A generative model for unsupervised discovery of relations and argument classes from clinical texts

Bryan Rink and Sanda Harabagiu
Human Language Technology Research Institute
University of Texas at Dallas
Richardson, TX, USA
{bryan,sanda}@hlt.utdallas.edu

Abstract

This paper presents a generative model for the automatic discovery of relations between entities in electronic medical records. The model discovers relation instances and their types by determining which context tokens express the relation. Additionally, the valid semantic classes for each type of relation are determined. We show that the model produces clusters of relation trigger words which better correspond with manually annotated relations than several existing clustering techniques. The discovered relations reveal some of the implicit semantic structure present in patient records.

1 Introduction

Semantic relations in electronic medical records (EMRs) capture important meaning about the associations between medical concepts. Knowledge about how concepts such as medical problems, treatments, and tests are related can be used to improve medical care by speeding up the retrieval of relevant patient information or alerting doctors to critical information that may have been overlooked. When doctors write progress notes and discharge summaries they include information about how treatments (e.g., aspirin, stent) were administered for problems (e.g. pain, lesion) along with the outcome, such as an improvement or deterioration. Additionally, a doctor will describe the tests (e.g., x-ray, blood sugar level) performed on a patient and whether the tests were conducted to investigate a known problem or revealed a new one. These textual descriptions written in a patient's record encode important information about the relationships between the problems a patients has, the treatments taken for the problems, and the tests which reveal and investigate the problems.

The ability to accurately detect semantic relations in EMRs, such as *Treatment-Administered-for-Problem*, can aid in querying medical records. After a preprocessing phase in which the relations are detected in all records they can be indexed and retrieved later as needed. A doctor could search for all the times that a certain treatment has been used on a particular problem, or determine all the treatments used for a specific problem. An additional application is the use of the relational information to flag situations that merit further review. If a patient's medical record indicates a test that was found to reveal a critical problem but no subsequent treatment was performed for the problem, the patient's record could be flagged for review. Similarly, if a *Treatment-Worsens-Problem* relation is detected previously in a patient's record, that information can be brought to the attention of a doctor who advises such a treatment in the future. By considering all of the relations present in a corpus, better medical ontologies could be built automatically or existing ones can be improved by adding additional connections between concepts that have a relation in text.

Given the large size of EMR repositories, we argue that it is quite important to have the ability to perform relation discovery between medical concepts. Relations between medical concepts benefit translational medicine whenever possible relations are known. Uzuner et al. (2011) show that super-

Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, pages 519–528,
Edinburgh, Scotland, UK, July 27–31, 2011.

vised methods recognize such relations with high accuracy. However, large sets of annotated relations need to be provided for this purpose. To address both the problem of discovering unknown relations between medical concepts and the related problem of generating examples for known relations, we have developed an unsupervised method. This approach has the advantages of not requiring an expensive annotation effort to provide training data for semantic relations, which is particularly difficult for medical records, characterized by many privacy concerns. Our analysis shows a high level of overlap between the manually annotated relations and those that were discovered automatically. Our experimental results show that this approach improves upon simpler clustering techniques.

The remainder of this paper is organized as follows. Section 2 discusses the related work. Section 3 reports our novel generative model for discovering relations in EMRs, Section 4 details the inference and parameter estimation of our method. Section 5 details our experiments, Section 6 discusses our findings. Section 7 summarizes the conclusions.

2 Related Work

Previous methods for unsupervised relation discovery have also relied on clustering techniques. One technique uses the context of entity arguments to cluster, while another is to perform a post-processing step to cluster relations found using an existing relation extraction system. The approaches most similar to ours have taken features from the context of pairs of entities and used those features to form a clustering space. In Hasegawa et al. (2004), those features are tokens found within a context window of the entity pair. Distance between entity pairs is then computed using cosine similarity. In another approach, Rosenfeld and Feldman (2007) use hierarchical agglomerative clustering along with features based on token patterns seen in the context, again compared by cosine similarity.

Other approaches to unsupervised relation discovery have relied on a two-step process where a number of relations are extracted, usually from a predicate-argument structure. Then similar relations are clustered together since synonymous predicates should be considered the same relation (e.g. “acquire” and “purchase”). Yates (2009) considers the output from an open information extraction system (Yates et al., 2007) and clusters predicates and arguments using string similarity and a combination of constraints. Syed and Viegas (2010) also perform a clustering on the output of an existing relation extraction system by considering the number of times two relations share the same exact arguments. Similar relations are expected to have the same pairs of arguments (e.g. “Ford produces cars” and “Ford manufactures cars”). These approaches and others (Agichtein and Gravano, 2000; Pantel and Pennacchiotti, 2006) rely on an assumption that relations are context-independent, such as when a person is born, or the capital of a nation. Our method will discover relations that can depend on the context as well. For instance, “penicillin” may be causally related to “allergic reaction” in one patient’s medical record but not in another. The relation between the two entities is not globally constant and should be considered only within the scope of one patient’s records.

