Thesis Proposal
Verb Semantics for Natural Language Understanding

Derry Tanti Wijaya

Abstract

A verb is the organizational core of a sentence. Understanding the meaning of the verb is therefore key to understanding the meaning of the sentence. Natural language understanding is the problem of mapping natural language text to its meaning representation: entities and relations anchored to the world. Since verbs express relations over their arguments in text, a lexical resource about verbs can facilitate natural language understanding by mapping verbs in text to relations over entities expressed by their arguments in the world. In this thesis, we propose an automatic construction of a verb resource that contains important semantics for natural language understanding. We propose to learn three important semantics about verbs that will complement existing resources on verbs such as WordNet and VerbNet. The three semantics are (1) the mapping of verbs to relations in knowledge bases, (2) the pre- and post-conditions of each verb on its arguments: the entry condition of entities that allow an event expressed by the verb to take place and the condition of entities that will be true after the event occurs, (3) the temporal sequences of verbs. The mapping of verbs to relations in knowledge bases such as NELL, YAGO, or Freebase can provide a direct link between the text and the background knowledge about the world in the knowledge bases; enabling inferences over the world knowledge to better understand the text. The pre- and post-conditions of verbs on their arguments and the temporal sequences of verbs will open a range of new inference options for natural language understanding; for example for predicting events, for temporal scoping of events, for inferring the cascading effect of events, for inferring the states of entities in text, or for understanding the meaning of a sentence in the context of other sentences in the same document.

1 Introduction

Modern theories of grammar see sentences as consisting of predicates and their arguments where the predicates are relations or functions over the arguments. Since verbs express relations over their arguments in sentences, this representation suggests that the verbs (and their auxiliaries) in sentences are predicates and the noun phrases that they appear with are their arguments. Using computational linguistics' definition of the term "meaning" as a relation between a linguistic sign and some non-linguistic entity to which the sign refers – a concept in the mind, an entity in the world, etc (Ovchinnikova, 2012); verbs are therefore an important link between the *surface* form of a sentence (i.e., the predicate-argument construction) and the *meaning* of the sentence – the entities anchored to the world and the relations among them – denoted by the arguments and predicates in the sentence. Learning about verbs is therefore important because a verb resource can potentially enable a system that can read *any* relation in the world expressed by predicates in sentences. Even where predicates in sentences are implicit, e.g. in noun compounds (e.g. sleeping pill vs. headache pill), possessives (e.g. Shakespeare's tragedy vs. Shakespeare's house), or prepositional phrases (e.g. John in the house vs. John in anger), they can be rephrased to include predicates and therefore verbs (Wijaya and Gianfortoni, 2011): pill that *causes* sleep vs. pill that *alleviates* headache, the

tragedy that is *written* by Shakespeare vs. the house where Shakespeare *lives*, and John is *located* in the house vs. John is *feeling* anger. Thus, verbs provide a high-coverage *vocabulary* of relations described in natural language.

Natural language understanding (NLU) is the problem of mapping natural language text to its meaning representation: entities and relations anchored to the world. A verb resource can facilitate NLU by mapping predicates in text (and arguments therefore, since predicates are functions over arguments) to relations over entities anchored to the world. The benefit of anchoring to the world is multifold. In order to understand natural language text, it is often not enough to know only the lexical meaning of the words; knowledge and reasoning about world entities and relations to which the text refers is often necessary. Some typical natural language phenomena that require world knowledge for their resolution include (1) syntactic ambiguity (e.g. Jen look at the *man* with a telescope vs. Jen look at the *beach* with a telescope – the first sentence is ambiguous while the second is not since a telescope can be carried by the *man* but not the *beach*), (2) anaphoric resolution (e.g. Jen gave the bananas to the monkeys because they were *hungry* vs. Jen gave the bananas to the monkey because they were *ripe* – *they* in the first sentence refers to the monkeys while *they* in the second sentence refers to the bananas), (3) discourse relation (e.g. events standing in different temporal relations: John and Marry are married. He proposed to her last year. vs. John and Marry are married. They celebrate their first anniversary last year. – proposal is temporally *before* marriage, while anniversary is temporally *after* marriage), and (4) implicit predicates (e.g. morning coffee vs. morning newspaper – coffee *drunk* in the morning vs. newspaper *read* in the morning since coffee is typically drunk while newspaper is typically read). Knowledge and reasoning about the world entities and relations to which the text refers is crucial for interpretation of these natural language phenomena. Knowledge bases such as NELL, YAGO, Freebase, or ConceptNet provide the ontologies that capture this world knowledge about entities and their relations that can enable an inference engine to reason about them. To better understand a natural language text, we therefore need to interface our lexical semantic resource with these knowledge bases of real world entities and relations.

If we think about verbs in sentences as predicates over arguments, then natural language understanding can be formulated as the mapping of *verbs* to *relations* and their *arguments* to *entities* in some knowledge bases so as to enable reasoning over them. A lexical semantic resource about verbs that can directly enable such mapping would be beneficial towards natural language understanding. Unfortunately existing resource on verbs are limited in their interface to knowledge bases. Some existing resources classify verbs into semantic classes either manually (e.g. WordNet) or automatically (e.g. DIRT). However, it is not directly clear how these verb classes map to real world relations in knowledge bases. Other resources provide a basis for defining semantic relations between verbs and their arguments in terms of semantic roles (e.g. PropBank, VerbNet, FrameNet) where verbs express frames, with a separate set of roles defined for each frame. However, it is also not clear how the verb classes or frames map to real world relations in knowledge bases, or how the semantic roles, which are just labels in natural language form e.g., *Agent*, *Patient*, *Theme*, map to real world entities in knowledge bases.

Furthermore, existing verb resources that are semantically richest: WordNet and FrameNet are manually constructed, making scalability a challenge. Knowledge bases have thousands of real world relations that are expected to grow over time. Manually annotating verbs to map to the growing vocabularies of real world relations is expensive. Such verb resource should be automatically constructed and grow in coverage with the knowledge bases, leveraging on corpus statistics from large corpora such as the Web to learn high coverage mapping of verbs to real world relations.

In addition, to achieve better natural language understanding, we need to go beyond understanding a sentence in isolation. A document or a collection of sentences is more than the sum of its words. Suppose we are reading the following document fragment.

> *John and Marry are married.*

Without any other information, we cannot be sure whether John and Mary are married to each other or separately. But if we see this following sentence from the same document.

> *He proposed to her last year.*

Then we know that they are married to each other instead of separately. The document is not just the sum of its verbs 'married' and 'proposed'; there is a semantic relationship between the relations denoted by the two verbs – the relation denoted by the verb 'married' follows temporally to the relation denoted by the verb 'proposed'. Such semantic relation between relations denoted by verbs, in this case temporal sequence, is important for better natural language understanding of sentences that is not done in isolation, rather in the context of other sentences in the same document.

Verb resources like FrameNet and VerbOcean differ from other existing resources in that they define semantic relations between verb classes (frames). FrameNet also has additional information that link roles of the connected frames, e.g., in FrameNet GIVING and GETTING frames are connected by the causation relation and the role *Recipient* in GIVING is linked to the role *Recipient* in GETTING. This feature make these resources useful for reasoning. However, FrameNet are constructed manually hence its coverage is limited. VerbOcean is constructed automatically from Web data but relies on manually constructed regular expression patterns for extracting semantic relations between paraphrases. Other approaches for finding semantic relations between verbs that have potentially higher coverage can be found in textual entailment works (Weisman et al., 2012) and narrative schemas (script) finding (Chambers and Jurafsky, 2008), which leverage on verb co-occurrence in large corpus such as the Web to learn relationships between verbs. A verb resource whose goal is to improve natural language understanding should leverage on existing resource such as FrameNet that contains precisely constructed information useful for reasoning, while constructed automatically from large corpus such as the Web for higher coverage.

