How Text Mining Can Help Lexical and Commonsense Knowledgebase Construction

Niloofar Montazeri
niloofar@isi.edu

Jerry R. Hobbs
Hobbs@isi.edu

Eduard Hovy
hovy@isi.edu

Information Sciences Institute
University of Southern California
Marina del Rey, California

Abstract

In an enterprise called "deep lexical semantics", we develop various core theories of fundamental commonsense phenomena and define English word senses by means of axioms using predicates explicated in these theories. This enables deep inferences that require commonsense knowledge about how the world functions. There are difficulties in our approach to manually axiomatize words and commonsense knowledge. First, developing axioms is done by experts and this means the process is slow and expensive. Second, it is hard, if possible at all, to predict in advance all the kinds of axioms that should be encoded in core theories. In this paper we present a method for harvesting from free-form text on the web, simple axioms for change-of-state verbs which is a combination of text-mining and manual filtering. Focusing on two change-of-state verbs, "break" and "cut", we show how the harvested axioms can help in addressing the above problems.

Introduction

If computers are to understand and reason about statements as humans do, they need a lot of knowledge as well as appropriate mappings from words to this knowledge. We are engaged in an enterprise we call "Deep Lexical Semantics" in which we develop various core theories of fundamental commonsense phenomena and define English word senses by means of axioms using predicates explicated in these theories. Among the core theories are causality, scales and change-of-state. The combination of the axioms defining words and axioms in core theories should make deep inferences possible. For example, if our core theory of composite entities has an axiom that says "pieces are smaller than the whole", and an axiom that defines "cut" as "causing a change from being whole into being in pieces", we can infer from "she cut the paper" that "the pieces of paper were smaller than the starting page".

However, there are difficulties in our approach to manually axiomatize words and commonsense knowledge. The first problem is that developing axioms is done by experts and this means the process is slow and expensive. The second problem is that one cannot predict in advance all the kinds of axioms that should be encoded in core theories.

In this work, we investigate the possibility of addressing these difficulties by extracting lexical knowledge from large corpora. First, we present a method for harvesting axioms for change-of-state verb senses, which is a combination of text mining and manual filtering. Then we present a case study on the extracted axioms for two change-of-state verbs "break" and "cut". We observe that in many cases, the manually-encoded axioms can be found in the harvested inferences. This suggests the possibility of replacing the process of manual axiomatization by experts with a semi-automatic method consisting of text mining and non-expert annotations. We also show how the harvested inferences can help us to enrich the core theories by revealing missing concepts and axioms in these theories.

Related Work

There are a number of relevant works that extract inference rules from text. (Lin and Pantel 2001) developed a method for learning inference rules using distributional similarity between dependency tree paths. The rules obtained by their method mostly express synonymy or similarity relations such as "x is the author of y ↔ x wrote y". (Chklovski and Pantel 2004) use lexico-syntactic patterns over the Web to detect similarity, strength, antonymy, enablement and temporal happens-before relations between pairs of strongly associated verbs. (Girju 2003) extracts causal relations between event nouns using lexico-syntactic patterns. An example of such relations is ―Earthquakes generate tsunami.

The difference between our work and the aforementioned works is that we are interested in entailments rather than synonymy, before-after or causality relations. (Gordon and Schubert 2011) describe a method for acquiring a collection of conditional (if–then) knowledge by exploiting presuppositional discourse patterns (such as ones involving 'but', 'yet', and 'hoping to') and abstracting the matched material into general rules. Their method is able to extract rules that describe complex consequences or reasons. An example of such rules is "If a male stands before a female in the doorway, then he may expect to be invited in". Another rule which is more similar to what our method extracts is "driver in a car crash might be injured". Although the results of our procedure include expected consequences of an event (for example, "cutting skin results in bleeding") extracting such "expectation" rules is not our final goal. Instead, we are more interested in lexical entailment rules such as "breaking of a device" entails "a change from the device being functional" which is almost always true.