Additionally, these two-step approaches tend to rely on predicate-argument structures such as subject-verb-object triples to detect arbitrary relations (Syed and Viegas, 2010; Yates et al., 2007). Such approaches can take advantage of the large body of research that has been done on extracting syntactic parse structure and semantic role information from text. However, these approaches can overlook relations in text which do not map easily onto those structures. Unlike these approaches, our model can detect relations that are not expressed as a verb, such as “[cough] + [green sputum]” to express a conjunction or “[Cl] 119 mEq / L [High]” to express that a test reading is indicating a problem.

The 2010 i2b2/VA Challenge (Uzuner et al., 2011) developed a set of annotations for medical concepts and relations on medical progress notes and discharge summaries. One task at the challenge involved developing systems for the extraction of eight types of relations between concepts. We use this data set to compare our unsupervised method with others.

The advantage of our work over existing unsupervised approaches is the simultaneous clustering of both argument words and relation trigger words. These broad clusters handle: (i) synonyms, (ii) argument semantic classes, and (iii) words belonging to the same relation.

3 A Generative Model for Discovering Relations

3.1 Unsupervised Relation Discovery

A simple approach to discovering relations between medical entities in clinical texts uses a clustering approach, e.g. Latent Dirichlet Allocation (LDA) (Blei et al., 2003). We start with an assumption that relations exist between two entities, which we call arguments, and may be triggered by certain words between those entities which we call *trigger words*. For example, given the text "[x-ray] revealed [lung cancer]", the first argument is *x-ray*, the second argument is *lung cancer*, and the trigger word is *revealed*. We further assume that the arguments must belong to a small set of semantic classes specific to the relation. For instance, *x-ray* belongs to a class of medical tests, whereas *lung cancer* belongs to a class of medical problems. While relations may exist between distant entities in text, we focus on those pairs of entities in text which have no other entities between them. This increases the likelihood of a relation existing between the entities and minimizes the number of context words (words between the entities) that are not relevant to the relation.

With these assumptions we build a baseline relation discovery using LDA. LDA is used as a baseline because of its similarities with our own generative model presented in the next section. Each consecutive pair of entities in text is extracted, along with the tokens found between them. Each of the entities in a pair is split into tokens which are taken along with the context tokens to form a single *pseudo-document*. When the LDA is processed on all such pseudo-documents, clusters containing words which co-occur are formed. Our assumption that relation arguments come from a small set of semantic classes should lead to clusters which align with relations since the two arguments of a relation will co-occur in the pseudo-documents. Furthermore, those argument tokens should co-occur with relation trigger words as well.

This LDA-based approach was examined on electronic medical records from the 2010 i2b2/VA Challenge data set (Uzuner et al., 2011). The data set contains manually annotated medical entities which were used to form the pairs of entities needed. For example, Figure 1 illustrates examples of two clusters out of 15 discovered automatically using LDA on the corpus. The first cluster appears to contain words which indicate a relation whose two arguments are both medical problems (e.g. "disease", "cancer", "edema"). The trigger words seem to indicate a possible causal relation (e.g., "due", "related", "secondary"). The second cluster contains words relevant to medical tests (e.g. "examination", "culture") and their findings ("revealed", "showed", "positive"). As illustrated in Figure 1, some of the context words are not necessarily related to the relation. The word "patient" for instance is present in both clusters but is not a trigger word because it is likely to be seen in the context of any relation in medical text. The LDA-based model treats all words equally and cannot identify which words are likely trigger words and which ones are *general words*, which merely occur frequently in the context

Cluster 1

Words: secondary, due, likely, patient, disease, liver, abdominal, cancer, pulmonary, respiratory, elevated, volume, chronic, edema, related

"Correct" instances: [Metastatic colon cancer] with [abdominal carcinomatosis]; [symptoms] were due to [trauma]

"Incorrect" instances: [mildly improving symptoms] , plan will be to continue with [his current medicines]; [prophylaxis] against [peptic ulcer disease]

Cluster 2:

Words: examination, no, positive, culture, exam, blood, patient, revealed, cultures, physical, out, urine, notable, showed, cells

"Correct" instances: [a blood culture] grew out [Staphylococcusaureus]; [tamponade] by [examination]

"Incorrect" instances: [the intact drain] draining [bilious material]; [a Pseudomonas cellulitis] and [subsequent sepsis]

Figure 1: Two clusters found by examining the most likely words under two LDA topics. The instances are pseudo-documents whose probability of being assigned to that cluster was over 70%

of a relation.