This thesis seeks to answer the following questions: Can we automatically construct a lexical semantic resource that maps verbs to real world relations in any given knowledge base, leveraging both on high coverage corpus and precisely constructed existing resources? Can we demonstrate the usefulness of such lexical semantic resource on verbs for improving natural language understanding? Can we add other information to the verb resource that can enable better understanding of sentences in the context of the document instead of in isolation? Can the verb resource grows to accommodate new verbs and new relations in knowledge bases? Can it extend the relation vocabulary of the knowledge bases?

To answer these questions, I propose to use an existing verb resource such as WordNet or FrameNet, a knowledge base of entities and relations connected to the real world such as NELL, YAGO or Freebase, and corpus statistics from large corpora (i.e., the Web) to create a resource that maps verbs in the corpora to relations in the knowledge base. I will do this by aligning the existing verb resource and the knowledge base with the corpus statistics acting as an *interlingua* that links the verb resource and the knowledge base. By aligning the verb resource and the knowledge base, we can use information from the verb resource such as semantic relationships between relations, syntactic realizations of verb classes, selectional restrictions and semantic roles of arguments which can be useful for reasoning on the knowledge base.

Further, to add structure that can be useful for reasoning on the knowledge base, I also propose to learn a particular semantic relation – temporal sequence – between event-expressing verbs, especially those that map to some relations in knowledge bases. Previous work on textual entailment between verbs or narrative ordering of verbs have focused on finding related verbs based on their co-occurrence in text or their sharing of arguments. I propose to look deeper into the reason behind such chaining of event-expressing verbs from the perspective of their shared arguments. I hypothesize that event-expressing verbs, in particular the change-of-state verbs typically have pre-condition and post-condition on their arguments i.e., they typically require that the arguments participating in the event satisfy some pre-condition and that the event changes the state of the arguments (post-condition). For example, the verb "*marry*" that map to HASSPOUSE relation in YAGO typically requires that its direct object changes state from being "*single*" (pre-condition) to being "*married*" (post-condition).

Some temporal event chains can then be modeled as a collection of events chained together by the pre-condition and post-condition overlap of their shared arguments. Knowledge of pre-condition and post-condition of arguments also opens up interesting opportunities for other inferences and reasoning e.g., to infer links between semantic roles of temporally related events, or to infer how the effect of one event can be *cascaded* down via the pre- and post-condition of the shared arguments to other events and other arguments. I will evaluate the usefulness of this temporal sequence between

events to temporally scope relation instances (facts) in knowledge bases – to find the beginning and end times of facts in the knowledge bases.

To extend the vocabulary of relations in the knowledge base, it suffices to add every semantically similar verb group as a new relation in the knowledge base. However, to make the additional relations more meaningful for reasoning, it is useful to add structures with these additional relations. Therefore, I will explore how the resulting temporal sequence of event-expressing verbs can be used to extend the vocabulary of relations whilst adding also structures, in this case temporal sequence, that can be useful for reasoning. For example, if we always see a group of verbs (expressing a relation in the knowledge base) being followed temporally by another group of verbs (not yet expressing any relation in the knowledge base), we can add a new relation in the knowledge base to complete this temporal chain of verbs. Completing knowledge bases can make the knowledge graph denser and improve inference.

To evaluate the usefulness of the verb resource for increasing precision and recall of a NLU system, now that we have the mapping of verbs to relations and the temporal sequence of verbs, we can use it to disambiguate a verb in a document to the relation it expresses in the context of other temporally relevant verbs in the document.

2 Background

2.1 Theory of Verbs

Finding and representing the meaning of verbs are important because verbs express events or states with their arguments as participants, making them the organizational core of the sentence, so their meaning is key to sentence meaning. In the linguistic study of verb meaning, verbs can be categorized into stative or dynamic (Dowty, 1979). The stative verbs express states of the arguments. The dynamic verbs express events; in particular the change-of-state verbs express the change of state of the arguments participating in the event (Fillmore, 1968). The stative verbs have a simple event structure: [y ⟨**STATE**⟩] or [**BECOME** [y ⟨**STATE**⟩]] while the change of state verbs have a complex event structure: [[x **ACT**] **CAUSE** [**BECOME** [y ⟨**STATE**⟩]]] where x and y are arguments of the verbs.

If we link event-expressing verbs to relations in a knowledge base and arguments of these verbs to entities in the knowledge base, then we can represent the meaning of a change-of-state verb as the creation (the defining of **begin_time**) or deletion (the defining of **end_time**) of the relation instance expressed by the verb over entities expressed by the verb's arguments.

Some temporal chain of events can be modeled as chain of atomic events occurring on entities such that the post-condition of affected entities of the preceding atomic event is the pre-condition of the next atomic event on the entities (Croft, 1991). Representing change-of-state verbs as change in the knowledge base (i.e, creation/deletion of relation instances) allows us to model the event chain as the cascading of changes to other relation instances in the knowledge base that share an entity with the changed instance i.e., events are chained together by the *pre-condition* and *post-condition* of their shared arguments. Duration of events expressed by verbs, an important aspectual notion of verbs (Dowty, 1979), can be approximated from the duration between the initial state change to the final state change brought about by the verbs.

Representing the meaning of verbs as changes in the arguments' relation values is compelling because it links the change-of-state expressed by verbs in text to the change of relation instances in the knowledge base, opening up other avenues for discovering temporal chains of these event-expressing verbs based on temporal sequence of changes in the knowledge base. Such chains can help us disambiguate the meaning of a verb in a sentence better, given the presence of other verbs in other sentences of the document.

2.2 Conceptual Dependency Theory

The idea of understanding a sentence by representing events underlying the sentence as concepts (objects and actions) and the change-of-state caused by the actions on the objects can be found in Conceptual Dependency Theory (Schank and Abelson, 2013). This theory focuses on extracting

concepts from the sentence and modeling their dependencies: objects can perform actions, actions can be performed on objects, objects can be described by attributes (states), actions can change the state of objects, etc. Actions are reduced to eleven primitives that can account for most actions in the physical world. For example, PTRANS is a primitive action that causes change-of-state, namely the change in the *location* of a physical object. States of objects can be described by scales with numerical values (e.g. health, fear, mental state, consciousness), absolute values (e.g., length, color, mass, speed), or relationships between objects (e.g., CONTROL, POSSESSION, PROXIMITY, etc).

The advantage of Conceptual Dependency Theory is that it is a language-independent meaning representation – different words and structures can represent the same concept. Words can trigger conceptual dependency structures that can provide predictions about what will come next – identifies conceptual roles and helps disambiguation. It also facilitates inference – unstated change of state or states of unknown words can be inferred. Inference is also attached to concepts so there is no need for complex inference rules. However, the critique of this theory is that it is incomplete. There is no ontology of objects or actions. The states of the objects are defined ad-hoc (e.g., the state of being "dead" means having a "health" score of -10 while being "tip top" means having a score of +7). The set of chosen primitive actions are also critiqued in terms of whether it is sufficient to represent all the possible events in the world. To resolve these issues, we propose to focus instead on verbs to represent actions. This will provide a high-coverage *vocabulary* of actions and events described in natural language. Secondly, we propose to map objects to entities in knowledge bases and actions to relations in knowledge bases. This will give an ontology to objects and actions and a clear definition of states and change-of-states as relation instances and their updates, respectively, in the knowledge bases.