(Pekar 2006) developed methods based on regular co-occurrence of two verbs, within locally coherent text, for learning the implications of an event. They extracted such rules as "(x was appointed as y) suggests (x became y)". However, they do not require that verb entailment holds in all conceivable contexts and view it as a relation that may be more plausible in some contexts than others while the kinds of axioms that we are interested in, are context independent. In addition, we do not limit the entailment between two verbs; rather we may have entailment between a verb (an event) and an adjective related to the state before or after that event.

Deep Lexical Semantics

In this section we give a brief introduction to our deep lexical semantics project (Hobbs, 2008).

Framework

We use first-order logic for encoding axioms in our commonsense theories, in the syntax of Common Logic (Menzel et al., 2008). Since human cognition concerns itself with actual and possible events and states, which we refer to as eventualities, we reify these and treat them in the logic as ordinary individuals. Most axioms are only normally true, and we thus have an approach to defeasibility—proofs can be defeated by better proofs. Our approach to defeasibility is based on weighted abduction (Hobbs et al., 1993) and is similar to McCarthy's circumscription (McCarthy, 1980).

We use a logical notation in which states and events (eventualities) are reified. Specifically, if the expression *p(x)* says that **p** is true of **x**, then *p'(e, x)* says that **e** is the eventuality of **p** being true of **x**. Eventuality **e** may exist in the real world (Rexist), in which case *p(x)* holds, or it may only exist in some modal context, in which case that is expressed simply as another property of the possible individual **e**.

Core Theories

A *core theory* is a set of predicates and axioms that describe the relationships among a set of very abstract concepts that govern or characterize many aspects of the world we live in. In fact, core theories are the kind of commonsense knowledge that are trivial and humans acquire them in their early childhood. For example, a child understands such concepts as change, composite entity, scale, falling an moving. There are a number of works that have successfully extracted less basic types of commonsense knowledge from resources such as Wikipedia (Sangweon, et.al 2006), eHow (Jihee, 2010) which contain such knowledge as nutrition of eggs, step by step instructions for cracking and separating an egg and that eggs break easily. However, we believe that the kind of commonsense knowledge encoded in core theories can hardly be obtained from written resources or by asking human subjects to generate them.

In the following, we briefly describe two core theories, composite entities and change-of-state, as examples.

A composite entity is a thing composed of other things. The concept is general enough to include complex physical objects (e.g., a telephone), complex events (e.g., the process of erosion) and complex information structures (e.g., a theory). A composite entity is characterized by a set of components, a set of properties, and a set of relations. Example predicates that are defined in this theory are *compositeEntity(x)* which simply says **x** is a composite entity; *componentsOf(s, x)* which says that **s** is the set of **x**'s components; and *componentOf(y, x)* which says that **x** is a component of the composite entity **y**. The relations between these concepts are captured in several axioms. For example the following axiom defines the relationship between the above predicates:

componentOf(y, x) ↔ compositeEntity(x) & componentsOf(s, x) & member(y, s)

The above axiom says that **y** is a component of **x** if and only if **x** is a composite entity, **s** is the set of components of **x** and **y** is a member of **s**. In this axiom, the predication *member(y, s)* comes from our core theory of sets.

The theory of change-of-state is the most relevant to this paper. The predication *change'(e, e1, e2)* says that **e** is a change-of-state whose initial state is **e1** and whose final state is **e2**. The chief properties of change are that there is some entity whose state is undergoing change, that change is defeasibly transitive, that **e1** and **e2** cannot be the same unless there has been an intermediate state that is different, and that change is consistent with the before relation from our core theory of time (Hobbs et al., 2004). An example axiom in this theory is:

$$change'(e, e1, e2) \;\&\; change'(e0, e2, e3) \rightarrow change'(e4, e1, e3)$$

Which states that change is transitive: if there is a change from state **e1** to state **e2** and a change from state **e2** to state **e3**, then we have a change from state **e1** to state **e3**.

