

Towards Converting Clinical Phrases into SNOMED CT Expressions

Rohit J. Kate
University of Wisconsin-Milwaukee
katerj@uwm.edu

Abstract

Converting information contained in natural language clinical text into computer-amenable structured representations can automate many clinical applications. As a step towards that goal, we present a method which could help in converting novel clinical phrases into new expressions in SNOMED CT, a standard clinical terminology. Since expressions in SNOMED CT are written in terms of their relations with other SNOMED CT concepts, we formulate the important task of identifying relations between clinical phrases and SNOMED CT concepts. We present a machine learning approach for this task and using the dataset of existing SNOMED CT relations we show that it performs well.

1 Introduction

Many clinical applications, including clinical decision support, medical error detection, answering clinical queries, generating patient statistics, and biosurveillance, would be automated if the clinical information locked in natural language clinical text could be converted into computer-amenable structured representations. To enable this, a long-term goal is to convert entire natural language clinical documents into structured representations. As an important step in that direction, in this paper we focus on a task that can help in converting clinical phrases into a structured representation. SNOMED CT (IHTSDO, 2009) is a standardized representation for clinical concepts whose extensiveness and expressivity makes it suitable for precisely encoding clinical phrases. A concept in SNOMED CT is defined in terms of its relations with other concepts and it currently includes around four hundred thousand pre-defined clinical concepts. If a natural language clinical phrase represents a concept which is already present in SNOMED CT then the conversion process reduces to a matching function; some previous work (Stenzhorn et al., 2009; Barrett et al., 2012) as well as existing SNOMED CT browsers, like CliniClue,¹ can automatically perform such a matching. But our focus in this paper is instead on the task of creating new SNOMED CT concepts for clinical phrases for which no SNOMED CT concept already exists.

Since new concepts in SNOMED CT can be created by identifying their relations with existing SNOMED CT concepts, we formulate the important task of identifying relations between clinical phrases and SNOMED CT concepts. That is, given a clinical phrase (for example, “acute gastric ulcer with perforation”) and a description of a SNOMED CT concept (for example, “stomach structure”), whether a particular kind of relation (for example, “finding site”) is present between them or not (in this example it is present). To the best of our knowledge, there is no other work which has attempted this type of relation identification task. Note that this task is very different from the relation extraction task (Zelenko et al., 2003). In that task, two entities are given in a sentence and the system determines whether the two entities are related or not mostly based on what the sentence says. In contrast, there is no sentence in this task and the presence of a relation is determined entirely based on the two entities.

Since several thousand relations already exist in SNOMED CT, we used these existing relations to form our dataset. Both training and test relation example pairs were obtained from this dataset. To identify each kind of relation we separately trained a machine learning method. We employed SVM

¹<http://www.cliniclue.com>

(Cristianini and Shawe-Taylor, 2000) machine learning method in combination with a new kernel that we specifically designed for this relation identification task. The experimental results show that the trained system obtains a good accuracy.

Such a system could be used for creating precise SNOMED CT expressions for clinical phrases. For example, “acute gastric ulcer with perforation” could be represented as an “acute gastric ulcer”, whose finding site is “stomach structure” and whose associated morphologies are “perforated ulcer” and “acute ulcer” (this is also shown in Table 1 under phrase (c)). In this example, “is a”, “finding site” and “associated morphology” are the identified relations, and “acute gastric ulcer”, “stomach structure”, “perforated ulcer” and “acute ulcer” are already present concepts in SNOMED CT. This representation would be obtained by efficiently testing the phrase for all the relations and with all the existing SNOMED CT concepts.

2 Background and Related Work

Realizing the importance of unlocking the clinical information present in free-text clinical reports, researchers started working on automatically converting them into structured representations years ago. Previous systems to convert natural language clinical information into structured representation, like LSP (Sager et al., 1987, 1995), MedLEE (Friedman et al., 1994, 2004) and Menelas (Spyns et al., 1992; Spyns and Willems, 1995), were manually built by linguistically and medically trained experts over a long course of time. The builders manually encoded how different natural language patterns should convert into the target structured representations. They also developed their own suitable structured representations (Friedman et al., 1993; Zweigenbaum et al., 1995) which restricts their systems from being useful elsewhere where a different type of structured representation is in use. Although we are limiting ourselves to clinical phrases instead of full sentences at this stage, we use machine learning techniques to minimize the manual cost of building such a system. For the structured representation, we are using the standardized clinical terminology, SNOMED CT, which is already widely in use.