3 Related Work

In this section I discuss existing lexical semantic resources and point out where information relevant for natural language understanding is still missing. I would also discuss the importance of mapping lexical resources, which contain semantic knowledge about words, and knowledge bases, which contain conceptual knowledge about entities and relations anchored to the world.

3.1 Lexical Semantic Resources

In discussing lexical semantic resources which contain meaning representation for words, it is useful to talk about meaning representation itself to understand what representations are suitable for expressing linguistic meaning, which information should be included, and how it can be constructed. Meaning representation in linguistic theories can be discussed in terms of these three frameworks (Ovchinnikova, 2012): *formal semantics*, *lexical semantics*, and *distributional semantics*.

3.1.1 Formal Semantics

Formal semantics mainly focuses on the logical properties of natural language – rules that allow translating the syntax (surface structures) to semantics (logical forms) in a compositional way. For example, the sentences *a monkey eats a banana* is assigned to a logical representation: $\exists m, b, s(monkey(m) \wedge banana(b) \wedge eat(e, m, b))$. However, this representation only concentrates on the logical features expressed by function words (e.g., *and*, *if*) while the meaning of the non-logical features expressed by content words (e.g., *monkey*, *banana*, and *eat*) are mapped to atomic predicate names. The criticism towards this approach is that many natural language phenomena require more knowledge for their resolution than just logical structure. For example, the sentences *a monkey eats a banana* and *a banana attracts a fruit fly* are mapped to the same representation $\exists x, y, e(P(x) \wedge Q(y) \wedge R(e, x, y))$ even though the relation *eat* is semantically different from *attract*. Formal semantics approaches mostly result in semantic parsers (Ovchinnikova, 2012).

3.1.2 Lexical Semantics

Most existing lexical semantic resources can be discussed in terms of the *lexical semantics* meaning representation. Lexical semantics mainly focuses on the organization of lexicons into groups (word senses or verb classes or frames) and the semantic links between these groups (hyponymy,

meronymy, antonymy, causation, inheritance, temporal precedence, etc.). The main paradigms underlining the lexical semantics meaning representation are definition-based model of lexical meaning and prototype-based model of lexical meaning.

Definition Model of Lexical Semantics. The definition-based model involves decomposing lexical meaning into *semantic markers* – atomic units of meaning and conceptualization; the most successful of which is the decomposition of verbs meaning into the thematic roles of its arguments where the arguments have *selectional preferences* – semantic constraints that the verb imposes on its arguments (Ovchinnikova, 2012). A hand-crafted verb resource that is based on this decomposition approach is VerbNet (Kipper et al., 2000). In VerbNet, for example, the verb *give* has an associated "giver" (*Agent* role), the thing that is given (*Theme* role), and the *Recipient* role. The agent and recipient have semantic restrictions that they must either be *animate* or an *organization*. The semantic restrictions are organized in a small hierarchy of around 40 nodes that consist of labels such as *animate*, *organization*, *concrete*, etc. Other verbs with similar set of thematic roles and syntactic realization as *give* such as *lend*, *pass* are put in the same verb class `give-13.1`. The classification of verbs into classes is based on Levin's classification of verbs (Levin, 1993), which is motivated by the notion that verbs that are semantically similar (i.e. in the same verb class) also have similar syntactic realizations. Each verb class has several syntactic frames – possible surface realizations for the verbs in the class. For example, the verb class `give-13.1` has the following syntactic frames: "*Agent* V *Theme* {TO} *Recipient*" (e.g. *They lent a bicycle to me*), "*Agent* V *Recipient Theme*" (e.g. *They lent me a bicycle*), etc. The semantics of each frame is expressed through conjunction of semantic predicates. For example, the syntactic frame "*Agent* V *Theme* {TO} *Recipient*" has the semantics: HAS_POSSESSION(START(E), *Agent*, *Theme*), HAS_POSSESSION(END(E), *Recipient*, *Theme*), TRANSFER(DURING(E), *Theme*), CAUSE(*Agent*, E).

The criticism towards this decomposition approach concerns the difficulty of fixing a universal inventory of thematic roles and the ambiguity in assigning them (Riemer, 2010). At the moment semantic predicates in VerbNet as well as its semantic restrictions are just labels – they are not axiomatized or linked to any formal theory (Ovchinnikova, 2012). An application of VerbNet in NLP concerns usage of its syntactic patterns for mapping verb arguments to appropriate roles (semantic role labeling).

Prototype Model of Lexical Semantics. The prototype-based model of lexical meaning involves representing meaning via a prototype – a structure of concepts underlying lexical meaning, an example of which is Frame Semantics (Fillmore, 1967) that considers lexical meanings to be related to prototypical situations captured by *frames* – structures of related concepts. A hand-crafted lexical semantic resource that is based on this prototype approach and supported by corpus evidence is FrameNet (Ruppenhofer et al., 2006). The lexical meaning in FrameNet is expressed in terms of frames, which are supposed to describe prototypical situations spoken about in natural language. Every frame contains a set of roles corresponding to the participants of the described situation, e.g., *Donor*, *Recipient*, *Theme* for the GIVING frame. In contrast to VerbNet, FrameNet assigns roles not only to verbs, but also nouns, adjectives, adverbs and prepositions. Lexemes belonging to different parts of speech with similar semantics evoke the same frame, e.g., *contribute* and *contribution* both evoke the GIVING frame. FrameNet roles are also more specific than VerbNet and often referring to concrete scenarios e.g., *Donor* instead of *Agent*. Similar to VerbNet, FrameNet also gives the syntactic realization patterns of frame elements e.g., the role *Recipient* in the frame GIVING is most frequently filled by a noun phrase in the indirect object position. In addition however, FrameNet introduces semantic relations between frames e.g., the GIVING and GETTING frames are connected by the causation relation. Their roles are also connected e.g., *Recipient* in GIVING is connected to *Recipient* in GETTING. This feature opens a range of new reasoning options and can also be useful for paraphrasing (Ovchinnikova, 2012).

This prototype model faces the same difficulty as the decomposition approach in fixing a universal inventory of participants and roles of the prototypical situations.

Lexical Semantic Relations. An alternative to having to manually define a universal set of labels or roles (i.e., semantic primitives) to define meaning, this approach represents meaning as a network of relationships between word senses. A hand-crafted lexical semantic resource that is based on this approach is WordNet (Miller et al., 1990). In WordNet, lexical-semantic knowledge is repre-

sented in a network-like structure. Nouns, verbs, adjectives, and adverbs are grouped into synonym sets (synsets) which express word senses. For example, the word "play" is in several synsets ({play, drama, dramatic play}, {play, fun, sport} etc.), each referring to to the different senses. Semantic relations such as hyponymy, meronymy, antonymy, etc. are defined among synsets and among words. Synsets in WordNet however, are often criticized for being too fine-grained to enable automatic word sense classification (Agirre and De Lacalle, 2003).

