Semantic Type Classification of Common Words in Biomedical Noun Phrases

Amy Siu
Max Planck Institute for Informatics
66123 Saarbrücken, Germany
siu@mpi-inf.mpg.de

Gerhard Weikum
Max Planck Institute for Informatics
66123 Saarbrücken, Germany
weikum@mpi-inf.mpg.de

Abstract

Complex noun phrases are pervasive in biomedical texts, but are largely underexplored in entity discovery and information extraction. Such expressions often contain a mix of highly specific names (diseases, drugs, etc.) and common words such as "condition", "degree", "process", etc. These words can have different semantic types depending on their context in noun phrases. In this paper, we address the task of classifying these common words onto fine-grained semantic types: for instance, "condition" can be typed as "symptom and finding" or "configuration and setting". For information extraction tasks, it is crucial to consider common nouns only when they really carry biomedical meaning; hence the classifier must also detect the negative case when nouns are merely used in a generic, uninformative sense. Our solution harnesses a small number of labeled seeds and employs label propagation, a semisupervised learning method on graphs. Experiments on 50 frequent nouns show that our method computes semantic labels with a micro-averaged accuracy of 91.34%.

1 Introduction

1.1 Motivation

In biomedical texts, entities are written as natural language expressions – often complex noun phrases. Previous works on information extraction in this domain have focused on short phrases that work well, for instance, with dictionary-based approaches. The most notable method is the MetaMap tool by Aronson and Lang (2010). Often, however, expressions are long and complex, mixing domain-specific names (of diseases, symptoms, drugs, etc.) with common nouns such as "condition", "degree" or "process". Examples for such complex phrases are:
1) monitoring of the carcinogenic process
2) development of processes for the prognosis of malaria.

In the first example, "process" is a vital part of the phrase and carries biomedical meaning, namely, denoting a body function. In the second example, "process" is used in the generic sense of the common noun and is relatively uninformative for the purpose of detecting biomedical entities in text. For information extraction tasks like entity discovery, relation mining and knowledge base population, it is crucial to distinguish these two situations. Moreover, in the first case, we would like to further annotate the common noun with a semantic type that captures the role of the word within the surrounding noun phrase.

This kind of semantic typing could be based on WordNet senses (Fellbaum, 1998), using techniques for word sense disambiguation (Navigli, 2009), or on UMLS entries. However, WordNet has limited coverage of the biomedical domain, and UMLS has rather coarse-grained and sometimes fuzzy types. Therefore, we devised a small collection of *fine-grained semantic types* ourselves. The novelty of our proposed semantic types lies in the explicit provision for non-biomedical types, as well as the uninformative type where applicable; Table 1 shows both of these elements in play for the target words *culture* and *degree*.

Our goal then is to automatically label common words in complex noun phrases with the most appropriate semantic type or inferring that the word is merely used in a generic sense without specific biomedical meaning. We focus on a judiciously chosen list of common nouns, referred to as *target words*, that frequently appear within long noun phrases in biomedical texts. The resulting annota-

Proceedings of the 2015 Workshop on Biomedical Natural Language Processing (BioNLP 2015), pages 98–103,
Beijing, China, July 30, 2015. ©2015 Association for Computational Linguistics

Target word	Semantic types
culture	medical sample social construct
degree	metric for temperature metric for bending stage in progression (e.g. second degree burn) academic degree degree of freedom in statistics edges out of a node in a graph generic, uninformative

Table 1: Semantic types for the target words *culture* and *degree*.

tions – for example, labeling "process" in "monitoring of the carcinogenic process" as body function – can in turn enhance the coverage and quality of information extraction tasks.

1.2 Approach and contribution

We develop a semisupervised method for labeling a target word, within a given noun phrase, with its most suitable semantic type or tagging it as biomedically unspecific and uninformative. Our method is based on label propagation over a graph that connects noun phrases and has a small number of manually labeled seed nodes. Each distinct noun phrase becomes a node, and an edge connects two nodes that share a target word with a weight reflecting the similarity between the contexts of the respective phrase occurrences. We then apply the MAD label propagation algorithm (Talukdar and Crammer, 2009) to infer the best type labels for the target words in the graph's nodes.

