity: using hundreds of machine learning models to recognize and normalize biomedical names in text

Victor Bellon
MINES ParisTech,
PSL-Research University,
CBIO-Centre de bio-informatique
victor.bellon@mines-paristech.fr

Raul Rodriguez-Esteban
Roche Innovation Center Basel
raul.rodriguez-esteban@roche.com

Abstract

We explored a new approach to named entity recognition based on hundreds of machine learning models, each trained to distinguish a single entity, and showed its application to gene name identification (GNI). The rationale for our approach, which we named "one model per entity" (OMPE), was that increasing the number of models would make the learning task easier for each individual model. Our training strategy leveraged freely-available database annotations instead of manually-annotated corpora. While its performance in our proof-of-concept was disappointing, we believe that there is enough room for improvement that such approaches could reach competitive performance while eliminating the cost of creating costly training corpora.

1 Background

Recognizing names in text is a longstanding task in natural language processing (NLP) known as named-entity recognition (NER). In biomedical text mining (or BioNLP), the focus of NER is on certain technical names (terms) such as those of chemical compounds, genes, species and anatomical parts. Recognizing such names alone, however, is of limited application as, in practice, they often need to be linked to other facts. This can be done by first mapping them to unique name identifiers—a task known as normalization or grounding. Recognizing and normalizing names of genes and gene products, in particular, has drawn much attention from the BioNLP community (Leser and Hakenberg, 2005). These names are usually considered a single class of terms due to their overlapping vocabularies (Hatzivassiloglou et al., 2001). Thus, here we refer to them as simply gene names.

The tasks of gene name recognition (GNR) and gene name normalization (GNN) involve, respectively, the recognition and normalization of gene names found in text. Gene name identification (GNI) is the combination of gene name recognition and normalization (GNR + GNN) (see the framework by Krauthammer and Nenadic (2004)). State-of-the-art GNI methods involve machine learning algorithms, such as conditional random fields (CRF), trained under supervised learning. Supervised learning requires gold-standard training and testing sets, which for GNI typically are sets of documents (corpora) that have been manually annotated for gene names by expert curators. Several community challenges have been organized to foster the improvement of GNI algorithms (Morgan et al., 2008; Lu et al., 2011). However, despite such efforts, even the best algorithms suffer from an important weakness. Namely that their performance has been shown to degrade outside of their training and testing corpora, decaying to levels barely above those of rule-based systems involving dictionary-matching rules together with filtering of noisy names (Rebholz-Schuhmann et al., 2013; Rodriguez-Esteban, 2016b,a).

It has been suggested that the shortcomings of current GNI machine learning algorithms could be addressed in two ways: (1) by training models with different, diverse corpora (Rebholz-Schuhmann et al., 2013) (see, in that respect, the work of Kaewphan et al. (2016) with cell line names), and (2) by using "domain adaptation" techniques, which consist in adapting machine learning models to the characteristics of the input text. The limited size and number of existing gold-standard corpora, and the cost of creating new ones, represent, however, a bottleneck for (1). For (2), experiments with domain adaptation in biomedical text have led, thus far, only to modest improvements in performance (Miwa et al., 2012).

Here we describe an alternative approach that

Proceedings of the Biomedical NLP Workshop associated with RANLP 2017, pages 49–54,
Varna, Bulgaria, 8 September 2017.
https://doi.org/10.26615/978-954-452-044-1_007

can be applied to problems that require the identification of large but finite sets of entities, particularly in biomedicine. To begin with, instead of using a gold-standard corpus as training set, we propose utilizing the wealth of manual annotations that currently exist in biomedical databases. Indeed, several freely-available databases provide a growing number of annotations concerning gene name identifiers associated to biomedical documents. The main drawback of these annotations is that they are weakly labeled, as they do not specify the precise location in which the genes are mentioned within the documents. However, there are ways to infer these locations (Jain et al., 2016).

While past GNI studies have not leveraged annotations from biomedical databases, there are examples of their use for GNN (Wermter et al., 2009; Zwick, 2015; Chen et al., 2015). In these studies contextual features were created out of the annotated biomedical documents to resolve ambiguous gene mentions. Besides for GNN, weakly-labeled database annotations have been used in BioNLP for identifying protein-specific residues (Ravikumar et al., 2012) and annotating Medline abstracts with Gene Ontology terms (Gobeill et al., 2013). In another example, Furrer et al. (2014) used a biomedical database called BioGRID (Chatr-Aryamontri et al., 2015) for the purpose of training and testing an algorithm for extracting protein-protein interactions (PPI).