Systematized Nomenclature of Medicine - Clinical Terms (SNOMED CT; IHTSDO, 2009) is the most comprehensive clinical terminology in the world today and is widely used in electronic health record systems for documentation purposes and reporting (Benson, 2010). Its extensive content and expressivity makes it suitable for precisely encoding clinical concepts. Not only does it specify around four hundred thousand pre-defined medical concepts and relations between them, but also its compositional grammar (IHTSDO, 2008) can be used to build new expressions that represent new medical concepts in terms of the existing concepts. SNOMED CT has been developed in the description logic formalism (Baader et al., 2003) which also makes it suitable for automated reasoning. For all these reasons, we think that it is the best structured representation into which natural language clinical phrases may be converted. There are browsers and tools available that can help users search SNOMED CT as well as interactively build new expressions, like CliniClue. Lee et al. (2010) presented a method for manually encoding text with SNOMED CT. There also has been recent work in automatically mapping text to SNOMED CT pre-defined concepts (Jung et al., 2009; Stenzhorn et al., 2009; Barrett et al., 2012) or UMLS pre-defined concepts (Aronson and Lang, 2010). However, these systems at best do an approximate match from clinical phrases to pre-defined concepts, also known as *pre-coordinated expressions*. In contrast, the system presented in this paper can help to automatically map natural language clinical phrases which do not match any pre-defined concepts into their semantically equivalent new SNOMED CT expressions. The new SNOMED CT expressions are also known as *post-coordinated expressions*. We did a preliminary analysis of the i2b2 2010 clinical text corpus (Uzuner et al., 2011) and found that out of around 8300 unique annotated concepts (noun phrases) in it, only around 1600 were pre-defined concepts in SNOMED CT. This shows that new phrases are abundantly present in clinical text and hence the ability to convert them into new SNOMED CT expressions is important.

Natural Language Clinical Phrase	SNOMED CT Expression
(a) severe pain in the stomach	116680003 is a = 22253000 pain {363698007 finding site = 69695003 stomach structure , 272141005 severity = 24484000 severe }
(b) neoplasm of right lower lobe of lung	116680003 is a = 126713003 neoplasm of lung {116676008 associated morphology = 108369006 neoplasm , 363698007 finding site = 266005 structure of right lower lobe of lung }
(c) acute gastric ulcer with perforation	116680003 is a = 95529005 acute gastric ulcer {363698007 finding site = 69695003 stomach structure , 116676008 associated morphology = 26317001 acute ulcer , 116676008 associated morphology = 91182001 perforated ulcer }
(d) family history of aplastic anemia	116680003 is a 243796009 situation with explicit context {246090004 associated finding = 306058006 aplastic anemia {408732007 subject relationship context = 303071001 person in the family } }

Table 1: Some examples of natural language clinical phrases and their corresponding SNOMED CT expressions. The numbers are the SNOMED CT concept and relation identifiers and their natural language descriptions are shown for human readability. The “=” character indicates relation kind on its left side and the related concept on its right side.

3 Identifying SNOMED CT Relations

3.1 Formulation of the Task

Table 1 shows some examples of clinical phrases and their associated SNOMED CT expressions. The expressions are shown using the syntax of SNOMED CT’s compositional grammar (IHTSDO, 2008). The numbers are the unique SNOMED CT concept and relation identifiers. Each concept in SNOMED CT has at least one natural language description. A description for each concept and relation is shown within vertical bars for human readability. The “=” character denotes relation kind on its left side and the related concept on its right side. The “is a” relation identifies the basic concept a clinical phrase represents and this is then further qualified using more relations which are shown in “{ }” brackets. Note that there could be multiple relations of the same kind in an expression, for example, in phrase (c) the “associated morphology” relation occurs twice. Similarly, even the “is a” relation can occur more than once because SNOMED CT allows multiple inheritance of concepts. There is more than one way to write an expression in SNOMED CT, ranging from close-to-user form to normal form (IHTSDO, 2009). We have shown close-to-user forms in the Table 1 which are simpler and easier for humans to understand. For the record, the concepts for phrases (b) and (c) are already present in the current version of SNOMED CT but the concepts for phrases (a) and (d) are not present.