Since many lexical items focus only on the initial or the final state of a change, we introduce for convenience the predications *changeFrom'(e, e1)* and *changeTo'(e, e2)*. The first predication says that **e** is a change from state **e1** to another state that is inconsistent with **e1**. In other words after the change, state **e1** no longer exists. Similarly, the second predication says that **e** is a change to state **e2** that didn't exist before the change.

Linking Word Senses to Core Theories

There are existing resources such as VerbNet that have explicated the meaning of verbs in terms of basic concepts and have the potential to be used as a basis for axiomatizing meaning of verbs. Despite its high quality, VerbNet has its own limitations. The first limitation is its coverage. For example, VerbNet only covers those senses of "break" that refer to "appearing", "splitting", "hurting", "giving up a habit" and "breaking apart" and the senses referring to "breaking a law", "breaking a record" and "breaking news" are missing. The synonym of "breaking a record" which is "better a record", is categorized under "exceed", but no semantic decomposition for "exceed" is provided. The second limitation of VerbNet is that the axioms obtained from this resource are very general. For example, while we would like the axiomatization of "breaking an instrument" to contain such information as "making non-functional", we cannot get this information from the semantic decomposition of verb classes "hurt" or "break apart". Therefore, although VerbNet is a good resource for axiomatizing verbs, still other methods are needed to enrich the set of its axioms by axiomatizing missing word senses as well as finding more specific axioms for the existing word senses.

In the reminder of this section, we describe our method for harvesting from free-form web text, axioms for change-of-state verb senses from large corpora. First we describe how to identify verb senses by considering their patient types and then we show how to harvest axioms for different verb senses. In the later sections we show that axioms defining some of these verb senses can be found among a small set of harvested axioms.

Identifying Word Senses

We cannot axiomatize a word that has multiple meanings or senses, rather we should axiomatize a word sense which refers to a single concept. WordNet (Miller, 1995) is our main reference for identifying senses of words; however, we do not stick to the WordNet senses. Our observation about change-of-state verbs is that in many cases, different senses of a verb can be identified by the type of patient that it takes. For example, the meaning of the verb "break" when its patient P is of type "instrumentality" (e.g., radio, motor, car) is "P stops working". On the other hand, when P is of type "message" (e.g., news, story), the meaning of "break" is "P becomes known". Thus we represent verb senses by a tuple (V, T), where V represents the verb and T represents the type of its patient. A word is of type T if T is its ancestor in the WordNet hierarchy.

To identify different patient types that can distinguish between different senses of V, we create small groups of patient word senses, such that each group corresponds to a different sense of V. For each group, we take the first common hypernym in WordNet hierarchy as the type of that group. For example, we can create two groups of patients for "break": group1 = {car-s1, radio-s1, device-s1} and group2 = {news-s1, information-s1, story-s5}[1]. The first common hypernym of these groups are "instrumentality" and "message" respectively.

Encoding Axioms

Since in this paper we deal with change-of-state verbs, we describe here the general format of an axiom defining a change-of-state verb sense (V, T), which can be any of the following:

(1) $V'(e, p) \;\&\; T(p) \rightarrow changeFrom'(e, e1) \;\&\; \langle predications\ specifying\ e1 \rangle$

(2) $V'(e, p) \;\&\; T(p) \rightarrow changeTo'(e, e2) \;\&\; \langle predications\ specifying\ e2 \rangle$