Leveraging database annotations for GNI is not straightforward. We have implemented our approach in a way that, as far as we can tell, has not been described in the NER literature before (biomedical or otherwise). Our method involves training many machine learning models, each model trained to identify a single entity (i.e. a single gene) rather than, as it is commonplace, training one or a handful of models to identify all entities. We call this approach "one model per entity" (OMPE).

2 Methods

As building block for our OMPE system we used BANNER (Leaman and Gonzalez, 2008), which is a machine learning algorithm for NER built on CRF. While BANNER is based on a generic model that can be trained to identify any class of terms, it has shown state-of-the-art performance in GNR (Kabiljo et al., 2009). Our strategy consists in using multiple BANNER models, each model being responsible for detecting the mentions of a single gene. That means that a gene name mention that is recognized by a BANNER model can be automatically mapped to the gene for which the model was trained.

2.1 Training set

To create our training set we built first a database of positive training examples containing sentences that mention gene names. Each sentence in the database was associated to a gene identifier (NCBI Gene ID), corresponding to a gene mentioned in the sentence, and to a document identifier (PubMed ID), corresponding to the document source of the sentence. The {NCBI Gene ID, PubMed ID} pairs came from the following publicly-available databases: gene2pubmed, UniProt, BioGRID (Chatr-Aryamontri et al., 2015) and Gene Reference into Function (GeneRIF) (see Table 1).

Source	Genes	Documents	Mentions
gene2pubmed	34 004	493 620	1 087 465
UniProt	21 383	22 539	68 966
BioGRID	11 832	23 925	66 358
GeneRIF	17 462	386 927	641 354

Table 1: Statistics of the different datasets used.

Because these databases do not specify the location of the gene mentions in the source documents, we retrieved each source document from the Medline baseline 2015 and attempted to find their locations. In order to do that we leveraged gene names and synonyms from the NCBI Gene database. This database, however, does not include all the gene name variations and synonyms that authors use in practice (Hirschman et al., 2002; Liu et al., 2006). To increase recall we therefore expanded the list of gene names and synonyms following Schuemie et al. (2007). By using this expanded list to look up gene names in the source documents we created a set of positive examples for each of the genes annotated in the aforementioned databases.

For training (and testing) we only considered genes for which we had at least 32 positive examples (this cut-off was a compromise between coverage and amount of training data available), which totaled 2180 with a median of 281 positive examples per gene. These genes covered approximately 80% of all gene mentions appearing in the test corpus.

Negative training examples were selected according to different strategies. First, we created certain modified versions of the positive examples. Modifications consisted in the deletion of words within the gene names that made reference to a certain function, such as *receptor*, *inhibitor*, *enhancer*. For example, while *TNF-α receptor* refers to gene ID *7132*, *TNF-α* corresponds to gene ID *7124*.

The second type of negative examples that we selected consisted in positive examples belonging to genes that share synonyms. For example, the gene name *FAT* may refer to gene ID *2195* or *948*. Thus, positive examples for gene *948* can be used as negative examples for gene *2195*. Finally, we included as negative examples randomly selected sentences from the English Wikipedia (not from any particular domain) and positive examples from randomly selected genes.

2.2 Test set

For testing the performance of our OMPE system we used a modified dataset based on the gold standard from the BioCreative 2 Gene Normalization (BC2GN) challenge (Morgan et al., 2008). The BC2GN training set covers 281 abstracts and 684 gene annotations, and the testing set covers 262 abstracts and 785 gene annotations. As we used an independent training set based on freely-available database annotations we could employ both BC2GN training and testing datasets to create our BC2GN$_{mod}$ test dataset.

When building the BC2GN$_{mod}$ dataset we only considered gene annotations for 621 unique genes from the 1156 genes present in the original BC2GN datasets—those for which we had more than 32 positive examples in our training database. Thus, BC2GN$_{mod}$ contained a total of 841 human gene annotations.

We compared our OMPE system against GNAT (Hakenberg et al., 2008, 2011), which is a state-of-the-art system for GNI (Rebholz-Schuhmann et al., 2013). We evaluated the prediction quality of our system according to the number of true positives (TP), false positives (FP) and false negatives (FN), and according to precision (P), recall (R) and F-measure (F).