As it can be observed, relations are the basis for forming SNOMED CT expressions. Hence, in this paper, we formulate the task of identifying relations between clinical phrases and SNOMED CT concepts. A new SNOMED CT expression could then be formed for a new clinical phrase by identifying its relations with exiting concepts. We present a machine learning method for training a separate relation identifier for each of the relations present in SNOMED CT (for example, “is a”, “finding site” etc.). Since every concept in SNOMED CT has a basic type (for example, “substance”, “disorder”, “body structure” etc.), and the basic type can also be determined for every clinical phrase (either directly from the context it is used in or by using a trained classifier),² we treat each relation with different types separately. For

²Alternatively, the type of a clinical phrase could also be identified by first determining the “is.a” relation.

example, the “finding site” relation that relates “disorder” to “body structure” is treated separately from the “finding site” relation that relates “finding” to “body structure”. The first column of Table 2 shows the most frequent relations in SNOMED CT along with their types which we used in our experiments.

Since several hundred thousand concepts and the relations between them are already present in SNOMED CT, we decided to use them as our dataset for training and testing our method. Every concept in SNOMED CT has a unique identifier and is also given a unique fully specified natural language name. In addition, it may have several natural language descriptions which are essentially a few different ways of expressing the same concept. To create our dataset, for every kind of relation, we randomly took some pairs of related concepts as positive examples and some pairs of unrelated concepts as negative examples. For each of the two concepts in a relation example, we randomly selected one description (phrase) out of all the descriptions it may have (including its fully specified name). We did so because a clinical phrase may not always be a fully specified name and the method should also be trained to work with alternate descriptions. Then the task of relation identification is: given the two descriptions of two concepts of particular types, determine whether they are related by a particular relation or not. We are not aware of any other work that has considered such a relation identification task for SNOMED CT.

3.2 Machine Learning Approach for the Task

For every kind of relation along with its types, we built a separate relation identifier. It may be noted that sometimes a presence of a relation can be identified simply by detecting overlap between the words in the two descriptions. For example, for the phrase (a) in Table 2, the word “pain” overlaps, hence “severe pain in the stomach” is a “pain”. Similarly for the phrase (b), “neoplasm of right lower lobe of lung” is a “neoplasm of lung”. However, this is not the case for many other relations. For example, the phrase (c) does not contain “stomach structure” which is its “finding site”. Hence besides mere overlap, the relation identifier system should be able to use several other clues. In the previous example, it should know that “gastric” generally means related to “stomach structure”. As it will be a formidable task to manually encode every piece of such knowledge, we use machine learning approach so that the system would automatically learn this kind of knowledge from training examples.

Another kind of knowledge a relation identifier would need is what words in the clinical phrase indicate what relations. For example, the word “in” in a disorder concept would usually indicate a “finding site” relation to a “body structure”. The machine learning system is expected to also learn this kind of knowledge from training examples. In our experiments, we used a baseline for comparison that uses only the amount of overlap for identifying relations.

We decided to use Support Vector Machine (SVM; Cristianini and Shawe-Taylor, 2000) as our learning algorithm because it has been shown to work well with thousands of features and hence has been widely used in natural language processing tasks which often involve use of several thousand features, for example, words and their combinations. An additional advantage of SVM is that one can implicitly specify potentially infinite number of features without actually enumerating them by defining a similarity function between the examples, called a *kernel*. This is also known as the *kernel trick*. For our relation identification task, we designed a specific kernel to enable the SVM learner to learn the kinds of knowledge it needs to learn which were mentioned earlier. It also incorporates word overlap which is sometimes a good indication of a relation. The kernel is defined as follows. Let A and B be two examples. Let c_1^A and c_2^A be the descriptions of the first and the second concepts of the example A respectively. Similarly, let c_1^B and c_2^B be the descriptions of the first and the second concepts of the example B respectively. Then the kernel $K(,)$ between examples A and B is defined as:

$$K(A, B) = sim(c_1^A, c_1^B) * sim(c_2^A, c_2^B) + sim(c_1^A, c_2^A) * sim(c_1^B, c_2^B) \quad (1)$$

where, $sim(,)$ is the similarity function between any two descriptions (which are phrases). In our experiments, we defined similarity as the number of common words. We also tried defining it as the number of common word subsequences (Lodhi et al., 2002; Bunescu and Mooney, 2005), but it did not result in any gain in the performance. Note that the above is a well-defined kernel because products

and summations of kernels are also well-defined kernels (Haussler, 1999). The kernel is normalized by dividing it by the square-root of $K(A, A) * K(B, B)$.