In addition, while the LDA approach can detect argument words which co-occur with trigger words (e.g., "examination" and "showed"), the clusters produced with LDA do not differentiate between contextual words and words which belong to the arguments of the relation. An approach which models arguments separately from context words could learn the semantic classes of those arguments and thus better model relations. Considering the examples from Figure 1, a model which could cluster "examination", "exam", "cultures", and "culture" into one *medical test* cluster and "disease", "cancer" and "edema" into a *medical problem* cluster separate from the relation trigger words and general words should model relations more accurately by better reflecting the implicit structure of the text. Because of these limitations many relations discovered in this way are not accurate, as can be seen in Figure 1.

3.2 Relation Discovery Model (RDM)

The limitations identified in the LDA-based approach are solved by a novel relation discovery model (RDM) which jointly models relation argument semantic classes and considers them separately from the context words. Relations triggered by pairs of medical entities enable us to consider three observable features: (A1) the first argument; (A2) the second argument; and (CW) the context words found between A1 and A2.

For instance, in sentence S1 the arguments are A1="some air hunger" and A2="his tidal volume" while the context words are "last", "night", "when", "I", and "dropped".

S1: *He developed [some air hunger]$_{PROB}$ last night when I dropped [his tidal volume]$_{TREAT}$ from 450 to 350.*

In the RDM, the contextual words are assumed to come from a mixture model with 2 mixture components: a relation trigger word ($x = 0$), or a general word ($x = 1$), where x is a variable representing which mixture component a word belongs to. In sentence S1 for example, the word "dropped" can be seen as a trigger word for a *Treatment-Causes-Problem* relation. The remaining words are not trigger words and hence are seen as general words.

Under the RDM's mixture model, the probability of a context word is:

$$
\begin{aligned}
&P(w^C|t^r, z) = \\
&\quad P(w^C|t^r, x = 0) \times P(x = 0|t^r) \; + \\
&\quad P(w^C|z, x = 1) \times P(x = 1|t^r)
\end{aligned}
$$

Where w^C is a context word, the variable t^r is the relation type, and z is the general word class. The variable x chooses whether a context word comes from a relation-specific distribution of trigger words, or from a general word class. In the RDM, the two argument classes are modeled jointly as $P(c^1, c^2|t^r)$, where c^1 and c^2 are two semantic classes associated with a relation of type t^r. However the assignment of classes to arguments depends on a directionality variable d. If $d = 0$, then the first argument is assigned semantic class c^1 and the second is assigned class c^2. When $d = 1$ however, the class assignments are swapped. This models the fact that a relation's arguments do not come in a fixed order, "[MRI] revealed [tumor]" is the same type of relation as "[tumor] was revealed by [x-ray]". Figure 2 shows the graphical model for the RDM. Each candidate relation is modeled independently, with a total of I relation candidates. Variable w^1 is a word observed from the first argument, and w^2 is a word observed from the second argument. The model takes parameters for the number of relations types (R), the number of argument semantic classes (A), and the number of general word classes (K). The generative process for the RDM is:

1. For relation type $r = 1..R$:
    (a) Draw a binomial distribution σ_r from $Beta(\alpha^x)$ representing the mixture distribution for relation r
    (b) Draw a joint semantic class distribution $\psi_r^{1,2} \in \mathbb{R}^{C \times C}$ from $Dirichlet(\alpha^{1,2})$.
2. Draw a categorical word distribution $\phi_{z'}^z$ from $Dirichlet(\beta^z)$ for each general word class $z' = 1..K$
3. Draw a categorical word distribution $\phi_{r'}^r$ from $Dirichlet(\beta^r)$ for each $r' = 1..R$
4. for semantic class $a' = 1..A$:
    (a) Draw categorical word distributions $\omega_{a'}^1$ and $\omega_{a'}^2$ from $Dirichlet(\beta^1)$ and $Dirichlet(\beta^2)$ for the first and second arguments, respectively.