3.1.3 Distributional Semantics

Distributional semantics' representation of meaning is based on the quotation: "You shall know a word by the company it keeps" (Firth, 1961) – where lexical meaning is obtained from the distributional properties of words: "words which are similar in meaning occur in similar contexts" (Rubenstein and Goodenough, 1965). A lot of automatically constructed lexical resources are developed out of this idea, for example DIRT (Lin and Pantel, 2001) and VerbOcean (Chklovski and Pantel, 2004).

The DIRT (Discovery of Inference Rules from Text) is a collection of paraphrases automatically learned from corpora. The approach is motivated by the hypothesis that a path, extracted from a dependency parse tree, expresses a binary relationship between two nouns. If two paths tend to link the same sets of words then the meanings of the corresponding patterns are similar. For example, the patterns *X wrote Y* and *X is the author of Y* are similar, with some measure of similarity. DIRT contains around 231,000 unique patterns.

VerbOcean contains semantic relationships between verb paraphrases automatically learned by querying the Web with Google for hand-crafted lexico-syntactic patterns indicative of each relation. For example, *Xed * and then Yed* is a pattern for the `happens-before` relation. After verb pairs that have semantic relation are found, DIRT approach is used to extract the paraphrases. The result is for example *X outrage Y* `happens-after`/`is stronger than` *X shock Y*.

Works on learning semantic relations between verbs that go beyond distributional similarity and manually constructed verb co-occurrence patterns can be found in the textual entailment paradigm (Weisman et al., 2012). This work utilizes information from various textual scopes: verb co-occurrence within a sentence, within a document, as well as over-all corpus statistics. They propose a rich novel set of linguistically motivated cues (such as in sentence-level: dependency relations between clauses, tense ordering between co-occurring verbs, sentence level point wise mutual information (PMI) between the verbs, in document level: narrative score based on narrative chain detection in (Chambers and Jurafsky, 2008), and in corpus-level: *typed* distributional similarity), for detecting entailment between verbs and combine them as features in a supervised classification framework. Their experiment over a manually labeled dataset showed that their model that uses co-occurrence signals at multiple levels significantly outperforms several state-of-the-art models both in terms of Precision and Recall.

Formal semantics focuses on logical features of language, focusing mainly on functional words and compositionality while representing content words as atomic predicate names having referential meaning (Ovchinnikova, 2012). Lexical semantics on the other hand mostly ignores the logical aspects of language or compositionality, focusing mainly on the specification of meaning of content words. Distributional similarity represents meaning of words via their distribution and provides account of compositionality by assessing the acceptability of a word combination through the distributional similarity of its components, e.g. *boil a potato* is more acceptable than *boil an idea* because of the higher distributional similarity between *boil* and *potato* as compared to *boil* and *idea* (Ovchinnikova, 2012).

In this thesis, I propose to use insights from lexical and distributional semantics to build a lexical semantic resource on verbs that (1) specify the meaning of verbs by mapping them to relations in knowledge bases and (2) provide some accounts of verbs compositionality by learning the pre-condition and post-condition of their shared arguments using distributional information of the arguments, thus avoiding the need to manually define a set of labels or semantic primitives for defining these pre-conditions and post-conditions. The pre-condition and post-condition of arguments of a verb are represented as word vectors containing verbs, adjectives and nouns surrounding the arguments before and after they participate in the event expressed by the verb – these vectors reflect the

state of the arguments, intuitively representing what the arguments do or what are done to them, how they are described (i.e., their attributes), and what they possess (Bamman et al., 2013).

3.2 Lexical and World Knowledge

We have already discussed how knowledge about the world is crucial for natural language understanding. To discuss possible differences between lexical and world knowledge, I use the illustration from (Ovchinnikova, 2012) that points to the different levels of knowledge relevant for NLU.

1. If *NP* is a noun phrase and *V* is an intransitive verb, then the concatenation *NP V* is a clause.
2. The phrase *x wrote y* corresponds to the proposition *write(x,y)*.
3. The proposition *write(x,y)* refers to a "text creation" event, such that *x* plays the role of *author* and *y* plays a role of *text* in this event.
4. If something is a tragedy then it is a play.
5. The main function of a playwright is writing plays.
6. "Romeo and Juliet" was written by Shakespeare.

Example (1) represents a typical syntactic rule and is included in the grammar of English language. In example (2), a surface realization of the predicate *write* is mapped to its logical form – for example, in the output of a semantic parser. In example (3), the predicate and its arguments are mapped to the "text creation" frame and its thematic roles. This rule can be included in lexical resource such as FrameNet. Example (4) is an example of type-of relation that can be included in lexical resource such as WordNet, while example (5) is a common sense knowledge about playwrights. This can be part of a definition of a category "playwrights" in a knowledge base ontology or learned automatically by an inference engine such as PRA (Lao et al., 2011) over the knowledge base. Example (6) is a specific fact about the world that can be part of a factual ontology containing knowledge about the *write* relation instances (e.g. YAGO).

It is straightforward to see that example (1) and (2) are language dependent and belong to lexical knowledge while example (6) is not. Example (6) is a part of the knowledge about the world. Everything in between (2) and (6) is more difficult to classify (Ovchinnikova, 2012), they are both language dependent and anchored to the world. For a better NLU, it is clear that we need both lexical and world knowledge. Thus, there is a need to bridge the knowledge from (2) to (3) and use inference over a network of linked (3) to (6) to better interpret natural language expression.

An example of lexical semantic resource that is purely textual is ReVerb (Fader et al., 2011). ReVerb automatically extracts relation phrases – phrases that denote binary relations in English sentences, e.g. *made a deal with*. ReVerb operates in an Open IE paradigm; it makes a pass over its corpus – requiring no manual tagging of relations nor any pre-specified relations, identify relation phrases that satisfy the syntactic and lexical constraints and then finds a pair of NP arguments for each identified relation phrase. The resulting extractions are then assigned a confidence score using a logistic regression classifier. The syntactic constraint specifies that every multiword relation phrase must begin with a verb, end with a preposition, and be a contiguous sequence of words. To avoid overly-specific relation phrases, the lexical constraint specifies that a binary relation phrase ought to appear with at least a minimal number of distinct argument pairs in a large corpus.

Strictly speaking, lexical semantic resources provide information about words and not about the world. Although the generalization which these resources give (VerbNet with its verb classes, FrameNet with its frames, WordNet with its synsets, DIRT and VerbOcean with its paraphrases) can be seen as referring to conceptual entities, it is not clear how much inference we can do over them. Reasoning over lexical semantic resources alone has a significant shortcoming in that they imply too little structure (Ovchinnikova, 2012). In most cases, semantic relations defined in these lexical resources are just two-place predicates: *relation_name*($word_1$, $word_2$) (where *relation_name* is for example semantic relations such as hyponymy, antonymy, causation, inheritance, etc.) that are difficult to use for defining complex relationships like the fact that a PERSONAFRICA (a NELL's category) is an person who has citizenship of a country that is located in Africa. Lexical semantic relations seem to be not enough for representing detailed world knowledge. A purely textual resource such as ReVerb that lacks generalization is even more limited in the reasoning that can be done over it. For example, given a sentence from a document: *Shakespeare's Romeo and Juliet is*

a tragedy, reasoning over example (4) - (6) enables us to infer that *Shakespeare* is therefore a *playwright*. This additional information can be useful for understanding other parts of the document, which reasoning over text alone may not allow us to do.