Experiments show that our method achieves 91.34% micro-averaged and 83.57% macro-averaged accuracy over 50 frequently appearing target words. Moreover, our method is capable of classifying both target words with and without an uninformative semantic type.

2 Related work

In general, the semantic interpretation of complex phrases is a long-studied problem in computational linguistics, and widely viewed as a very demanding task (see, e.g., Sag et al. (2002); Nakov and Hearst (2013)). For biomedical texts, however, complex phrases are an infrequently studied problem. Golik et al. (2013) propose to handcraft rules based on linguistic cues to identify longer noun phrases beyond dictionary entries. Similar to this paper, they are also motivated by the needs of a knowledge acquisition application. Their work makes a point in analyzing "semantically poor" terms, some of which essentially entail the uninformative semantic type we propose.

The problem setting closest to word usage detection is undoubtedly word sense disambiguation (WSD) of free text. For the general domain, the vast body of work has been surveyed by Navigli (2009), and mature software tools such as It Makes Sense (Zhong and Ng, 2010) covers most words. For the biomedical domain, the majority of previous works center around two WSD datasets (Weeber et al., 2001; Jimeno-Yepes et al., 2011) that together contain 253 ambiguous words, multi-word terms, and abbreviations. In addition, Stevenson et al. (2008), Fan et al. (2009), and Cheng et al. (2012) propose methods to generate labeled data. As for methodologies, vector space models (McInnes, 2008; Savova et al., 2008) are a common choice. Another common approach is to exploit the rich knowledge embedded in UMLS. Agirre et al. (2010) and Humphrey et al. (2006) leverage entity-entity relations and semantic type information in UMLS, respectively.

Entity disambiguation is another highly relevant research area. For the general domain, most efforts focus on named entities, and software systems such as AIDA (Hoffart et al., 2011) and Wikifier (Ratinov et al., 2011) are both robust and scalable. In contrast, for the biomedical domain, existing works target restricted scopes such as species (Wang et al., 2010) and acronyms (Harmston et al., 2012). Although MetaMap (Aronson and Lang, 2010) covers all the diverse entities in UMLS, its entity disambiguation functionality remains limited.

3 Methodology

3.1 Outline of methodology

Our method operates on one target word at a time. We collect noun phrases in our text corpus that contain the selected target word. On the one hand comes the manual preparation of the target semantic types and their seed phrases. On the other hand comes the automatic computation of similarities of noun phrase pairs. This similarity is based on *context* – a window of k words before and after the target word in a noun phrase (for clarity purposes, we denote by *context words* those words in the window surrounding the target). This context, in turn, is captured by three features, namely word occurrences, part-of-speech tags, and entity types (again for clarity purposes, we distinguish *context entity types* that are precomputed, from target semantic types that we want to classify). Using the seed phrases and context similarities, we cast the the noun phrases into a graph and apply the MAD label propagation algorithm.

In the following subsections, we describe how we construct each component.

3.2 Target semantic types

In our corpus, we observe that 90% of all noun occurrences come from 5000+ distinct nouns. Since it is infeasible to study so many of them, we pick 50 highly common but semantically ambiguous ones to be our target words. For each target word, we manually specify its applicable target semantic types based on two criteria. First, a target semantic type should have a discernible presence in the corpus. Second, the contexts of target semantic types should be amenable to a learning algorithm, i.e. they should be sufficiently distinct from each other. Recall that we would also like to identify the case when the target word is used in a generic, uninformative way. We facilitate this by adding a uninformative semantic type. We observe, however, that not all target words require this uninformative type. For instance, *culture* has two overwhelmingly dominant types (medical sample and social construct) such that the rest are negligible and do not need an explicit representation. This specification of target semantic types is based on manual observation, over both the corpus noun phrases and UMLS entries relevant to the target word.