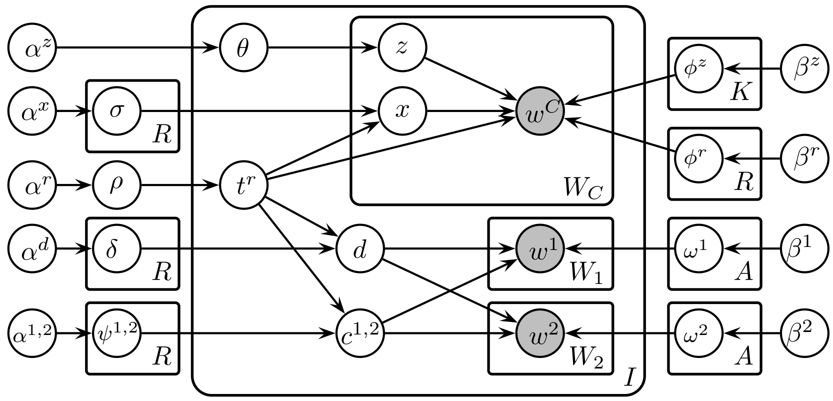

Figure 2: Graphical model for the RDM. $c^{1,2}$ represents the joint generation of c^1 and c^2

$$P(t^r, d|\boldsymbol{t^r_{-i}}, \boldsymbol{d}_{-i}, \boldsymbol{c^{1,2}_{-i}}, \boldsymbol{x}_{-i}, \boldsymbol{z}_{-i}, \boldsymbol{w^C_{-i}}, \boldsymbol{w^1_{-i}}, \boldsymbol{w^2_{-i}}; \boldsymbol{\alpha}, \boldsymbol{\beta}) \propto u_1 \times u_2 \times u_3$$
$$u_1 = \frac{f(t^r)+\alpha^r}{I+R\alpha^r} \times \frac{f(t^r,d)+\alpha^d_d}{f(t^r)+\alpha^d_0+\alpha^d_1} \times \frac{f(t^r,c^1,c^2)+\alpha^{1,2}}{f(t^r)+C\times C\alpha^{1,2}}$$
$$u_2 = \prod_j^{W_C} \frac{f_i(z_j)+\alpha^z}{W_C+K\alpha^z} \times \frac{f(t^r,x_i)+\alpha^x}{f(t^r)+2\alpha^x} \times (\mathbf{1}_{x=0}\frac{f(t^r,w^C_j)+\beta^r}{f(t^r)+W\beta^r} + \mathbf{1}_{x=1}\frac{f(z_j,w^C_j)+\beta^z}{f(z_j)+W\beta^z})$$
$$u_3 = \prod_j^{W_1} \frac{f(a^1,w^1_j)+\beta^1}{f(a^1)+W\beta^1} \times \prod_j^{W_2} \frac{f(a^2,w^2_j)+\beta^2}{f(a^2)+W\beta^2}$$

Figure 3: Gibbs sampling update equation for variables t^r and d for the i^{th} relation candidate. The variables $a^1 = c^1$ and $a^2 = c^2$ if $d = 0$, or $a^1 = c^2$ and $a^2 = c^1$ if $d = 1$. W is the size of the vocabulary. $f(\bullet)$ is the count of the number of times that event occurred, excluding assignments for the relation instance being sampled. For instance, $f(t^r, d) = \sum_{k\neq i}^{I} I[t^r_k = t^r_i \wedge d_k = d_i]$

5. Draw a categorical relation type distribution ρ from $Dirichlet(\alpha^r)$
6. For each pair of consecutive entities in the corpus, $i = 1..I$:
   (a) Sample a relation type t^r from ρ
   (b) Jointly sample semantic classes c^1 and c^2 for the first and second arguments from $\psi^{1,2}_{t^r}$
   (c) Draw a general word class categorical distribution θ from $Dirichlet(\alpha^z)$
   (d) For each token $j = 1..W_1$ in the first argument: Sample a word w^1_j from $\omega^1_{c^1}$ if $d = 0$ or $\omega^2_{c^2}$ if $d = 1$
   (e) For each token $j = 1..W_2$ in the second argument: Sample a word w^2_j from $\omega^2_{c^2}$ if $d = 0$ or $\omega^1_{c^1}$ if $d = 1$
   (f) For each token $j = 1..W_C$ in the context of the entities:
      i. Sample a general word class z from θ
      ii. Sample a mixture component x from σ_{t^r}
      iii. Sample a word from $\phi^r_{t^r}$ if $x = 0$ or ϕ^z_z if $x = 1$.

In the RDM, words from the arguments are informed by the relation through an argument semantic class which is sampled from $P(c^1, c^2|t^r) = \psi^{1,2}_{t^r}$. Furthermore, words from the context are informed by the relation type. These dependencies enable more coherent relation clusters to form during parameter estimation because argument classes and relation trigger words are co-clustered.

We chose to model two distinct sets of entity words (ω^1 and ω^2) depending on whether the entity occurred in the first argument or the second argument of the relation. The intuition for using disjoint sets of entities is based on the observation that an entity may be expressed differently if it comes first or second in the text.