2.3 Computation

We made use of two different computational configurations for training and testing. First, we used a server with 40 CPU cores at 2.4 GHz and 567 GB RAM. This server was used for both generating the training set and making the final predictions. As training the models is the most computationally demanding task, we used a cluster computer with 164 nodes, each node possessing 2 CPUs with 12 cores (Intel Xeon Processor E5-2680 v3) and 256 GB of memory. In this configuration the median model training time was 212 seconds.

3 Results

Two versions of the OMPE system were tested and compared against the output of GNAT. The first version (OMPE1) used the standard BANNER implementation, in which the most probable class is associated to every token. The second version (OMPE2) used a modified BANNER that required the probability of a token being a mention to be larger than a certain threshold, which we set to 0.95.

Results for predictions over the BC2GN$_{mod}$ dataset can be seen in Table 2. GNAT showed a high performance, with a recall of 0.762 and a precision of 0.881, corresponding to 892 TPs and only 121 FPs. The OMPE1 system achieved, on the other hand, a recall of 0.331 and a precision of 0.215 caused by the large number of FPs, 1413.

Method	**TP**	**FP**	**FN**	**P**	**R**	**F**
GNAT	892	121	278	.881	.762	.817
OMPE1	387	1413	783	.215	.331	.261
OMPE2	355	575	815	.382	.303	.338

Table 2: Performance of the 3 different methods.

To reduce the number of FPs we set a threshold to the probability of accepting a prediction (OMPE2). By using the threshold we dramatically reduced the number of FPs from 1413 to 575, increasing the precision to 0.382 while slightly decreasing the recall to 0.303.

In Figure 1 we show a comparison of each method's individual performance. In this figure, the first row compares OMPE1 and GNAT, while the second row compares OMPE2 and GNAT. The last row compares OMPE1 and OMPE2. Light colors represent the individual performance of the methods and dark colors the difference between them. The first and second column show the precision and recall, respectively. Genes were ordered according to performance differences between methods.

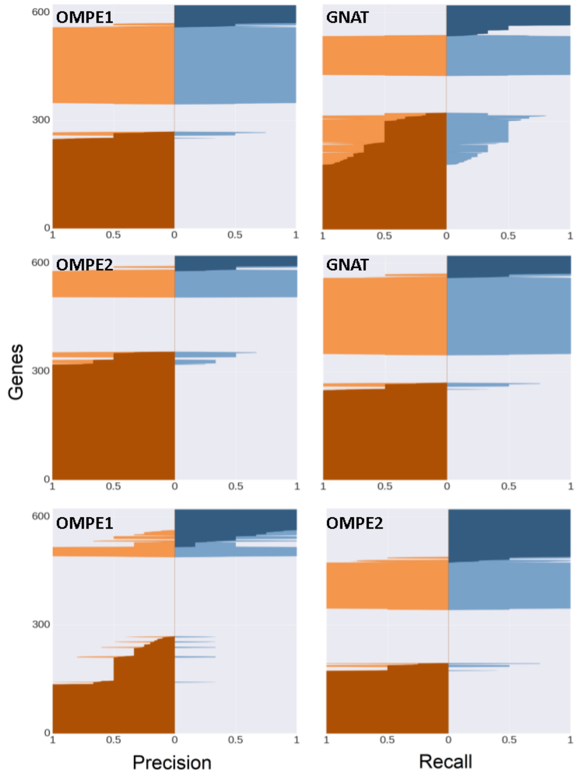

Figure 1: Difference in precision and recall of the different methods on an individual gene basis. The first row compares OMPE1 and GNAT, while the second row compares OMPE2 and GNAT. The last row compares OMPE1 and OMPE2. The first and second column show the precision and recall, respectively. Light colors represent the individual performance of the methods and dark colors the difference between them. Genes were ordered according to performance differences between methods.

Figure 1 shows that there is a set of genes in which one of the algorithms works well but the other algorithms do not. Moreover, the use of a threshold in OMPE2 leads to an increase in the precision over a large number of genes and to a decrease in only a small number of them. Figure 2, on the other hand, shows the cumulative frequency distribution of precision and recall for genes predicted in the test datasets with the different methods.