In contrast to the lexical semantic resources, knowledge bases are designed for conceptualizing the world: its entities and the more complex relationships. However, being built up with lexemes makes lexical-semantic resources more applicable in NLP, while knowledge bases require an additional lexical interface to be mapped to linguistic structures (Ovchinnikova, 2012). There is a need therefore to map the lexical semantic resources to knowledge bases containing conceptual representation of the world to facilitate deep inference over the data for better natural language understanding (Soderland et al., 2010).

4 Proposed Approach

At this point we have covered the related work for lexical semantic resources and the motivation for anchoring to world knowledge for better natural language understanding. In this section, I propose an approach for constructing a lexical semantic resource on verbs for better natural language understanding. I will also outline my proposed methods for illustrating and evaluating the usefulness of this constructed resource in terms of (1) reading natural language text (2) temporal scope detection, and (3) ontology extension of knowledge bases. To read a natural language document, I propose a NLU system that uses the mapping of verbs to KB relations and the temporal chains of verbs in the constructed resource, which are either learned from corpora or inferred from the pre-condition and post-condition of the verbs, to identify the relation expressed by a verb in this document jointly with other temporally related verbs in the document. I will evaluate the precision and recall of this NLU system that identify relations expressed by verbs in the document jointly instead of in isolation. Perhaps more interestingly, if we can evaluate the usefulness of the constructed resource for predicting events, for doing event-focused summarization – providing a single coherent interpretation of events from a collection of observations, or for inferring unstated state changes resulting from events in the document. I will also outline the usefulness of the pre-condition and post-condition of change-of-state verbs for temporal scope detection (i.e., to temporally scope facts of relations expressed by these verbs). Finally, I will outline a method for using temporal chain of verbs in the constructed resource to extend the ontology of relations in knowledge bases.

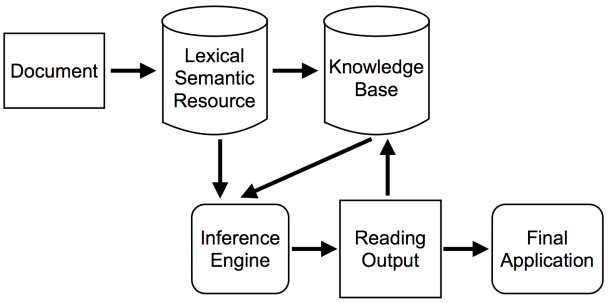

Figure 1: The architecture of the proposed work.

The architecture of my proposed work is illustrated in Figure 1. To build a better NLU system, I propose to construct a lexical semantic resource on verbs that can act as an interface between the natural language text (document) that is in the lexical domain and the knowledge base that is in the conceptual domain. I propose to do this by learning the mapping between the verb lexicons and the knowledge base relations. I propose to also learn the pre-condition and post-condition of change-of-state verbs and include them in the constructed resource. Temporal chains of verbs that are learned directly from corpora or inferred from these pre- and post-condition of the verbs are also included. The constructed resource and the knowledge base are then fed into an inference engine – a NLU system that performs inference over the linked lexical resource and the knowledge base to produce a reading output: a disambiguation of verbs in the document to relations in the knowledge base and the temporally relevant chain of verbs in the document (details in Figure 2). This output can be used

for further application such as summarization or question answering, and also for knowledge base completion: extraction of new instances or extension of its relation ontology.

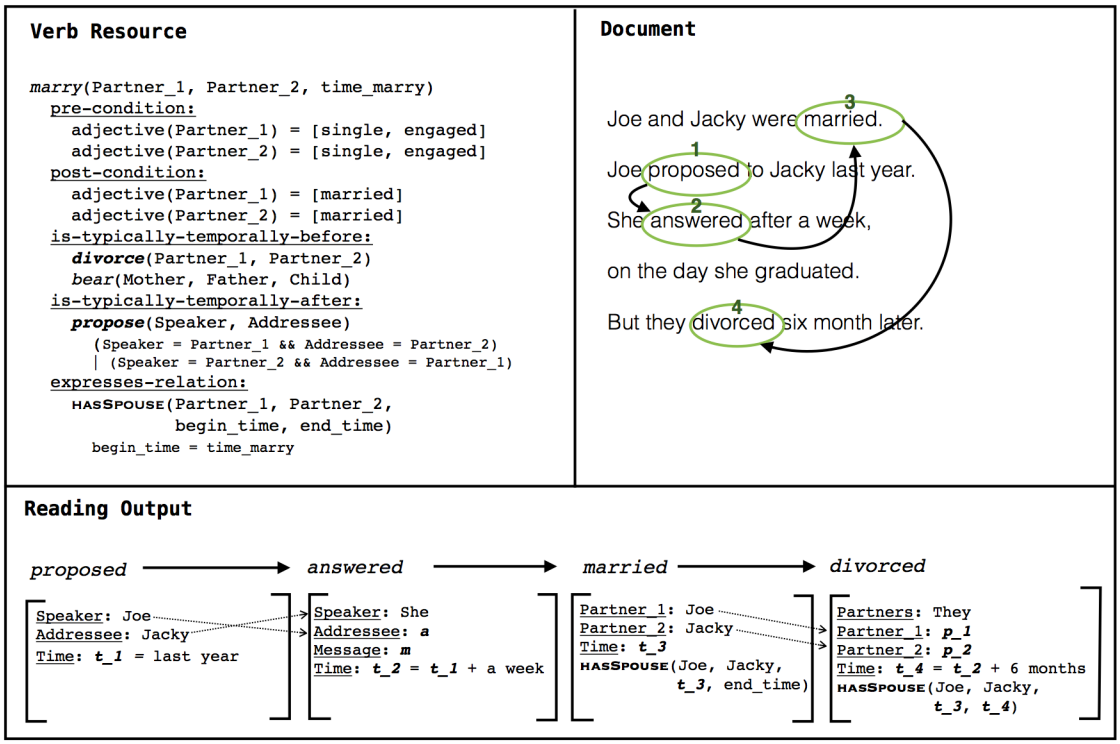

Figure 2: Details of the proposed verbs resource and reading output for a toy document.

Figure 2 details the proposed verbs resource and the desired reading output on a toy example of a document. For each verb in the resource, whenever available, the pre- and post-condition of its arguments (semantic roles) are listed. The pre- and post-condition of each argument are word vectors containing the frequency of verbs, adjectives, and nouns co-occurring with an argument that express the state of the argument before and after it participates in the event expressed by the verb. These verbs, adjectives and nouns in the vectors intuitively represent respectively what the argument does/what are done to it i.e., the verbs for which the argument is the *Subject*/*Object*, how it is described (i.e., its attributes), and what it possesses. For example, the pre-condition of *Partner_1* of the verb *marry* is a vector containing the adjectives *single* and *engaged*, while the post-condition is a vector containing the adjective *married*. Note that some post-condition vectors can contain the words in the pre-condition vectors but with a decrease in frequency. For example, the post-condition of *marry* can contain the word *single* at a decreased frequency. The change in frequency can be used to detect incidence of state change.