4 Inference and Parameter Estimation

Assignments to the hidden variables in RDM can be made by performing collapsed Gibbs sampling (Griffiths and Steyvers, 2004). The joint probability of the data is:

$$
\begin{aligned}
&P(\boldsymbol{w^C}, \boldsymbol{w^1}, \boldsymbol{w^2}; \boldsymbol{\alpha}, \boldsymbol{\beta}) \propto \\
&P(\sigma|\alpha^x)P(\rho|\alpha^r)P(\delta|\alpha^d)P(\psi^{1,2}|\alpha^{1,2}) \\
&\times P(\phi^z|\beta^z)P(\phi^r|\beta^r)P(\omega^1|\beta^1)P(\omega^2|\beta^2) \\
&\times \prod_i^I [P(\theta_i|\alpha^z)P(t_i^r|\rho)P(d_i|t^r, \delta_{t^r})P(c_i^1, c_i^2|t^r, \psi^{1,2}) \\
&\times \prod_j^{W_{C,i}} P(z_{i,j}|\theta_i)P(x_{i,j}|t_i^r, \sigma_{t_i^r})P(w_{i,j}^C|x_{i,j}, t_i^r, z_{i,j}) \\
&\times \prod_j^{W_{1,i}} P(w_j^1|d_i, c_i^{1,2}, \omega^1) \\
&\times \prod_j^{W_{2,i}} P(w_j^2|d_i, c_i^{1,2}, \omega^2)]
\end{aligned}
$$

We need to sample variables t^r, d, $c^{1,2}$, x, and z. We sample t^r and d jointly while each of the other variables is sampled individually. After integrating out the multinomial distributions, we can sample t^r and d from the equation in Figure 3

The update equations for the remaining variables can be derived from the same equation by dropping terms which are constant across changes in that variable.

In our experiments the hyperparameters were set to $\alpha^x = 1.0, \alpha^z = 1.0, \alpha^{1,2} = 1.0, \alpha_0^d = 2, \alpha_1^d = 1, \beta^r = 0.01, \beta^z = 0.01, \beta^1 = 1.0, \beta^2 = 1.0$. Changing the hyperparameters did not significantly affect the results.

5 Experimental Results

5.1 Experimental Setup

We evaluated the RDM using a corpus of electronic medical records provided by the 2010 i2b2/VA Challenge (Uzuner et al., 2011). We used the training set, which consists of 349 medical records from 4 hospitals, annotated with medical concepts (specifically problems, treatments, and tests), along with any relations present between those concepts. We used these manually annotated relations to evaluate how well the RDM performs at relation discovery. The corpus is annotated with a set of eight relations: *Treatment-Addresses-Problem*, *Treatment-Causes-Problem*, *Treatment-Improves-Problem*, *Treatment-Worsens-Problem*, *Treatment-Not-Administered-due-to-Problem*, *Test-Reveals-Problem*, *Test-Conducted-for-Problem*, and *Problem-Indicates-Problem*. The data contains 13,460 pairs of consecutive concepts, of which 3,613 (26.8%) have a relation belonging to the list above. We assess the model using two versions of this data set consisting of: those pairs of consecutive entities which have a manually annotated relation (DS1), and secondly, all consecutive pairs of entities (DS2). DS1 allows us to assess the RDM's clustering without the noise introduced from those pairs lacking a true relation. Evaluations on DS2 will indicate the level of degradation caused by large numbers of entity pairs that have no true relation. We also use a separate test set to assess how well the model generalizes to new data. The test set contains 477 documents comprising 9,069 manually annotated relations.

Relation 1	Relation 2	Relation 3	Relation 4
mg	(	due	showed
p.r.n.	)	consistent	no
p.o.	Working	not	revealed
hours	ICD9	likely	evidence
prn	Problem	secondary	done
q	Diagnosis	patient	2007
needed	30	(	performed
day	cont	started	demonstrated
q.	):	most	without
4	closed	s/p	normal
2	SNMCT	seen	shows
every	**ID-NUM	related	found
one	PRN	requiring	showing
two	mL	including	negative
8	ML	felt	well

Figure 4: Relation trigger words found by the RDM

5.2 Analysis

Figure 4 illustrates four of the fifteen trigger word clusters (most likely words according to ϕ^r) learned from dataset DS1 using the best set of parameters according to normalized mutual information (NMI) as described in section 5.3. These parameters were: $R = 9$ relations, $K = 15$ general word classes, and $A = 15$ argument classes. Examination of the most likely words reveals a variety of trigger words, beyond obvious explicit ones. Example sentences for the relation types from Figure 4 are presented in Figure 5 and discussed below.