In the above axioms, *V'(e, p)* means that **e** is the eventuality of **p** undergoing **V** and *T(p)* means that **p** is of type **T**. Axiom (1) says that the eventuality of **p** undergoing event **V**, where **p** is of type **T**, is a change from a state **e1**, where **e1** is specified by an additional predication. Similarly axiom (2) says eventuality **e** is a change to a state **e2**. The additional predications that define **e1** and **e2** are the essential part of an axiom defining a change-of-sate verb sense (V, T) and differ from one verb sense to the other. For example, for the verb sense (break, instrumentality), the previous state **e1** can be specified by the predication *function'(e1,p)*. In other words, if an instrumentality **p** breaks, there is a change from the state "p functions". Thus the complete axiom is:

$$break'(e, p) \;\&\; instrumentality(p) \rightarrow$$

[1] The postfix s-i represents the sense number in WordNet.

changeFrom'(e, e1) & function'(e1, p)

In the above axiom, the predicate "function" comes from our core theory of functionality.

The essence of an axiom that defines a change of state is the predication that describes exactly which state changes. This information must be made explicit and coupled to the change-of-state verb. To obtain such information at large scale, we turn to automated text harvesting. We note that an important part of an axiom is its argument structure. In the above example, the state that is changed (function) is the state of the patient of the verb. However this is not always the case. For example, consider breaking a law which we define as a change to the law-breaker being guilty:

break'(e, x, l) & law(l) →
changeTo'(e, e1) & guilty'(e1, x)

Here, the agent of the event "break" (x) is undergoing the change, not the patient (l). It should be possible to use syntax for automatically capturing the argument structure of an axiom. However, sometimes this information is not present in syntax or one cannot obtain it without co-reference resolution. For now we leave the task of determining the argument structures for future work and only focus on the predicates. In the next section, we present a method for harvesting predicates that specify the before-state (e1) or after-state (e2) of a given change-of-state verb sense (V, T).

Harvesting State Predicates

We harvested information from the ClueWeb09 dataset, whose English portion contains just over 500 million web pages extracted from the Web in 2009 by Prof. Jamie Callan and his group at CMU[2].

Predicates refer to basic concepts which may be represented by words with different parts of speech[3]. Since we are looking for concepts that describe a state, and since a state is usually described by adjectives or verbs, we only look for words with these parts of speech. In particular, we harvest a list of verbs (ZList) and a list of adjectives (JList) that specify predicates describing before-states and after-states of a verb sense (V, T). Here we describe the procedure for extracting ZList only. JList can be extracted by slightly modifying this procedure.

Finding Patterns

The first step is to find patterns that capture change-of-state. The patterns for before-state are different from those for after-state. To find patterns for before-state, we create a seed (V, P, Z), where V is the verb for which we are harvesting states, P is a patient that verb sense (V, T) can take and Z is a verb that is strongly related to before-state of (V, T). An example of a good seed is (break, machine, work), since we know that "a machine works before it breaks". We use these seeds to extract from the corpus a list of sentences that contain all the three words in the seed. Then we filter these sentences and keep only those in which P is the patient of V and there is a before-relationship between Z and V. Next, we parse these sentences with a dependency parser to find the dependencies between V, P and Z in each sentence. This gives us a list of dependency patterns. Some of these patterns which for simplicity we translate into phrases are:

- P Zed until it Ved (the TV worked until it broke)
- P no longer Zs since it Ved (the TV no longer works since it broke)
- Even though P Ved, it still Zs (even though the machine broke, it still works).

Patterns for after-states are extracted similarly. With this method we have extracted 11 patterns for before-states and 7 patterns for after-states. We also add an additional pattern which matches any sentence in which both V and Z occur such that either of them has P as its patient. We will refer to this pattern as P_0. These 19 patterns are further expanded to 19*3 patterns by adding the feature of whether the patient of V, the patient of Z or none of them is a pronoun.