4 Discussion

An advantage of the OMPE approach is that it allows targeted performance improvements with respect to specific gene names. Positive examples and synonyms belonging to particularly challenging gene names can be modified interactively without the need for retraining the entire system (all models in our case), unlike in interactive single-model approaches such as *tagtog* (Cejuela et al., 2014). Another advantage of OMPE is its robustness, as it is not trained on a particular hand-selected corpus. Thus, our results with the BC2GN$_{mod}$ corpus are not biased by the training set utilized.

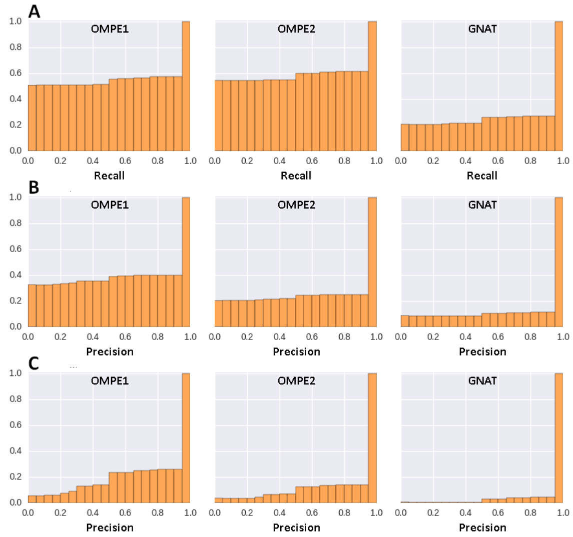

Figure 2: Cumulative frequency distribution of genes at each precision and recall level. (A) Recall for prediction of genes in the test dataset. (B) Precision for prediction of genes in the test dataset. (C) Precision for algorithms focused on the genes known to be present in the test dataset.

A challenge for training the OMPE system is the selection of negative examples. It is important to select negative examples that are as similar as possible to the positive examples, meaning examples that are closest to the class separation boundary—analogous to what support vectors represent for support vector machines (SVMs). One of our approach's limitations is its reduced recall due to the low number of positive examples that exist for many genes. Such genes are, on the other hand, less likely to be mentioned in the biomedical literature and, as biomedical databases continue to grow, the number of positive examples for those genes will keep increasing as well.

Finally, our focus was only on human genes. The identification of genes from additional species would have required greater computational resources. An OMPE system that covered all protein-expressing genes would need to be trained for around 20 000 genes (Ezkurdia et al., 2014).

In this sense, and in the reliance on large, growing biomedical databases, our approach has a futuristic stance, meaning that it will become more feasible with time. As “Big Computing” infrastructure, such as cloud computing, becomes increasingly available and more powerful, it will become more practical to implement systems such as OMPE. It is important to stress that computational requirements differ greatly between training an OMPE system and deploying it for prediction, which requires far lower computational power.

Beyond the GNI example shown here, OMPE can be used to identify other types of entities with a finite cardinality, such as (in the BioNLP field) diseases, cell types, cell lines and anatomical parts. We have focused here on GNI because it has been already widely investigated and has multiple applications, such as the tracking of biomedical facts and trends (Cokol and Rodriguez-Esteban, 2008; Cokol et al., 2007; Rodriguez-Esteban and Loging, 2013). Beyond NER, the OMPE approach could also be applied to other classification problems in which the class cardinality is below a computationally-feasible threshold. The rationale would again be that increasing the number of models could ease (”relieve”) the learning task to each individual model.

5 Conclusion

In this study we have shown a new approach for GNI that takes advantage of the decreasing costs of computing and the increasing availability of annotated data to train hundreds of machine learning models. Our proof of concept did not reach acceptable performance levels but, due to the fact that there remains ample room for potential improvements, such strategies could become competitive for GNI and other domains in the future. Following the remarks from Halevy et al. (2009) in “The unreasonable effectiveness of data,” we should learn to “use available large-scale data rather than hoping for annotated data that isn’t available.”

References

JM Cejuela, P McQuilton, L Ponting, SJ Marygold, R Stefancsik, GH Millburn, B Rost, and FlyBase Consortium. 2014. tagtog: interactive and text-mining-assisted annotation of gene mentions in PLOS full-text articles. *Database (Oxford)* 0(bau033). https://doi.org/10.1093/database/bau033.