We now explain this kernel and what its implicit features are. The first term of the addition is a product of the number of common words between the first concepts of the two examples and the second concepts of the two examples. This essentially counts the number of common word pairs present in the two examples such that in each example the first word is present in the first concept and the second word is present in the second concept. For example, if both examples have “gastric” present in the first concept and “stomach” present in the second concept, then it will count “gastric,stomach” as a feature present in both the examples. Thus this kernel term implicitly captures pairs of words, one in each concept, as features. Based on these features, the learner may learn what combination of word pairs indicate a relation.

The second term simply treats the number of words overlapping between the two concepts of an example as a feature. The product then indicates how similar the two examples are along this feature. As was indicated earlier, overlap is an important feature because often the descriptions of the related concepts have overlap of words. In general, the two addition terms could be weighed differently, however, presently we did not experiment with different weights and we simply let SVM learn appropriate weights as part of its learning process.

4 Experiments

In this section we describe our experiments on the relation identification task for SNOMED CT relations.

4.1 Methodology

As was noted earlier, we formed our dataset utilizing the existing relations present in SNOMED CT. There are hundreds of different kinds relations present in SNOMED CT, some of them are more important than others (examples of some of the unimportant relations are “duplicate concept” and “inactive concept”). We report our results on the fourteen important and most frequent relations, each of which had more than ten thousand instances. The “is_a(procedure,procedure)” relation had the highest number of 93,925 instances. Since we had enough examples to choose our training and test examples from, instead of doing standard cross-validation, we ran five folds and in each fold we randomly selected five thousand training and five thousand test examples. Training beyond five thousand examples would lead to memory problems, but as our learning curves showed, the learning would generally converge by five thousand training examples.

For each relation, positive examples for both training and testing were randomly selected without replacement as pairs of concepts for which the relation is known to exist. Then equal number of negative examples were randomly selected without replacement as pairs of concepts of the required types which are not related by that relation. There was no overlap between training and testing datasets. We employed SVM using the LibSVM package³ along with the user-defined kernel as defined in the previous section.

We measured *precision* and *recall*. Precision is the percentage of correctly identified relations out of all the identified relations. Recall is the percentage of correctly identified relations out of all the relations present in the test set. The evaluation was done for the combined output of all the folds. SVM can also give the confidences for its classification decisions through Platt’s method (Platt, 1999). We used these confidences to measure precision and recall at every confidence level and plotted precision-recall curves. We also measured the maximum *F-measure* across the precision-recall curves, where F-measure is the harmonic mean of precision and recall.

We compared our approach with a baseline method which only uses the amount of word overlap between the two concepts to identify a relation between them. It is not a learning-based approach. It outputs its confidence on a relation as the degree of overlap between the two concepts (i.e. the number of common words after normalization for word lengths). We call this baseline the *similarity baseline*. Note

³<http://www.csie.ntu.edu.tw/~cjlin/libsvm/>

that the similarity scores are already included as features in the kernel used in the learning approach.⁴ Since there were equal number of positive and negative examples, the accuracy of a random classifier would be 50%.

4.2 Results and Discussion

Relation	Similarity Baseline (%)	Trained System (%)
associated_morphology(disorder, morphologic_abnormality)	66.67	83.29
causative_agent(disorder, substance)	82.94	90.87
finding_site(disorder, body_structure)	66.67	84.35
finding_site(finding, body_structure)	66.67	89.11
has_active_ingredient(product, substance)	85.92	91.34
is_a(body_structure, body_structure)	79.82	89.55
is_a(disorder, disorder)	78.18	84.55
is_a(finding, finding)	78.98	87.06
is_a(organism, organism)	66.67	79.23
is_a(procedure, procedure)	73.49	86.58
is_a(product, product)	67.48	86.27
is_a(substance, substance)	66.67	68.73
part_of(body_structure, body_structure)	66.67	89.13
procedure_site_direct(procedure, body_structure)	66.67	87.47

Table 2: Maximum F-measures over the precision-recall curves obtained by the similarity baseline and by the trained system for the most frequent SNOMED CT relations.