Evaluating the Patterns

We use precision as the measure for evaluating patterns. To compute the precision of patterns, we used these patterns to harvest ZList and JList (as described in the next section) for 16 change-of-state verb senses from several Levin classes. We used the baseline function (described in

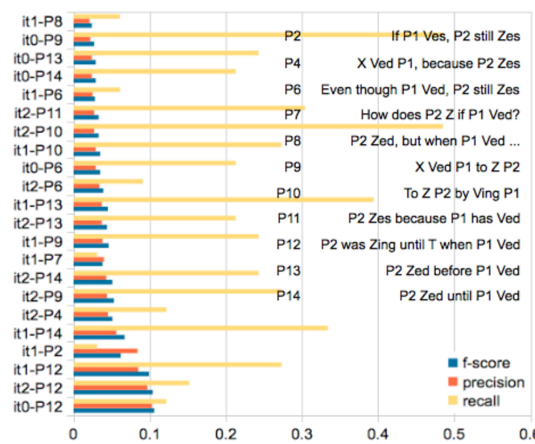

Figure 1: Distribution of pattern precision and recall (only patterns with precision>0.02 are shown.)

next section), for ranking ZList and JList. Then for each of the 16 verbs, we searched among the first 500 words to

---

[2] http://lemurproject.org/clueweb09.php/

[3] Hence we use the words "predicate", "adjective", "verb" and "word" interchangeably.

obtain a gold set of verbs and adjectives. We obtained about 100 adjectives and 30 verbs in total. Then for each verb, we computed the precision of different patterns over the first 500 words and adjectives and took the mean over all 16 cases. Figure 1 shows the precision, as well as the recall and f-score for 22 before/after-state patterns in extracting verbs. For space reasons, we have only shown the statistics for patterns with precision above 0.02 (as we will see later, this is the best threshold for scoring). Pattern Ids are on the vertical axis. The prefix it_0 in pattern Ids indicates that none of the patients of V or Z are pronouns; it_1 indicates that V's patient (P_1) is a pronoun and it_2 indicates that Z's patient (P_2) is a pronoun. A phrasal representation of the patterns (without the prefix it_i) is also shown in this figure.

Harvesting Candidate Predicates

To harvest state predicates for a verb sense (V, T), we first need the list of words with type T. Since T is a synset in WordNet, we can obtain this list from the synsets that are under synset T in the WordNet hierarchy. Each synset S that is under type T in WordNet hierarchy consists of several words. We filter these words and keep only those that have a high frequency and S is among their first two synsets. (This prevents a word such as "king" to be considered as an instrumentality). We add all such words for different synsets to a list called TWordList.

To obtain ZList we first filter the corpus and keep only sentences containing both V and a word in TWordList. Next, we parse these sentences using a dependency parser and match them against the set of patterns. We obtain a list of tuples (Z, f, l) where Z is the extracted word; f is the total number of times that Z is extracted and l is a list of pattern-frequency tuples. In a pattern-frequency tuple (p_i,f_i), pi is the pattern Id and f_i is the number of times that p_i extracted Z.

Ranking the Candidates

To rank the extracted words, we assign a score to each word based on its frequency, the patterns that extracted it, the frequency of patterns and the precision of patterns.

To find a good scoring function for ranking candidate tuples (Z, f, l), we experimented with 4 different scoring functions (m_1-m_4) and compared them with the baseline function (m_0) which simply takes f as the score. These 4 scoring functions are as follows:

m_1: $s = \sum f_i$, for all (p_i, f_i) in l, where $f_i > t$, $i \neq 0$

m_2: $s = \sum f_i$, for all (pi, f_i) in l, $i \neq 0$

$$m_3: s = \begin{cases} f & if\ \exists\ (pi, fi) \in l; fi > t; i \neq 0 \\ 0 & otherwise \end{cases}$$

$$m_4: s = \begin{cases} f & if\ \exists\ (pi, fi) \in l;\ i \neq 0 \\ 0 & otherwise \end{cases}$$

where t is the threshold for pattern precision. To evaluate these scores, we use the same gold data that we created for evaluating patterns. The best scoring method should be the one that has higher recall values in lower ranks. It turned out that the best scoring functions for verbs and adjectives are m_3 with t=0.02 and m_0 respectively. Figure 2 shows the percentage of gold verbs and adjectives that the best scoring function (among m_1-m_4) and the baseline function (m_0) delivered. We can see that for verbs, m_3 has a 25% recall at the cut-off rank 30, while the baseline has a recall