Relation Type 1

Instances of this discovered relation are often found embedded in long lists of drugs prescribed to the patient. Tokens such as "p.o." and "p.r.n.", meaning respectively "by mouth" and "when necessary", are indicative of a prescription relation. The learned relation specifically considers arguments of a drug

Instances of Relation Type 1

1. Haldol 0.5-1 milligrams p.o. q.6-8h. p.r.n. agitation
2. plavix every day to prevent failure of these stents
3. KBL mouthwash , 15 ccp .o. q.d. prn mouth discomfort
4. Miconazole nitrate powder tid prn for groin rash
5. AmBisome 300 mg IV q.d. for treatment of her hepatic candidiasis

Instances of Relation Type 2

1. MAGNESIUM HYDROXIDE SUSP 30 ML) , 30 mL , Susp , By Mouth , At Bedtime , PRN , For Constipation
2. Depression , major (ICD9 296.00 , Working , Problem) cont NOS home meds
3. Diabetes mellitus type II (ICD9 250.00 , Working , Problem) cont home meds
4. ASCITES (ICD9 789.5 , Working , Diagnosis) on spironalactone
5. *Dilutional hyponatremia (SNMCT **ID-NUM , Working , Diagnosis) improved with fluid restriction

Instances of Relation Type 3

1. ESRD secondary to her DM
2. slightly lightheaded and with increased HR
3. a 40% RCA , which was hazy
4. echogenic kidneys consistent with renal parenchymal disease
5. *Librium for alcohol withdrawal

Instances of Relation Type 4

1. V-P lung scan was performed on May 24 2007 , showed low probability of PE
2. a bedside transthoracic echocardiogram done in the Cardiac Catheterization laboratory without evidence of an effusion
3. exploration of the abdomen revealed significant nodularity of the liver
4. Echocardiogram showed moderate dilated left atrium
5. An MRI of the right leg was done which was equivocal for osteomyelitis

Figure 5: Examples for four of the discovered relations. Those marked with an asterisk have a different manually chosen relation than the others

and a symptom treated by that drug. The closest manually chosen relation is *Treatment-Addresses-Problem* which included drugs as treatments.

Relation Type 2

Relation 2 captures a similar kind of relation to Relation 1. All five examples for Relation 1 in Figure 5 came from a different set of hospitals than the examples for Relation 2. This indicates the model is detecting stylistic differences in addition to semantic differences. This is one of shortcomings of simple generative models. Because they cannot reflect the true underlying distribution of the data they will model the observations in ways that are irrelevant to the task at hand. Relation 2 also contains certain punctuation, such as parentheses which the examples show are used to delineate a treatment code. Instances of Relation 2 were often marked as *Treatment-Addresses-Problem* relations by annotators.

Relation Type 3

The third relation captures problems which are related to each other. The manual annotations contain a very similar relation called *Problem-Indicates-Problem*. This relation is also similar to Cluster 1 from Section 3.1, however under the RDM the words are much more specific to the relation. This relation is difficult to discover accurately because of the infrequent use of strong trigger words to indicate the relation. Instead, the model must rely more on the semantic classes of the arguments, which in this case will both be types of medical problems.

Relation Type 4

The fourth relation is detecting instances where a medical test has revealed some problem. This corresponds to the *Test-Reveals-Problem* relation from the data. Many good trigger words for that relation have high probability under Relation 4. A comparison of the RDM's Relation 4 with LDA's cluster 2 from Figure 1 shows that many words not relevant to the relation itself are now absent.

Argument classes

Figure 6 shows the 3 most frequent semantic classes

Concept 1	Concept 2	Concept 3
CT	pain	Percocet
scan	disease	Hgb
chest	right	Hct
x-ray	left	Anion
examination	renal	Vicodin
Chest	patient	RDW
EKG	artery	Bili
MRI	-	RBC
culture	symptoms	Ca
head	mild	Gap

Figure 6: Concept words found by the RDM

for the first argument of a relation (ω^1). Most of the other classes were assigned rarely, accounting for only 19% of the instances collectively. Human annotators of the data set chose three argument classes: *Problems*, *Treatments*, and *Tests*. Concept 1 aligns closely with a test semantic class. Concept 2 seems to be capturing medical problems and their descriptions. Finally, Concept 3 appears to be a combination of treatments (drugs) and tests. Tokens such as "Hgb", "Hct", "Anion", and "RDW" occur almost exclusively in entities marked as tests by annotators. It is not clear why this cluster contains both types of words, but many of the high ranking words beyond the top ten do correspond to treatments, such as "Morphine", "Albumin", "Ativan", and "Tylenol". Thus the discovered argument classes show some similarity to the ones chosen by annotators.