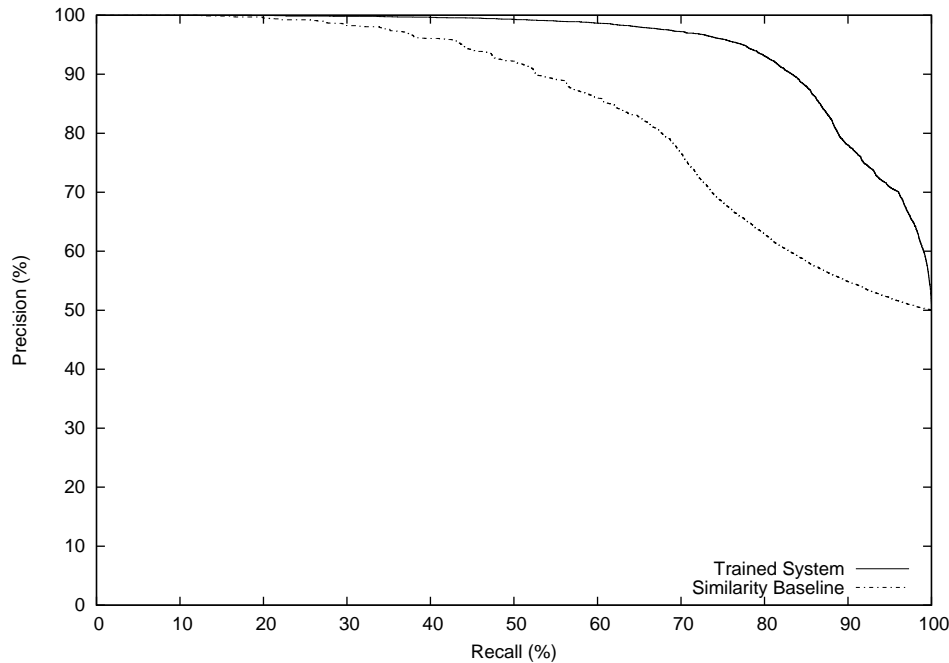


Figure 1: Precision-recall curves for the “is_a(procedure, procedure)” relation obtained using the similarity baseline and using the trained system.

Table 2 shows the maximum F-measures obtained across the precision-recall curves for the similarity

⁴We found that using the similarity score as the only feature in the learning method does not do well.

baseline and for the trained system for the fourteen most frequent relations in SNOMED CT. It may be first noted that the baseline does well on a few of the relations, obtaining close to 80% or more on five relations. This shows that the similarity baseline is not a trivial baseline although on some other relations it does not do well at all. Note that 66.67% F-measure can be also obtained by a random classifier by calling every relation as positive which would result in 50% precision and 100% recall. The learned approach does substantially better than the baseline on every relation. On eight of the fourteen relations it exceeds the baseline's performance by more than 10% (absolute). On five other relations it exceeds by more than 6%. The performance is close to 90% on five relations and is close to 80% or more on thirteen relations. The only relation on which the performance is not high is the "is_a(substance,substance)" relation. We found that this is mostly because a lot of new names are used for substances and from the names themselves it is not easy to identify that a particular substance is a type of another substance, for example, "lacto-n-tetrasylceramide, type 2 chain" is a "blood group antigen precursor".

Figure 1 shows the entire precision-recall curves obtained using the trained system and the similarity baseline for the "is_a(procedure, procedure)" relation. We are not showing these graphs for other relations due to space limitations, but this graph shows the typical curves obtained by the two methods. It may be noted looking at the lower part of the recall side that there are examples on which the relation can be identified with high precision even by the similarity baseline. But the learned approach continues to obtain high precision even on the high recall side when the precision of the baseline drops off. Finally, Figure 2 shows the learning curves for the same relation for the maximum F-measures on the precision-recall curves. Since the baseline method is not a learning method, its learning curve is horizontal. It can be seen that the learning method has almost converged and more training examples are unlikely to improve the performance substantially. It may also be noted that even with a few hundred training examples, the trained system already does much better than the baseline.