Once the semantic types are set, we nominate a few representative phrases as seed phrases. This process is again manual, where we aim for phrases which are sufficiently prevalent, and which convey the target semantic type with high certainty. Table 2 shows all semantic types and all the seed phrases for the target word *activity*, and the complete list is available at `http://mpi-inf.mpg.de/~siu/bionlp2015/`. In our compilation, one target word has on average 3.78 target semantic types, which in turn has on average 2.68 seed phrases.

3.3 Context entity type estimation

We would like to assign an entity type to each context word. However, since a comprehensive entity disambiguation tool is not available, we estimate the entity types by a popularity-based approach that exploits the repetitiveness of thesauri entries and semantic assets in UMLS. First, take note of UMLS entity names that contain a single word. Next, for each distinct entity name, take note of the entities (distinct CUIs), as well as the number of occurrences (MRCONSO entries) represented. A few heuristics determine which entity is the most popular, and the corresponding CUI's UMLS semantic type[1] becomes the word's entity type. Taking *cat* as an example, it appears 16 times as a mammal, 3 times as the abbreviation for CAT scan, and 1 time as an enzyme. Therefore *cat*'s entity type is *Mammal*, the UMLS semantic type for CUI 0007450. In essence, this approach approximates the entity type with the largest prior distribution probability. Since biomedical word senses are often highly skewed (Jimeno-Yepes et al., 2011), we believe this approach is a reasonable interim substitute to a full-fledged entity disambiguation tool.

In addition to the 133 UMLS semantic types, we introduce an extra type to represent measurement units such as mg/kg and μmol.

We investigate two variants of entity type similarity. Under the hard variant, only the same entity type occurrences contribute towards context similarity (e.g. *Cell* and *Cell Component* would therefore be considered completely dissimilar). Under the soft variant, similar entity types also contribute (*Cell* and *Cell Component* now have a similarity of 0.9375). The similarity between two entity types A and B is:

$$0.5 \times group(A, B) + 0.5 \times lch(A, B)$$

where $group()$ returns 1 if A and B belong to the same UMLS semantic group, and 0 otherwise. $lch(A, B)$ is the similarity score between A, B in the UMLS semantic type hierarchy according to Leacock and Chodorow's method (1998), normalized to range between 0 and 1. The use of $group()$ is necessary because some semantic type pairs are highly similar but far apart in the hierarchy (e.g. *Body System* and *Tissue*).

3.4 Context similarity

We model the similarity between two phrases by calculating a similarity score between their contexts. Specifically, the similarity score is a linear combination of the contributions from the contexts' words, part-of-speech (POS) tags, and entity

[1] Not to be confused with the custom target semantic types in Section 3.2. They are used independently in this work.

Semantic type	**Seed phrases**	**Sample classified noun phrases**
physical activity	fetal activity physical activity	instruction in self-directed exercises and activity diaries day-to-day household activities that create the backbone of healthy environments
body & protein process	catalytic activity disease activity inflammatory activity kinase activity	histochemically demonstrable esterase activity in the hypothalamus of the developing rat lower insulin-stimulated GS activity in PCOS patients compared with controls plasma anti-pneumococcal polysaccharide antibody activity (serotypes 3, 6a and 23) polymerase activity relative to the wild-type protein
generic, uninformative	of activity of of activity in	dual activity of exploring karanjin isolation for medicinal purposes the orchestration of a set of activities that should be executed in order to deliver an output

Table 2: Semantic types, seed phrases, and sample classified noun phrases for the target word *activity*.

types (either the hard or the soft variant):

$$sim(\text{context}_1, \text{context}_2) = \alpha_1 \times J_w(\text{words}_1, \text{words}_2) + \alpha_2 \times J_w(\text{POS tags}_1, \text{POS tags}_2) + \alpha_3 \times J_w(\text{entity types}_1, \text{entity types}_2)$$

where $\alpha_1 + \alpha_2 + \alpha_3 = 1$, and $J_w()$ is the weighted Jaccard similarity function. Intuitively, $J_w()$ captures not only the overlap between two sets of items, but also the significance or weight of the items. In our setting, an item is a word, a POS tag, or a context entity type, and the weight depends on the item's distance to the target word – the smaller the distance, the higher the weight. Based on preliminary experiments, $1/d$ is found to be the best weighting scheme, where d is the distance between target and context words. For word, POS, and the hard variant of context entity type, only exact matches count towards $J_w()$ item overlap (singular/plural and American/British spellings of the same word qualify as exact matches). For the soft variant of context entity type, the $1/d$ weight is further scaled by the entity type-entity type similarity score.