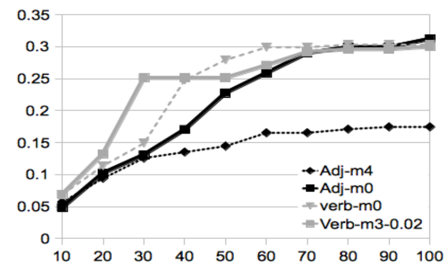

Figure 2.Comparison of the best scoring function (among m1-m4) and the baseline function (m0)

of 15% only. For adjectives, the baseline is as good as m_3 before rank 30 and is much better thereafter. According to this graph, a good cut-off rank that contains 25% of good results is 30 for verbs and 60 for adjectives.

Case Study

In this section, we study the positive results (i.e., adjectives and verbs that describe the previous or next state of an event) harvested for different senses of two change-of-state verbs "break" and "cut" to investigate their potential for axiomatizing these verb senses as well as helping experts to enrich the core theories. These positive cases are chosen by one of the authors, among the words with a rank lower than the corresponding cut-off. Tables 1 and 2 show a comparison between our manually encoded predicates and the extracted verbs and adjectives for the verbs "break" and "cut" respectively. For space reasons, we only present the results for seven senses of each verb, for three of which we could obtain good axioms. Each row represents a verb sense specified by the patient type T. The first column shows the patient type T. The second column shows our hand-coded states and the third and fourth columns show those members of ZList and JList that we have selected from the top 30 and 70 items in these lists. We have marked entries related to previous states with P and those related to next states with N. The successful extractions are marked in bold. We have categorized the axioms that will be obtained from each verb and adjective in ZList and JList into defeasible and indefeasible. An axiom is indefeasible if it is true without more assumptions and is defeasible otherwise. The verbs and adjectives that yield defeasible axioms are marked in italic.

Definitional Axioms

If the extracted verbs and adjectives are to be used in axioms defining verb senses, they need to represent states that are indefeasibly true before or after the corresponding event. As the results in Tables 1 and 2 show, in many cases the harvested verbs and adjectives correspond to indefeasible axioms and in some cases they exactly match our manually encoded predicates: breaking a law changes the state of the law-breaker to being "guilty"; breaking a piece of information (such as news) changes the state of it from being "unknown". Cutting a substance changes the state of it from being "more" and cutting a piece of writing changes its state to being "short". In some cases such as "breaking a law" and "breaking a record", there are several indefeasible candidates such as {guilty, accountable, responsible} and {best, leader, hold, stand} respectively. In such cases, we can have several axioms, one per each candidate.

Table 1. Verbs and adjectives extracted for "break"

Patient Type (T)	Hand-Coded State	ZList	JList
Law	N: **guilty**		N: **guilty**, accountable, responsible, *criminal*
Record	P: **best**	P: hold, stand	P: **best**, leader
Communication	P: **unknown**	P: keep	P: happy **unknown**, N: *happy*
Device	P: function	P: worked	P: *good*
Solid	P: integrated		P: *good*
Body Part	N:injured	P: *play, work, run*	
Group	P: complete	P: stay	

Table 2. Verbs and adjectives extracted for the verb "cut"

Patient Type (T)	Hand-Coded State	ZList	JList
Body Part	N: **open**, injured	N: *bleed, die, heal* P: *grow*	N:**open** , *dead*
Substance	P: **more**	N: *starve*	P: **more**, many, most, *enough* N: little, low, small, few
Piece of Writing	P: **long**	N: *fit*	P: **long** N: short
Person	P: member		
Vehicle	P: go straight	N: spin	
Check	?		