A Chatr-Aryamontri, B J Breitkreutz, R Oughtred, L Boucher, S Heinicke, D Chen, C Stark, A Breitkreutz, N Kolas, L O’Donnell, T Reguly, J Nixon, L Ramage, A Winter, A Sellam, C Chang, J Hirschman, C Theesfeld, J Rust, M S Livstone, K Dolinski, and M Tyers. 2015. The BioGRID interaction database: 2015 update. *Nucleic Acids Res* 43(Database issue):D470–478. https://doi.org/10.1093/nar/gku1204.

G Chen, J Zhao, T Cohen, C Tao, J Sun, H Xu, E V Bernstam, A Lawson, J Zeng, A M Johnson, V Holla, A M Bailey, H Lara-Guerra, B Litzenburger, F Meric-Bernstam, and W Jim Zheng. 2015. Using ontology fingerprints to disambiguate gene name entities in the biomedical literature. *Database (Oxford)* 2015:bav034. https://doi.org/10.1093/database/bav034.

M Cokol and R Rodriguez-Esteban. 2008. Visualizing evolution and impact of biomedical fields. *J Biomed Inform* 41(6):1050–1052. https://doi.org/10.1016/j.jbi.2008.05.002.

M Cokol, R Rodriguez-Esteban, and A Rzhetsky. 2007. A recipe for high impact. *Genome Biol* 8(5):406. https://doi.org/10.1186/gb-2007-8-5-406.

I Ezkurdia, D Juan, J Rodriguez, A Frankish, M Diekhans, J Harrow, J Vazquez, A Valencia, and M Tress. 2014. Multiple evidence strands suggest that there may be as few as 19 000 human protein-coding genes. *Hum Mol Genet* 23(22):5866–5878. https://doi.org/10.1093/hmg/ddu309.

L Furrer, S Clematide, H Marques, R Rodriguez-Esteban, M Romacker, and F Rinaldi. 2014. Collection-wide extraction of protein-protein interactions. *6th International Symposium on Semantic Mining in Biomedicine* pages 61–66. https://doi.org/10.5167/uzh-101472.

J Gobeill, E Pasche, D Vishnyakova, and P Ruch. 2013. Managing the data deluge: data-driven go category assignment improves while complexity of functional annotation increases. *Database (Oxford)* 2013:bat041. https://doi.org/10.1093/database/bat041.

J Hakenberg, M Gerner, M Haeussler, I Solt, C Plake, M Schroeder, G Gonzalez, G Nenadic, and C M Bergman. 2011. The GNAT library for local and remote gene mention normalization. *Bioinformatics* 27:2769–2771. https://doi.org/10.1093/bioinformatics/btr455.

J Hakenberg, C Plake, R Leaman, M Schroeder, and G Gonzalez. 2008. Inter-species normalization of gene mentions with GNAT. *Bioinformatics* 24:126–132. https://doi.org/10.1093/bioinformatics/btn299.

A Halevy, P Norvig, and F Pereira. 2009. The unreasonable effectiveness of data. *IEEE Intelligent Systems* 24(2):8–12. https://doi.org/10.1109/MIS.2009.36.

V Hatzivassiloglou, P A Dubou, and A Rzhetsky. 2001. Disambiguating proteins, genes, and rna in text: a machine learning approach. *Bioinformatics* 17 Suppl 1:S97–106. https://doi.org/10.1093/bioinformatics/17.suppl_1.S97.

L Hirschman, AA Morgan, and AS Yeh. 2002. Rutabaga by any other name: extracting biological names. *J Biomed Inform* 35(4):247–259. https://doi.org/10.1016/S1532-0464(03)00014-5.

S Jain, K R, T T Kuo, S Bhargava, G Lin, and C N Hsu. 2016. Weakly supervised learning of biomedical information extraction from curated data. *BMC Bioinformatics* 17 Suppl 1:1. https://doi.org/10.1186/s12859-015-0844-1.

R Kabiljo, A B Clegg, and A Shepherd. 2009. A realistic assessment of methods for extracting gene/protein interactions from free text. *BMC Bioinformatics* 10:233. https://doi.org/10.1186/1471-2105-10-233.

S Kaewphan, S Van Landeghem, T Ohta, Y Van de Peer, F Ginter, and S Pyysalo. 2016. Cell line name recognition in support of the identification of synthetic lethality in cancer from text. *Bioinformatics* 32(2):276–282. https://doi.org/10.1093/bioinformatics/btv570.