For each verb in the resource, whenever available, verbs that typically follow/precede it are also listed. These can be learned from the corpora (using narrative script finding or textual entailment methods) or inferred from the pre- and post-condition of verbs. For example, in Figure 2, the verbs in bold: *divorce* and *propose* can be inferred to follow/precede the verb *marry* from the pre- and post-condition of their arguments. For example, *divorce* necessitates that its arguments have the **pre**-condition of being *married* (i.e., a couple has to be married before they can get divorced), and *married* is the **post**-condition of *marry*. Similarly, *propose* has the **post**-condition that its arguments are often *engaged* after the proposal, and *engaged* is often the **pre**-condition of *marry*. On the other hand, other verbs like: *bear* are verbs that often follow/precede *marry* in the corpora without requiring that their arguments' pre- and post-condition match *marry*'s post- and pre-condition, respectively. Whenever verbs are temporally linked, the link between their semantic roles are also added when available. For example, *Partner_1* and *Partner_2* in *marry* are linked to *Partner_1* and *Partner_2* in *divorce* and are linked to either *Speaker* or *Addressee* in *propose*. The mapping of the

verb and the KB relation it expresses is also included. For example, the verb *marry* is mapped to HASSPOUSE relation and the time of *marry* is mapped to the **begin_time** of HASSPOUSE i.e., the verb *marry* signal a creation of a HASSPOUSE instance over entities expressed by *marry*'s arguments.

Given a document, the proposed verbs resource, and a knowledge base, the desired reading output is also shown in Figure 2. The desired output contains (1) temporally relevant verbs in the document and their temporal sequence (the solid arrowed line on the document), (2) for each verb, the values of their semantic roles – read directly from their syntactic realizations in the document or inferred using background knowledge, (3) for each verb, the (disambiguated) relation that it expresses.

For example, although the verb *married* is mentioned first in the document, its correct temporal sequence placement (after *answered* and before *divorced*) is outputted. This can be inferred from the background knowledge (temporal chains of verbs). Other, temporally irrelevant, verbs in the document are not included in this chain. For example, the verb *graduated* is not included. This can be useful for filtering out irrelevant events for **event-focused summarization**, for example. The temporal chain is useful for **disambiguating verb sense or the relation expressed by a verb** in the context of other verbs in the document. For example, *proposed* has many senses, but in the context of this document, it is temporally linked to the verb *married*. We can therefore identify the correct sense of *proposed* as in "to make an offer of marriage". The temporal chain is also useful for **temporal scoping**. For example, the time of each verb can be inferred from this chain: the time of *married*, ***t_3*** is sometime after ***t_2*** and before ***t_4***. The time of the expressed relation can also be inferred. For example, *marry* creates an instance of the relation HASSPOUSE (i.e., the time of *marry* is the **begin_time** of this relation instance) while *divorce* deletes this instance (i.e., the time of *divorce* is the **end_time** of this instance). To find the time ***t_3*** of *marry*, in a "macro"-reading fashion, we can use the pre- and post-condition of *marry* to find the time at which documents mentioning either *Partner_1* or *Partner_2* have their words surrounding these entities change from the words in the pre-condition (e.g., *single*, *engaged*) to the words in the post-condition (e.g., *married*). The time at which this change occurs can be used to infer ***t_3***.

Temporal chain outputs from documents can also be aggregated and fed back to the knowledge base to **extend its relation ontology**. For example, if we always see a group of verbs (expressing a relation in the knowledge base) being followed temporally by another group of verbs (not yet expressing any relation in the knowledge base), we can add a new relation in the knowledge base expressed by this group of verbs.

The values of semantic roles can be inferred from the background knowledge. For example, using links between semantic roles in the proposed resource, we can infer that the *Speaker* of *answered* is therefore "Jacky", as *Speaker* is linked to the *Addressee* of the previous verb *proposed*. This can be useful for **semantic role labeling** and **coreference resolution**. Since the *Speaker* of *answered* is referred to in the document as "She", we can also infer that "Jacky", an ambiguous name, is therefore a female. The value of *Message* in *answered* can be inferred to be "yes", if we have in our proposed resource the verb "*answer yes*" as the pre-condition of *marry*.

4.1 Constructing a Resource on Verbs

I propose to construct a verb resource that contains (1) the mapping between verbs and relations in knowledge bases, (2) the pre-condition and post-condition of change-of-state verbs, (3) the temporal sequence of verbs. The mapping will enable the anchoring from surface text to conceptual knowledge about the world. The pre-condition and post-condition of verbs will open up interesting opportunities for inferencing and reasoning – e.g., for inferring temporal sequence of verbs or for inferring relationships between semantic roles of temporally linked verbs. The temporal sequence of verbs can improve inference and understanding of the relation expressed by a verb in the context of other verbs in the document. Furthermore, (1), (2) and (3) will also be useful to temporally scope facts of relations in the knowledge bases.

4.1.1 Mapping of Verbs to Relations in Knowledge Bases

Preliminary results for mapping verbs to relations in the knowledge bases are found as a by-product of aligning ontologies of two knowledge bases that share few or no data entries in common (Wijaya et al., 2013). Given two knowledge bases to align, we can use a corpus statistics from a large corpora such as the Web as an *interlingua* to link the two. For example, using a corpus statistics of 600

million Subject-Verb-Object (SVO) triples from the entire ClueWeb (Callan et al., 2009) corpus of about 230 billion tokens, we can create a graph linking the two knowledge bases (Figure 3). Our approach, called PIDGIN, then performs inference over this graph to determine that KB_1:BORNIN is equivalent to KB_2:PERSONBORNINCITY: PIDGIN first associates the relation nodes from KB_1 with seed labels i.e., self-injection. Starting with this seed information, a graph based semi-supervised learning (SSL) algorithm (Talukdar and Crammer, 2009) is used to propagate these relation labels and classify the rest of the nodes in the graph. One of the advantages of this approach for alignment is that it takes the graph structure (specified by the ontologies of resources to be aligned) and transitivity into account.

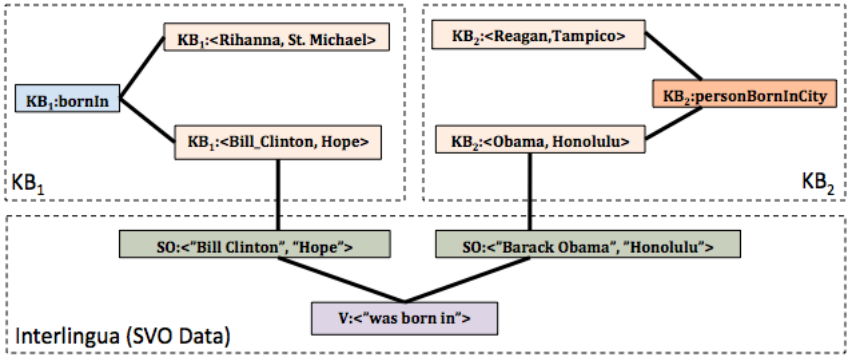

Figure 3: Graph construction using SVO as interlingua to link the knowledge bases to be aligned.