There are some verb senses for which there are no useful predicates among the extracted lists. As shown in Table 1, while we defined the previous state of "breaking a solid" as "being integrated" (alternatively we could define the next state as "being separated"), there are no predicates in ZList or JList that indicate integration or separation. This is a rather expected outcome as this kind of knowledge is too obvious to be stated in text. There are also cases such as "cut a check" for which we again didn't find any relevant predicate among the candidates; but this time, "cutting a check" does not refer to an obvious commonsense concept. In this case, it is even hard for an expert to think of a state that is changed by "cutting a check".

Enrichment of Core Theories

Now we show how the extracted data can reveal missing axioms in our core theories.

When there are several indefeasibly true before/after-states for one verb sense, there should be an entailment relationship among them. A good example is "breaking a record" for which we have 3 alternative before-states: "the record held", "the record was the best" and "the record breaker was the leader". In this case, there is an entailment relationship between each pair of states; since they all are referring to a single concept. To enable entailment between "the record being the best" and "the record breaker being the leader", we need to make sure that the following axioms are in our commonsense knowledge base:

(1) $record(r, x, f) \leftrightarrow best(r, f) \,\&\, have(x, r)$

(2) $have(x, r) \,\&\, best(r, f) \leftrightarrow best(x, f)$

(3) $best(x, f) \leftrightarrow leader(x, f)$

The first axiom says if **r** is a record by **x** in field **f**, then **r** is the best in **f** and **x** is the holder of **r**. The second axiom says **x**'s holding something that is the best in field **f** is the same as **x**'s being the best in field **f**. The last axiom says **x**'s being the best in field **f** is the same as **x**'s being the leader in that field. The above axioms also enable entailment between "being best" and "being leader" as well as entailment between "being a record" and "being the best". To enable entailment between "record holds" and "record is the best", we need a more complex set of rules that together make it possible to make such inferences as: "holding means being valid", "a record is valid at time **t** if and only if it is the best at time **t**", "a record is the best in a field only until another record is set in the same field", "when **r1** is the best in field **f** until time **t** and at time **t** , **r2** becomes the best, **r1** is not the best thereafter".

As a second example for how the extracted information help enriching the knowledgebase, we consider "breaking a law", for which we got three indefeasibly true after-states: "being guilty", "being accountable" and "being responsible". In order to make possible the entailment between "being guilty" and "being responsible", we need to make sure that the following axioms are in our commonsense knowledge base:

$guilty'(e, x, a) \rightarrow$
$\quad agentOf(x, a) \,\&\, against(a, l) \,\&\, law(l)$

$agentOf(x, a) \rightarrow responsible(x, a)$

The first axiom says: if **x** is guilty for action **a**, then **x** is the agent of **a** and **a** is against the law **l**. The second axiom says: if **x** is the agent of an action **a**, then **x** is responsible for **a**.

As mentioned before, defeasible axioms are those that are not necessarily true without making additional assumptions. In Table 1, we have three defeasible cases: the state after “breaking a law” is “being criminal”, the state before or after “breaking news” is “being happy”, and the state before “breaking a solid” is “being good”. Similarly, in Table 2 we have defeasible cases such as the state after “cutting an organ” is “bleeding”. None of these are necessarily true and thus they cannot be used in axioms defining breaking a law, news, solid or cutting an organ. However, they still provide us with interesting knowledge such as:

- *A criminal is a guilty person*
- *learning about information can make someone happy*
- *integrity of a solid entity is good (or a change from being integrated is bad)*

which can be axiomatized and added to the respective core theories.

One problem in using the extracted words as predicates is that these words may not be a predicate in our core theories. For example, the word responsible is not a predicate in our theory. In such cases, we have two choices: replacing these words with their definitions anchored in core theories (for example, the definition of responsible(x, e0) is agentOf(x, e0)). The second option is to add these words as new concepts to our core theories. This is where the harvested words help in enriching our knowledge base with new predicates (concepts).