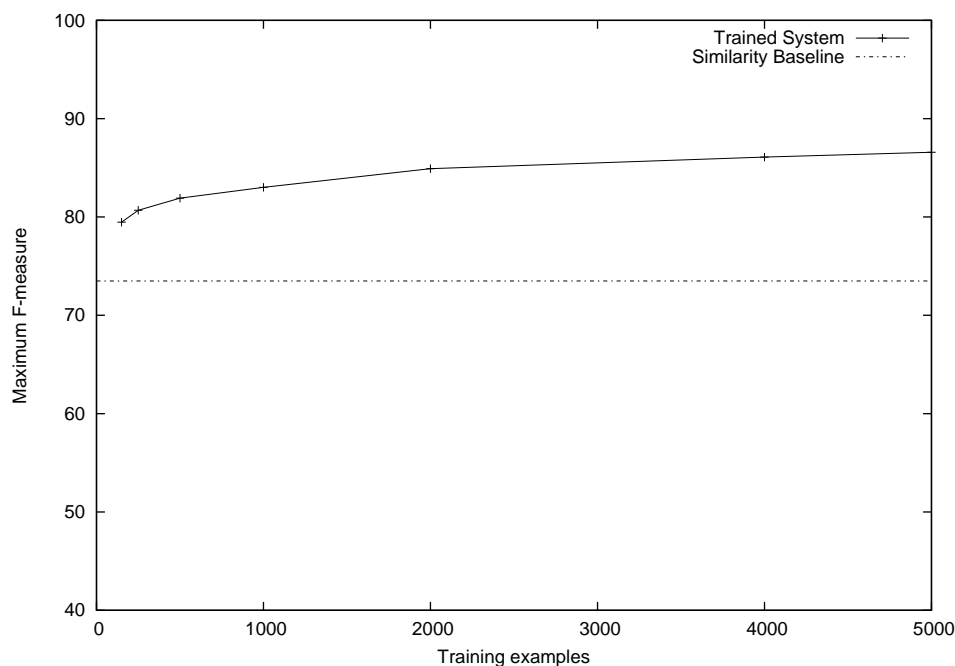


Figure 2: Learning curve for the "is_a(procedure, procedure)" relation obtained using the trained system. The similarity baseline is shown for comparison.

5 Future Work

There are several avenues for future work. Currently, our method does not do any syntactic analysis of the phrases. Clearly the syntactic structure of the phrase indicates the presence of relations with

other concepts. Hence it will be potentially useful information to exploit. One may do this by using syntactic tree kernels (Collins and Duffy, 2001; Moschitti, 2006) for computing the similarity between the descriptions. Another way to improve the performance could be by incorporating the hierarchical structure of concepts in SNOMED CT as additional features. This may help the learner generalize across concepts at similar places in the hierarchy. In future, we also want to evaluate the performance of our method on clinical phrases which are not in SNOMED CT. This will, however, require manual evaluation by experts which may be doable only on a small scale.

In future, we plan to apply the SNOMED CT relation identification method to convert clinical phrases into their SNOMED CT expressions. We have already done some preliminary experiments towards this end. In order to identify relations for a new phrase, the system needs to check every relation with every other concept. Given that there are around four hundred thousand concepts in SNOMED CT, doing this is computationally very intensive (testing an example in SVM requires computing kernels with all the training examples which have non-zero support vectors). However, we tested the idea on a subset of SNOMED CT with around 3000 concepts whose all relations are preserved within the subset. We obtained maximum F-measures for the relation identification task in this setting in the range of 10 – 20%. But given that this test dataset contains a few thousand negative examples for every positive example (random guessing will perform less than 1%), this is in fact not a bad performance, although it needs to be improved. One way to improve will be to design a top level classifier that will filter out several obvious negative examples. Some of the SNOMED CT expressions require nested use of relations, for example, the expression for the phrase (d) in Table 1. In order to compositionally build a nested SNOMED CT expression, in future one may leverage ideas from semantic parsing (Mooney, 2007), the task of converting natural language utterances into complete meaning representations.

6 Conclusions

We formulated the task of identifying SNOMED CT relations as a means for converting natural language clinical phrases into SNOMED CT expressions. We presented a machine learning approach for identifying relations and also introduced an appropriate kernel for the task. Experimental results showed that the trained system obtains a good performance on the relation identification task.

References

- Aronson, A. R. and F.-M. Lang (2010). An overview of MetaMap: Historical perspectives and recent advances. *Journal of the American Medical Informatics Association* 17, 229–236.
- Baader, F., D. Calvanese, D. McGuinness, D. Nardi, and P. Patel-Schneider (2003). *The Description Logic Handbook: Theory, Implementation, Applications*. Cambridge, UK: Cambridge University Press.
- Barrett, N., J. Weber-Jahnke, and V. Thai (2012). Automated clinical coding using semantic atoms and topology. In *Proceedings of Computer-Based Medical Systems (CBMS)*, pp. 1–6.
- Benson, T. (2010). *Principles of Health Interoperability HL7 and SNOMED*. Springer.
- Bunescu, R. C. and R. J. Mooney (2005). Subsequence kernels for relation extraction. In Y. Weiss, B. Schölkopf, and J. Platt (Eds.), *Advances in Neural Information Processing Systems 19 (NIPS 2006)*, Vancouver, BC.
- Collins, M. and N. Duffy (2001). Convolution kernels for natural language. In *Proceedings of Neural Information Processing Systems (NIPS 14)*.
- Cristianini, N. and J. Shawe-Taylor (2000). *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press.