3.5 MAD label propagation

Now we have all the ingredients to build a graph out of a collection of noun phrases. Take a phrase as a node. Compute the similarity score between two phrases' contexts, and make it the weight of the edge between the two corresponding nodes. A small number of nodes containing seed phrases become the seed nodes, and the seed phrase's semantic type is the label. Apply the MAD label propagation algorithm (Talukdar and Crammer, 2009) to label all the nodes, effectively classifying each node with the best target semantic type. Recall that each target word requires its own graph and hence separate application of MAD.

Label propagation, also known as belief propagation, is a semisupervised, iterative learning method on graphs. Some nodes, i.e. the seed nodes, in the graph are initially labeled. Informally, over the iterations, the seed nodes exert influence on their neighbors, whom in turn influence their neighbors, such that eventually all nodes become labeled. MAD is a state-of-the-art variant of the standard label propagation algorithm (Baluja et al., 2009), and it guarantees convergence. Based on preliminary experiments, $\mu_1 = 10 \times \mu_2 = 100 \times \mu_3$ were found to be the best hyperparameters for MAD. Since a graph with n nodes contains $O(n^2)$ edges, we prune low-weight edges to avoid excessively time consuming computations.

4 Results and discussion

4.1 The dataset

Our corpus consists of documents from a diverse set of biomedical free texts: PubMed abstracts and full-length articles, encyclopedic webpages from health portals, and online discussion forums. As a pre-processing step, each document is segmented into sentences by the LingPipe tool, and further tagged with POS and parsed into dependency graphs by the Stanford CoreNLP tool. We then extract the longest compound noun phrases from the sentences. Finally, for each target word, we make one collection by randomly selecting noun phrases containing that word. The average noun phrase length across collections are relatively uniform from 13 to 17 words.

4.2 Results

We tuned the method's parameters using a development dataset of 1,000 randomly selected nodes for each target word. Keeping the proportion of seed nodes at 5%, we obtained the best parameter setting (the α's in context similarity and window size k) for each individual word.

In the test dataset, each target word has a graph of 10,000 random nodes with also 5% seeds. On

Target word	#Types	Micro	Macro	Best context
activity	3	0.91	0.91	WPH
administration	2	0.93	0.84	WPS
area	6	0.92	0.89	WP
body	4	0.96	0.94	WPH
case	5	0.83	0.88	WPS
concentration	4	0.95	0.98	WPH
condition	2	0.95	0.96	WPH
control	4	0.98	0.97	WPS
culture	2	0.99	0.79	WP
degree	7	0.76	0.72	WP
development	5	0.88	0.86	WP
distribution	2	0.96	0.96	WPS
effect	2	0.93	0.75	WPS
expression	4	0.96	0.81	WPH
factor	6	0.96	0.72	WP
flow	5	0.83	0.90	WPH
form	4	0.92	0.63	WPS
function	3	0.94	0.94	WPS
group	3	0.92	0.74	WPS
information	4	0.95	0.95	WPH
line	5	0.89	0.85	WPS
measure	2	0.90	0.80	WPS
mechanism	2	0.85	0.76	WPS
model	3	0.96	0.63	WPS
pattern	6	0.77	0.81	WP
period	3	0.91	0.92	WPS
point	8	0.92	0.76	WP
pressure	6	0.79	0.89	WP
problem	4	0.89	0.67	WP
process	4	0.85	0.91	WPH
product	6	0.95	0.91	WP
profile	3	0.98	0.84	WP
program	5	0.92	0.85	WPH
rate	3	0.95	0.78	WP
reaction	5	0.97	0.94	WP
reduction	3	0.72	0.75	WPS
region	4	0.90	0.50	WPS
report	2	0.99	0.97	WPH
resistance	3	0.98	0.66	WPS
response	5	0.89	0.73	WPS
result	4	0.91	0.89	WPH
role	3	0.98	0.99	WPH
sequence	2	0.97	0.95	WPS
set	2	0.98	0.97	WPS
site	4	0.96	0.85	WPH
solution	2	0.99	0.94	WPS
state	4	0.98	0.82	WP
strain	3	0.66	0.59	WPS
system	4	0.92	0.85	WPS
technique	2	0.91	0.92	WPS