5.3 Evaluation

For a more objective analysis of the relations detected, we evaluated the discovered relation types by comparing them with the manually annotated ones from the data using normalized mutual information (NMI) (Manning et al., 2008). NMI is an information-theoretic measure of the quality of a clustering which indicates how much information about the gold classes is obtained by knowing the clustering. It is normalized to have a range from 0.0 to 1.0. It is defined as:

$$NMI(\Omega;\mathbb{C}) = \frac{I(\Omega;\mathbb{C})}{[H(\Omega)+H(\mathbb{C})]/2}$$

where Ω is the system-produced clustering, $\mathbb{C}$ is the gold clustering, I is the mutual information, and H is the entropy. The mutual information of two clusterings can be defined as:

$$I(\Omega,\mathbb{C}) = \sum_k \sum_j \frac{|\omega_k \cap c_j|}{N} log_2 \frac{N|\omega_k \cap c_j|}{|\omega_k||c_j|}$$

where N is the number of items in the clustering. The entropy is defined as

$$H(\Omega) = -\sum_k \frac{|\omega_k|}{N} log_2 \frac{|\omega_k|}{N}$$

The reference clusters consist of all relations annotated with the same relation type. The predicted clusters consist of all relations which were assigned the same relation type.

In addition to NMI, we also compute the F measure (Amigó et al., 2009). The F measure is computed as:

$$F = \sum_i \frac{|L_i|}{n} \text{max}_j \{F(L_i, C_j)\}$$

where

$$F(L_i, C_j) = \frac{2 \times Recall(L_i, C_j) \times Precision(L_i, C_j)}{Recall(L_i, C_j) + Precision(L_i, C_j)}$$

and $Precision$ is defined as:

$$Precision(C_i, L_j) = \frac{|C_i \cap L_j|}{|C_i|}$$

while $Recall$ is simply precision with the arguments swapped:

$$Recall(L, C) = Precision(C, L)$$

Table 1 shows the NMI and F measure scores for several baselines along with the RDM. Evaluation was performed on both DS1 (concept pairs having a manually annotated relation) and DS2 (all consecutive concept pairs). For DS2 we learned the models using all of the data, and evaluated on those entity pairs which had a manual relation annotated. The LDA-based model from Section 3.1 is used as one baseline. Two other baselines are K-means and Complete-Link hierarchical agglomerative clustering using TF-IDF vectors of the context and argument words (similar to Hasegawa et al. (2004)).

Method	DS1		DS2	
	NMI	F	NMI	F
Train set				
Complete-link	4.2	37.8	N/A	N/A
K-means	8.25	38.0	5.4	38.1
LDA baseline	12.8	23.0	15.6	26.2
RDM	18.2	39.1	18.1	37.4
Test set				
LDA baseline	10.0	26.1	11.5	26.3
RDM	11.8	37.7	14.0	36.4

Table 1: NMI and F measure scores for the RDM and baselines. The first two columns of numbers show the scores when evaluation is restricted to only those pairs of concepts which had a relation identified by annotators. The last two columns are the NMI and F measure scores when each method clusters all consecutive entity pairs, but is only evaluated on those with a relation identified by annotators.

Complete-link clustering did not finish on DS2 because of the large size of the data set. This highlights another advantage of the RDM. Hierarchical agglomerative clustering is quadratic in the size of the number of instances to be clustered, while the RDM's time and memory requirements both grow linearly in the number of entity pairs. The scores shown in Table 1 use the best parameterization of each model as measured by NMI. For DS1 the best LDA-based model used 15 clusters. K-means achieved the best result with 40 clusters, while the best Complete-Link clustering was obtained by using 40 clusters. The best RDM model used parameters $R = 9$ relation, $K = 15$ general word classes, and $A = 15$ argument classes. For DS2 the best number of clusters for LDA was 10, while K-means performed best with 58 clusters. The best RDM model used $R = 100$ relations, $K = 50$ general word classes, and $A = 15$ argument classes. The LDA-based approach saw an improvement when using the larger data set, however the RDM still performed the best.

To assess how well the RDM performs on unseen data we also evaluated the relations extracted by the model on the test set. Only the RDM and LDA models were evaluated as clusters produced by K-means and hierarchical clustering are valid only for the data used to generate the clusters. Generative models on the other hand can provide an estimate of the probability for each relation type on unseen text. For each model we generate 10 samples after a burn in period of 30 iterations and form clusters by assigning each pair of concepts to the relation assigned most often in the samples. The results of this evaluation are presented in Table 1. While these cluster scores are lower than those on the data used to train the models, they still show the RDM outperforming the LDA baseline model.