Conclusions and Future Work

We presented an experiment for using text mining to facilitate the process of building a lexical and a commonsense knowledge base. In this experiment we focused on change-of-state verb senses (whose axiomatization requires information about the state changed by their corresponding event) and developed a harvesting method to obtain a rich set of candidate states for each verb sense. We showed that this short list of states contains candidates that are as good as the states chosen by experts. This suggests that we may be able to replace the expert effort in encoding axioms with a simple annotation task done via Mechanical Turk. We also showed that the extracted information can help enriching the core theories by revealing missing axioms in these theories. Our future work includes improving the patterns as well as the scoring functions to obtain a richer and smaller set of candidate axioms. We would like to apply our methodology to a larger set of verb senses and have Mechanical Turk annotators filter the resulted candidate list. We also need to develop a procedure for enriching the axioms with argument structure which will also enable us to harvest more complex axioms consisting of several predicates. We used a manual method for clustering different patients of a verb, but in practice we need an automatic method that given a verb, a large corpus and perhaps resources such as WordNet, finds all possible patients of that verb and clusters them into different types, such that each cluster represents a different sense of the verb. Finally, we would like to measure the coverage and specificity of our axioms against VerbNet’s axioms, as well as the overlap between these two axiom collections.

References

(Chklovski, Pantel, 2004) Chklovski, Timothy, and Patrick Pantel. "Verbocean: Mining the web for fine-grained semantic verb relations." Proceedings of EMNLP. Vol. 4. 2004.

(Girju 2003) Girju, Roxana. "Automatic detection of causal relations for question answering." Proceedings of the ACL 2003 workshop on Multilingual summarization and question answering-Volume 12. Association for Computational Linguistics, 2003.

(Gordon, Schubert, 2011) Gordon, Jonathan, and Lenhart K. Schubert. "Discovering commonsense entailment rules implicit in sentences." Proceedings of the TextInfer 2011 Workshop on Textual Entailment. Association for Computational Linguistics, 2011.

(Hobbs, 1993) J.R. Hobbs, M. Stickel, D. Appelt, and P. Martin. Interpretation as abduction. Artificial Intelligence 63,

69–142. 1993.

(Hobbs et al., 2004) J.R. Hobbs and F. Pan. “An Ontology of Time for the Semantic Web”. ACM Transactions on Asian Language Processing (TALIP): Special issue on Temporal Information Processing, Vol. 3, No. 1, pp. 66-85. 2004.

(Hobbs, 2008) Hobbs, Jerry. "Deep lexical semantics." Computational Linguistics and Intelligent Text Processing (2008): 183-193.

(Jihee, 2010) Ryu, Jihee, et al. "Automatic extraction of human activity knowledge from method-describing web articles." 1st Workshop on Automated Knowledge Base Construction. 2010.

(Lin and Pantel 2001) Lin, Dekang, and Patrick Pantel. "DIRT@ SBT@ discovery of inference rules from text." Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2001.

(McCarthy, 1980) J. McCarthy, “Circumscription: A Form of Nonmonotonic Reasoning”, Artificial Intelligence, 13: 27-39.

1980.

(Miller, 1995) G. Miller, “WordNet: a lexical database for English”. In Communications of the ACM 38 (11), pp. 39-

41.22. 1995.

(Pekar 2006) Pekar, Viktor. "Acquisition of verb entailment from text." Proceedings of the Human Language Technology Conference of the NAACL, Main Conference. 2006.

(Sangweon, et.al 2006) Suh, Sangweon, Harry Halpin, and Ewan Klein. "Extracting common sense knowledge from Wikipedia." Proceedings of the Workshop on Web Content Mining with Human Language Technologies at ISWC. Vol. 6. 2006.

(Tenorth 2011)Tenorth, Moritz, et al. "Web-enabled robots." Robotics & Automation Magazine, IEEE 18.2 (2011): 58-68.