- Friedman, C., P. O. Alderson, J. H. M. Austin, J. Cimino, and S. B. Johnson (1994). A general natural language text processor for clinical radiology. *Journal of the American Medical Informatics Association* 2, 161–174.
- Friedman, C., J. Cimino, and S. Johnson (1993). A conceptual model for clinical radiology reports. In *Proceedings of the Seventeenth Annual Symposium on Computer Applications in Medical Care (SCAMC)*, pp. 829–833.
- Friedman, C., L. Shagina, Y. Lussier, and G. Hripsack (2004). Automated encoding of clinical documents based on natural language processing. *Journal of American Medical Informatics Association* 11(5), 392–402.
- Haussler, D. (1999). Convolution kernels on discrete structures. Technical Report UCSC-CRL-99-10, UC Santa Cruz.
- IHTSDO (2008). Compositional grammar for SNOMED CT expressions in HL7 version 3. External draft for trial use. Version 0.006, December 2008. Technical report, The International Health Terminology Standards Development Organization.
- IHTSDO (2009). SNOMED Clinical Terms user guide. International release, July 2009. Technical report, The International Health Terminology Standards Development Organization.
- Jung, S., S. Kim, S. Yoo, and Y. Choi (2009). Toward the automatic generation of the entry level CDA documents. *Journal of Korean Society of Medical Informatics* 15(1), 141–151.
- Lee, D. H., F. Y. Lau, and H. Quan (2010). A method for encoding clinical datasets with SNOMED CT. *BMC Medical Informatics and Decision Making* 10:53.
- Lodhi, H., C. Saunders, J. Shawe-Taylor, N. Cristianini, and C. Watkins (2002). Text classification using string kernels. *Journal of Machine Learning Research* 2, 419–444.
- Mooney, R. J. (2007). Learning for semantic parsing. In A. Gelbukh (Ed.), *Computational Linguistics and Intelligent Text Processing: Proceedings of the 8th International Conference, CICLing 2007, Mexico City*, pp. 311–324. Berlin: Springer Verlag.
- Moschitti, A. (2006). Making tree kernels practical for natural language learning. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL-06)*, Trento, Italy, pp. 113–120.
- Platt, J. C. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In A. J. Smola, P. Bartlett, B. Schölkopf, and D. Schuurmans (Eds.), *Advances in Large Margin Classifiers*, pp. 185–208. MIT Press.
- Sager, N., C. Friedman, and M. S. Lyman (1987). *Medical Language Processing: Computer Management of Narrative Data*. Reading, MA: Addison-Wesley.
- Sager, N., M. Lyman, N. T. Nhan, and L. Tick (1995). Medical language processing: Applications to patient data representations and automatic encoding. *Methods of Information in Medicine* 34:1/2, 140–146.
- Spyns, P. and J. L. Willems (1995). Dutch medical language processing: discussion of a prototype. In *Proceedings of the Eight World Congress on Medical Informatics*, pp. 37–40.
- Spyns, P., P. Zweigenbaum, and J. L. Willems (1992). Representation and extraction of information from patient discharge summaries by means of natural language processing. In *Proceedings of MIC 92 (in Dutch)*, pp. 309–316.

- Stenzhorn, H., E. Pacheco, P. Nohama, and S. Schulz (2009). Automatic mapping of clinical documentation to SNOMED CT. *Stud Health Technol Inform* 150, 228–232.
- Uzuner, O., B. South, S. Shen, and S. DuVall (2011). 2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text. *Journal of the American Medical Informatics Association* 18, 552–556.
- Zelenko, D., C. Aone, and A. Richardella (2003). Kernel methods for relation extraction. *Journal of Machine Learning Research* 3, 1083–1106.
- Zweigenbaum, P., B. Bachimont, J. Bouaud, J. Charlet, and J. F. Boisvieux (1995). A multilingual architecture for building a normalized conceptual representation from medical language. In *Proceedings of Ninth Annual Symposium on Computer Applications in Medical Care (SCAMC)*, pp. 357–361.