Table 3: Number of semantic types, micro- and macro-averaged accuracy, and the best context setting of 50 target words. W, P, H, S denote word, POS, hard and soft context entity types, respectively.

average, 1428 and 437 nodes were evaluated for each target word and for each target semantic type, respectively. Two annotators evaluated the labels suggested by the MAD algorithm, and the value of Fleiss' Kappa was 0.76, which indicates substantial inter-annotator agreement. Table 3 lists the micro- and macro-averaged accuracy, as well as the best context setting.

4.3 Discussion

Overall, micro-averaged accuracy is very encouraging at 80% or above for 45 target words. A few target words (*degree*, *pattern*, and *pressure*) have higher numbers (6 or 7) of target semantic types. As the number of target semantic types increases for one target word, it becomes harder for the types' contexts to be sufficiently distinct from each other. This phenomenon leads to noisy edge weights in the graph, which in turn leads to poorer classification results. Other target words (*reduction* and *strain*) also have week micro-averaged accuracy despite having fewer (3) target semantic types. In both cases here, the dominant target semantic type is used in such a broad way that a few seed phrases are not sufficient to describe the context. Specifically, a reduction of quantity can be about just anything; and an organism strain can be described at the population, experiment, organism, gene, or molecular level, or can be described via the characteristic effect the strain causes.

Macro-averaged accuracy performs less consistently and varies across target words. The overriding contributing factor here is the skew of the target semantic types' distribution. In our annotations, the most frequent label of one target word constitutes from 23% to 91% of occurrences. When a sparse type is represented by few labeled examples in the graph, naturally there is less generalization power to classify correctly.

In terms of how much context words, POS, and context entity types contribute towards the solution, we are surprised that the use of words and POS alone are sufficient for 28% of the target words to achieve the best experimental setting. While the rest of the target words benefit to varying degrees the hard and soft variants of context entity types, it is worth noting that even a rudimentary estimation of context entity types empowers better context comparisons for the other 72% of target words.

Errors in the classification stem from two main sources. In some cases, the critical cue, be it a word or a context entity type, lies outside of the context window. In other cases, significant expert knowledge is needed to put the puzzle together.

5 Conclusion

In this work, we present a semisupervised method that classifies a word's semantic type in complex noun phrases. With 50 common words, we demonstrate that a small number of labeled seeds can enable a label propagation algorithm to assign both conventional semantic type labels as well as the negative case of uninformative label. We envision that the semantic types of words in a noun phrase make one building block towards more fully utilizing that phrase. In the future, we plan to apply our method to other information extraction modules, and enrich their capability in handling longer phrases that go beyond dictionary entries.

References

Eneko Agirre, Aitor Soroa, and Mark Stevenson. 2010. Graph-based word sense disambiguation of biomedical documents. *Bioinformatics*, 26(22): 2889–2896.

Alan R. Aronson and François-Michel Lang. 2010. An overview of MetaMap: historical perspective and recent advances. *Journal of the American Medical Informatics Association*, 17(3): 229–236.

Shumeet Baluja, Rohan Seth, D. Sivakumar, Yushi Jing, Jay Yagnik, Shankar Kumar, Deepak Ravichandran, and Mohamed Aly. 2008. Video suggestion and discovery for YouTube: taking random walks through the view graph. *Proceedings of WWW*, pp. 895–904.