M Krauthammer and G Nenadic. 2004. Term identification in the biomedical literature. *J Biomed Inform* 37(6):512–526. https://doi.org/10.1016/j.jbi.2004.08.004.

R Leaman and G Gonzalez. 2008. BANNER: an executable survey of advances in biomedical named entity recognition. *Pac Symp Biocomput* pages 652–663. https://doi.org/10.1142/9789812776136_0062.

U Leser and J Hakenberg. 2005. What makes a gene name? Named entity recognition in the biomedical literature. *Brief Bioinform* 6(4):357–369. https://doi.org/10.1093/bib/6.4.357.

H Liu, ZZ Hu, M Torii, C Wu, and C Friedman. 2006. Quantitative assessment of dictionary-based protein named entity tagging. *J Am Med Inform Assoc* 13(5):497–507. https://doi.org/10.1197/jamia.M2085.

Z Lu, H Y Kao, C H Wei, M Huang, J Liu, C J Kuo, C N Hsu, R T Tsai, H J Dai, N Okazaki, H C Cho, M Gerner, I Solt, S Agarwal, F Liu, D Vishnyakova, P Ruch, M Romacker, F Rinaldi, S Bhattacharya, P Srinivasan, H Liu, M Torii, S Matos, D Campos, K Verspoor, K M Livingston, and W J Wilbur. 2011. The gene normalization task in BioCreative III. *BMC Bioinformatics* 12 Suppl 8:S2. https://doi.org/10.1186/1471-2105-12-S8-S2.

M Miwa, P Thompson, and S Ananiadou. 2012. Boosting automatic event extraction from the literature using domain adaptation and coreference resolution. *Bioinformatics* 28:1759–1765. https://doi.org/10.1093/bioinformatics/bts237.

A A Morgan, Z Lu, X Wang, A M Cohen, J Fluck, P Ruch, A Divoli, K Fundel, R Leaman, J Hakenberg, C Sun, H H Liu, R Torres, M Krauthammer, W W Lau, H Liu, C N Hsu, M Schuemie, K B Cohen, and L Hirschman. 2008. Overview of BioCreative II gene normalization. *Genome Biol* 9 Suppl 2:S3. https://doi.org/10.1186/gb-2008-9-s2-s3.

K Ravikumar, H Liu, J D Cohn, M E Wall, and K Verspoor. 2012. Literature mining of protein-residue associations with graph rules learned through distant supervision. *J Biomed Semantics* 3 Suppl 3:S2. https://doi.org/10.1186/2041-1480-3-S3-S2.

D Rebholz-Schuhmann, S Kafkas, J H Kim, C Li, A Jimeno Yepes, R Hoehndorf, R Backofen, and I Lewin. 2013. Evaluating gold standard corpora against gene/protein tagging solutions and lexical resources. *J Biomed Semantics* 4:28. https://doi.org/10.1186/2041-1480-4-28.

R Rodriguez-Esteban. 2016a. Additional knowledge-based analysis approaches. In W Loging, editor, *Bioinformatics and Computational Biology in Drug Discovery and Development*, Cambridge University Press, Cambridge, United Kingdom. https://doi.org/10.1017/CBO9780511989421.011.

R Rodriguez-Esteban. 2016b. Understanding human disease knowledge through text mining: What is text mining? In W Loging, editor, *Bioinformatics and Computational Biology in Drug Discovery and Development*, Cambridge University Press, Cambridge, United Kingdom. https://doi.org/10.1017/cbo9780511989421.004.

R Rodriguez-Esteban and W T Loging. 2013. Quantifying the complexity of medical research. *Bioinformatics* 29:2918–2924. https://doi.org/10.1093/bioinformatics/btt505.

M J Schuemie, B Mons, M Weeber, and J A Kors. 2007. Evaluation of techniques for increasing recall in a dictionary approach to gene and protein name identification. *J Biomed Inform* 40:316–324. https://doi.org/10.1016/j.jbi.2006.09.002.

J Wermter, K Tomanek, and U Hahn. 2009. High-performance gene name normalization with GeNo. *Bioinformatics* 25:815–821. https://doi.org/10.1093/bioinformatics/btp071.

M Zwick. 2015. Automated curation of gene name normalization results using the Konstanz information miner. *J Biomed Inform* 53:58–64. https://doi.org/10.1016/j.jbi.2014.08.016.