As a by product of label propagation on the graph, each verb and NP-pair node in the graph (i.e., the *V:* and *SO:* nodes in Figure 3) will be assigned scores for each relation label. Exploiting these scores, we can estimate the probability that a verb v represents a relation r as $P(v|r) \approx \frac{\hat{Y}(v,r)}{\sum_{v'} \hat{Y}(v',r)}$, where $\hat{Y}(v,r)$ is the score of label r assigned to verb node v. Since a verb may represent different relations depending on the NP-pair with which it co-occurs e.g., the verb *enter* has different meaning when it appears with an NP-pair $\langle$*Paul*, *room*$\rangle$ from when it appears with an NP-pair $\langle$*John*, *American Idol*$\rangle$; when estimating $P(v|r)$ we also take into account the scores of r on the NP-pair nodes $\langle NP_1, NP_2 \rangle$ with which verb v co-occurs. Similar as before, $P(v|r) \approx \frac{\hat{Y}(v,r)}{\sum_{v'} \hat{Y}(v',r)}$. But now we measure $\hat{Y}(v,r) = \sum_{T_v} \hat{Y}(T_v, r)$, where T_v is a SVO triple $\langle np_1, v, np_2 \rangle$, and where $\hat{Y}(T_v, r) = \hat{Y}(\langle np_1, np_2 \rangle, r) * \hat{Y}(v,r)$. We multiply this estimate with the tf-idf score of the verb, which is proportional to the number of times a verb appears for a relation, but is offset by the total number of times the verb appears with all the relations. This helps to reduce the effect of common verbs such as *is*, *become* that represent many relations.

Using this scoring, for each relation we can return a ranked list of verbs that represent the relation. Some of the verbs returned are shown in Table 1. As we can see in Table 1, the system is able to distinguish verbs representing the relation */medicine/medical_treatment/side_effects*: "*exacerbate*", "*can cause*" from the verbs representing the antonym relation *drugPossiblyTreatsPhysiologicalCondition*: "*relieve*", "*can help alleviate*" even when the two relations have the same domain (*drug*) and range (*physiological condition*). The system is also able to recognize the directionality of the relation. For example, for the relation *_acquired*, which represents the inverse of the relation *acquired* (as in company X acquiring company Y); the system is able to return the correct verbs: *_bought* and *_purchase*, which are the inverse forms of the verbs *bought* and *purchase* (as in *is bought by* and *is purchased by*). Of practical importance is the fact that PIDGIN can learn verbs representing relations in knowledge bases whose instances are created manually or extracted via carefully constructed regular-expression matching (e.g., Freebase and YAGO). We can for example use these verbs to then automate an extraction process for these knowledge bases.

Knowledge Base	Relation	Learned Verbs
Freebase	/sports/sports_team/ arena_stadium	played at, played in, defeated at, will host at, beaten at
	/medicine/medical_ treatment/side_effects	may promote, can cause, may produce, is worsen, exacerbate
NELL	drugPossiblyTreats PhysiologicalCondition	treat, relieve, reduce, help with, can help alleviate
	politicianHoldsOffice	serves as, run for, became, was elected
Yago2	actedIn	played in, starred in, starred, played, portrayed in
	isMarriedTo	married, met, date, wed, divorce

Table 1: Examples of relation-verb pairs automatically learned by PIDGIN. Although we use verb stems in experiments, for better readability, we present the original non-stemmed forms of the same verbs above.

4.1.2 Learning Pre-condition and Post-condition of Change-of-State Verbs

From my previous work (Wijaya and Yeniterzi, 2011), we discover that we can automatically identify changes that occur to an entity based on the changes in the words surrounding the entity over time. By clustering the words that surround the entity over time, we identify *when* (at which year) changes occur, and also *what* changes occur (i.e., what clusters of words are in transition). We also find that for the entities we test, the period that our method identifies coincides precisely with events that correspond to the change. For example, for the entity *Iran*, our approach is able to identify the country's transition from a monarchy to an Islamic republic with a new cluster consisting of words such as *republic* and *revolution* emerging around the entity after 1978 (1979 is the year of the Islamic revolution). Another similar example can be seen with the entity *Kennedy*. Our approach was able to identify the John F. Kennedy the *senator* before the election (one cluster of words surrounding the entity) and the John F. Kennedy the *president* after the election (another cluster of words surrounding the entity). The transition between the two clusters is at 1961, the exact year Kennedy was elected. Similar changes are observed for the entity *Clinton*, from *governor* to *president*. The transition occurs at 1993, the exact year he was elected. We obtain words surrounding our entities over time (with the year granularity) from Google Books NGram dataset (Michel et al., 2011). This dataset contains 1-gram to 5-gram extracted from the text of around 5 million digitized books and how often these n-grams were used over time: their match counts, page counts and volume counts each year; with the year ranging from as early as the 1500s and as late as 2008. In general for this work, we can use any corpus that contains documents labeled with their date creation times at any granularity e.g., GigaWord (Graff et al., 2003).

We learn two important insights from this work: (1) that some changes occurring to an entity can be identified from the changes in the words surrounding the entity over time and (2) that some changes occurring to an entity can coincide with events happening to the entity. I want to extend this work by learning for change-of-state verbs (i.e., verbs that express events and change the state of their arguments), the change caused by these verbs on their arguments. Specifically, I want to learn the pre-condition and post-condition of the arguments i.e., the entities participating in the events expressed by these verbs.

Inspired by this work (Wijaya and Yeniterzi, 2011), I hypothesize that the change happening to entities participating in a change-of-state event can be identified from the change in the words surrounding the entities over time. The state of an entity at a particular time is therefore represented by a vector of words and their counts surrounding the entity at that particular time. The vector can contain verbs, adjectives or nouns surrounding the entity, which intuitively represent what the en-

tity can do/what has been done to it, what it is described as (its attributes), and what it possesses, respectively. The words surrounding an entity at a particular time can be extracted from a corpus of time-stamped documents such as Google Books NGram dataset (Michel et al., 2011) or GigaWord (Graff et al., 2003), from news documents, or from the entity's Wikipedia page edit history.

Given a change-of-state verb and a dozen or so seed instances of the event expressed by this verb, we can learn the *pre*-condition (and *post*-condition) of the verb's arguments by aggregating the word vectors of entities expressed by the arguments *before* the event happens (and *after*, respectively). For example, for the verb *marry*, we can give a dozen or so instances of the event expressed by *marry*: e.g., *marry*(Kanye_West, Kim_Kardashian, 24 May 2014), *marry*(John_Legend, Christine_Teigen, 14 September 2013), etc. The *pre*-condition of *marry* is then an aggregate of word vectors of the entities: Kanye_West, Kim_Kardashian, John_Legend, Christine_Teigen *before* the marriage event happens to them i.e., on 23 May 2014 for Kanye_West and Kim_Kardashian and on 13 September 2013 for John_Legend and Christine_Teigen. Similarly, the *post*-condition of *marry* is an aggregate of the word vectors of these entities *after* the marriage event happens to them i.e., on 25 May 2014 for Kanye_West and Kim_Kardashian and on 15 September 2013 for John_Legend and Christine_Teigen.

There are many ways to aggregate the word vectors. We can do a simple sum or average of the word vectors and choose the top-k words of the sum/average vector to represent the pre-condition (or post-condition), or we can cluster the words surrounding the entities (Bamman et al., 2013) and select the cluster of words common to these entities to represent the typical pre-condition and post-condition of these entities *before* and *after* the event expressed by the verb happens.

4.1.3 Learning Temporal Sequence of Verbs

To learn temporal sequence of verbs, we can use the method for extracting narrative chain of verbs in a document by following a protagonist argument that is shared among the verbs (Chambers and Jurafsky, 2008). These narrative chains of verbs over a large collection of document can then serve as probabilistic training signals for learning the typical temporal order of verbs. In my work (Wijaya et al., 2012), using the hypothesis that the narrative order of verbs correlates with the temporal order of relations they express, we show how we can successfully use a graph-based label propagation algorithm that takes transitivity of temporal order into account and statistics on the narrative order of verbs over a large document collection to infer typical temporal order of relations expressed by these verbs. We can use this algorithm for inferring the typical temporal sequence of verbs based on statistics on the narrative order of verbs over a large document collection.