Weiwei Cheng, Judita Preiss, and Mark Stevenson. 2012. Scaling up WSD with automatically generated examples. *Proceedings of BioNLP*, pp. 231–239.

Jung-Wei Fan and Carol Friedman. 2009. Generating quality word sense disambiguation test sets based on MeSH indexing. *Proceedings of the AMIA Symposium*, pp. 183–187.

Christiane Fellbaum. 1998. WordNet: an electronic lexical database. *MIT Press*.

Wiktoria Golik, Robert Bossy, Zorana Ratkovic, and Claire Nédellec. 2013. Improving term extraction with linguistic analysis in the biomedical domain. *Proceedings of CICLing*, pp. 24–30.

Nathan Harmston, Wendy Filsell, and Michael Stumpf. 2012. Which species is it? Species-driven gene name disambiguation using random walks over a mixture of adjacency matrices. *Bioinformatics*, 28(2): 254–260.

Johannes Hoffart, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. 2011. Robust disambiguation of named entities in text. *Proceedings of EMNLP*, pp. 782–792.

Susanne M. Humphrey, Willie J. Rogers, Halil Kilicoglu, Dina Demner-Fushman, and Thomas C. Rindflesch. 2006. Word sense disambiguation by selecting the best semantic type based on Journal Descriptor Indexing: preliminary experiment. *Journal of the American Society for Information Science and Technology*, 57(1): 96–113.

Antonio J. Jimeno-Yepes, Bridget T. McInnes, and Alan R. Aronson. 2011. Exploiting MeSH indexing in MEDLINE to generate a data set for word sense disambiguation. *BMC Bioinformatics*, 12:223.

Claudia Leacock and Martin Chodorow. 1998. Combining local context and WordNet similarity for word sense identification. *WordNet: an electronic lexical database*, 49(2): 265–283.

Bridget T. McInnes. 2008. An unsupervised vector approach to biomedical term disambiguation: integrating UMLS and Medline. *Proceedings of ACL-HLT-SRWS*, pp. 49–54.

Preslav Nakov and Marti A. Hearst. 2013. Semantic interpretation of noun compounds using verbal and other paraphrases. *ACM Transactions on Speech and Language Processing*, 10(3): 13:1–13:51.

Roberto Navigli. 2009. Word sense disambiguation: a survey. *ACM Computing Surveys*, 41(2): 10:1–10:69.

Lev Ratinov, Dan Roth, Doug Downey, and Mike Anderson. 2011. Local and global algorithms for disambiguation to Wikipedia. *Proceedings of ACL-HLT*, pp. 1375–1384.

Ivan A. Sag, Timothy Baldwin, Francis Bond, Ann A. Copestake, and Dan Flickinger. 2002. Word sense disambiguation across two domains: biomedical literature and clinical notes. *Journal of Biomedical Informatics*, 41(6): 1088–1100.

Guergana K. Savova, Anni R. Coden, Igor L. Sominsky, Rie Johnson, Philip V. Ogren, Piet C. de Groen, and Christopher G. Chute. 2008. Multiword expressions: a pain in the neck for NLP. *Proceedings of CICLING*, pp.1–15.

Mark Stevenson, Yikun Guo, and Robert Gaizauskas. 2008. Acquiring sense tagged examples using relevance feedback. *Proceedings of COLING*, pp. 809–816.

Partha P. Talukdar and Koby Crammer. 2009. New regularized algorithms for transductive learning. *Proceedings of ECML PKDD*, part II, pp. 442–457.

Xinglong Wang, Junichi Tsujii and Sophia Ananiadou. 2010. Disambiguating the species of biomedical named entities using natural language parsers. *Bioinformatics*, 26(5): 661–667.

Marc Weeber, James G. Mork, and Alan R. Aronson. 2001. Developing a test collection for biomedical word sense disambiguation. *Proceedings of the AMIA Symposium*, pp. 746–750.

Zhi Zhong and Hwee Tou Ng. 2010. It Makes Sense: a wide-coverage word sense disambiguation system for free text. *Proceedings of ACL*, pp. 78–83.