6 Discussion

The relation and argument clusters determined by the RDM provide a better unsupervised relation discovery method than the baselines. The RDM does this using no knowledge about syntax or semantics outside of that used to determine concepts. The analysis shows that words highly indicative of relations are detected and clustered automatically, without the need for prior annotation of relations or even the choice of a predetermined set of relation types. The discovered relations can be interpreted by a human or labeled automatically using a technique such as the one presented in Pantel and Ravichandran (2004). The fact that the discovered relations and argument classes align well with those chosen by annotators on the same data justify our assumptions about relations being present and discoverable by the way they are expressed in text. Table 1 shows that the model does not perform as well when many of the pairs of entities do not have a relation, but it still performs better than the baselines.

While the RDM relies in large part on trigger words for making clustering decisions it is also capable of including examples which do not contain any contextual words between the arguments. In addition to modeling trigger words, a joint distribution on argument semantic classes is also incorporated. This allows the model to determine a relation type even in the absence of triggers. For example, consider the entity pair "[lung cancer] [XRT]", where XRT stands for external radiation therapy. By determining the semantic classes for the arguments (lung cancer is a Problem, and XRT is a test), the set of possible relations between the arguments can be narrowed down. For instance, XRT is unlikely to be in a causal relationship with a problem, or to make a problem worse. A further aid is the fact that the learned relationships may be specialized. For instance, there may be a learned relation type such as "Cancer treatment addresses cancer problem". In this case, seeing a type of cancer (lung cancer) and a type of cancer treatment (XRT) would be strong evidence for that type of relation, even without trigger words.

7 Conclusions

We presented a novel unsupervised approach to discovering relations in the narrative of electronic medical records. We developed a generative model which can simultaneously cluster relation trigger words as well as relation arguments. The model makes use of only the tokens found in the context of pairs of entities. Unlike many previous approaches, we assign relations to entities at the location those entities appear in text, allowing us to discover context-sensitive relations. The RDM outperforms baselines built using Latent Dirichlet Allocation and traditional clustering methods. The discovered relations can be used for a number of applications such as detecting when certain treatments were administered or determining if a necessary test has been performed. Future work will include transforming the RDM into a non-parametric model by using the Chinese Restaurant Process (CRP) (Blei et al., 2010). The CRP can be used to determine the number of relations, argument classes, and general word classes automatically.

References

Eugene Agichtein and Luis Gravano. 2000. Snowball: extracting relations from large plain-text collections. In *Proceedings of the Fifth ACM Conference on Digital libraries*, pages 85–94, San Antonio, Texas, United States. ACM.

E. Amigó, J. Gonzalo, J. Artiles, and F. Verdejo. 2009. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information Retrieval*, 12(4):461–486.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

David M. Blei, Thomas L. Griffiths, and Michael I. Jordan. 2010. The nested chinese restaurant process and bayesian nonparametric inference of topic hierarchies. *J. ACM*, 57(2):1–30.

T. L Griffiths and M. Steyvers. 2004. Finding scientific topics. *Proceedings of the National Academy of Sciences of the United States of America*, 101(Suppl 1):5228.

Takaaki Hasegawa, Satoshi Sekine, and Ralph Grishman. 2004. Discovering relations among named entities from large corpora. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, ACL '04, Stroudsburg, PA, USA. Association for Computational Linguistics. ACM ID: 1219008.

C. D Manning, P. Raghavan, and H. Schütze. 2008. *Introduction to information retrieval*, volume 1. Cambridge University Press.

P. Pantel and M. Pennacchiotti. 2006. Espresso: Leveraging generic patterns for automatically harvesting semantic relations. In *Annual Meeting Association for Computational Linguistics*, volume 44, page 113.

P. Pantel and D. Ravichandran. 2004. Automatically labeling semantic classes. In *Proceedings of HLT/NAACL*, volume 4, page 321–328.

Benjamin Rosenfeld and Ronen Feldman. 2007. Clustering for unsupervised relation identification. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, CIKM '07, page 411–418, New York, NY, USA. ACM. ACM ID: 1321499.

Z. Syed and E. Viegas. 2010. A hybrid approach to unsupervised relation discovery based on linguistic analysis and semantic typing. In *Proceedings of the NAACL HLT 2010 First International Workshop on Formalisms and Methodology for Learning by Reading*, page 105–113.

Ozlem Uzuner, Brett South, Shuying Shen, and Scott DuVall. 2011. 2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text. *Accepted for publication.*

A. Yates, M. Cafarella, M. Banko, O. Etzioni, M. Broadhead, and S. Soderland. 2007. TextRunner: open information extraction on the web. In *Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics*, page 25–26.

Alexander Yates. 2009. Unsupervised resolution of objects and relations on the web. *Journal of Artificial Intelligence Research*, 34(1).