Another way for learning typical temporal sequence of verbs is through learning verb inference rules (Weisman et al., 2012). This method utilizes information from various textual scopes: sentence, document and overall corpus statistics' verb co-occurrence to develop linguistically motivated features for detecting entailment between verbs. We can use this method for detecting a specific type of entailment i.e., temporal between verbs to extract typical temporal sequence of verbs.

Yet another way for learning typical temporal sequence of verbs is to start with a dozen or so entities that we know experience a common chain of events. We can then cluster the verbs surrounding the entities over time using methods such as topic over time (Wang and McCallum, 2006) (or its variation (Bamman and Smith)) – an LDA-like topic model that explicitly models time jointly with word co-occurrence patterns, generating both cluster of words and timestamps – to learn typical temporal sequence of verb clusters.

We can also infer typical temporal sequence of verbs by chaining the verbs based on matching typical pre-condition to typical post-condition of the verbs that we have learned.

4.2 Evaluation

As described previously, in this section I will outline my proposed methods for evaluating the usefulness of the proposed verbs resource in terms of (1) reading natural language text (2) temporal scope detection, and (3) ontology extension of knowledge bases.

4.2.1 Evaluation by Reading

To read a natural language document, I propose a NLU system that uses the mapping of verbs to KB relations and the temporal chains of verbs in the constructed resource to disambiguate the relation expressed by a verb in the document based on the presence of other temporally related verbs in the document. I will evaluate the precision and recall of this NLU system for identifying relations expressed by verbs in the document, compared to other NLU systems that read sentences in isolation.

The proposed NLU system is based on the theory of stances that human readers take in relation to the text during reading and interpretation of text (Langer, 1990): (1) being out and stepping into an envisionment, (2) being in and moving through an envisionment, (3) stepping back and rethinking what one knows, (4) stepping out and objectifying the experience. These stances are recursive; they can recur at any point in the reading where the reader's envisionment – a personal text- and conceptual- world embodying all that the reader understands up to that point in the reading – is updated. The final envisionment is therefore not the sum of previous traces, but is instead an evolving whole. The first stance: "being out and stepping into an envisionment" requires readers to **anchor** the text to their world using prior knowledge and surface features of the text in order to begin to construct an envisionment. In the second stance: "being in an moving through an envisionment", readers use their constructed envisionments, prior knowledge and the text to make **inferences** to further their creation of meaning. In the third stance: "stepping back and rethinking what one knows", readers used their envisionments of the text to rethink and **update** their prior knowledge. In the last stance: "stepping out and objectifying the experience", readers distanced themselves from their environments, reflecting and reacting to the content, to the text, or to the reading experience.

Based on stances (1) and (2) of this "reading as interpretation" theory, I propose an NLU system that (1) anchors verbs in text to relations in knowledge bases to build the *envisionment*, (2) does an inference over the linked text- and conceptual- world (i.e., the *envisionment*) to produce a reading output.

To anchor verbs in text to relations in knowledge bases, I propose to use the learned mapping from verbs to relations that is discussed in section 4.1.1. The anchoring and resulting envisionment is illustrated in Figure 4. The dashed lines indicate the mapping from text to the lexical resource and the (learned) mapping from the lexical resource to the knowledge base. The solid lines indicate typical (learned) temporal sequences between verbs in the lexical resource and typical (learned) temporal sequences between relations in the knowledge base.

As can be seen in Figure 4, a verb in the lexical resource (e.g., *play*) can be mapped to several relations in the knowledge base (e.g., TEAMPLAYSAGAINSTTEAM or MOVIESTARACTOR). Fortunately the presence of other verbs in the document i.e., *direct* and *produce* can help disambiguate *play* in the context of the document as more likely of expressing the relation MOVIESTARACTOR than TEAMPLAYSAGAINSTTEAM. Indeed, in Figure 4, we can see that the network of relation nodes related to movie (i.e., those shaded in grey) is more densely connected than the network of nodes related to sports team and game. If we hypothesize that edges between nodes in Figure 4 have a generic "related to" semantics, then we can use algorithms such as PageRank for ranking the importance of nodes in the graph based on their connectedness. Starting from initial PageRank values of 1 assigned to all the verbs in the constructed envisionment, PageRank can compute the rank values of other nodes in the graph. We can then select for each verb, the relation that is ranked highest for that verb as the relation that the verb most likely expresses given the constructed envisionment e.g., for *play* select MOVIESTARACTOR as the relation that it most likely expresses.

Since the relations expressed by verbs depend on the type of the arguments (e.g., the verb *play* expresses different relations when used with different argument types – *play* expresses NELL relation TEAMPLAYSAGAINSTTEAM when used with argument types *Sportsteam* as its subject and *Sportsteam* as its object; and expresses ACTORSTARREDINMOVIE when used with argument types *Actor* as its subject and *Movie* as its object), we can try to identify types of arguments in text. Similar to how we disambiguate verbs in text to relations in knowledge base, we first anchor the arguments to entities in the knowledge base. The anchoring and resulting envisionment is illustrated in Figure 5. The dashed lines are the mapping from surface text to possible entities in the knowledge base whose noun phrase mentions match the noun phrase in text. The solid lines indicate that there are some relationships between the linked entities i.e., two entities are linked when they are arguments of some relation instances in the knowledge base.

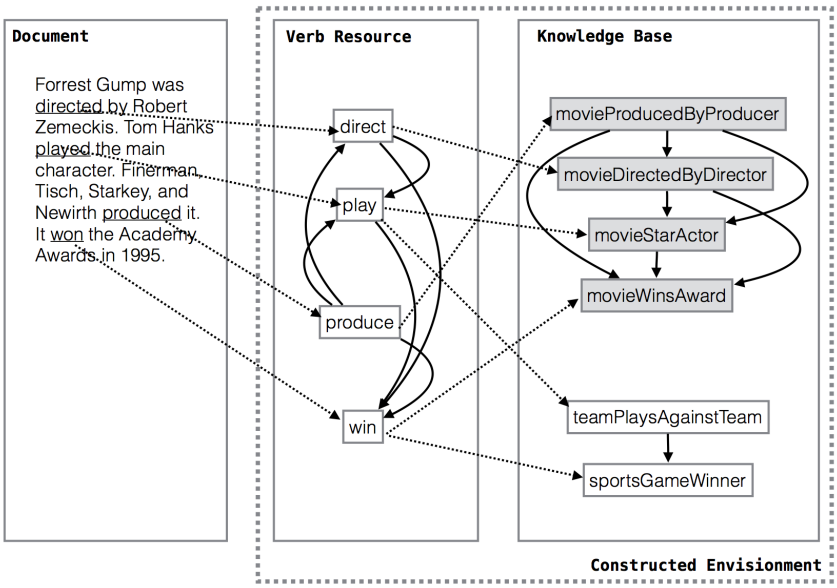

Figure 4: Constructing envisionment by linking verbs in text to verbs in lexical resource and relations in knowledge base.

As can be seen in Figure 5, an argument in text (e.g., "*Tisch*") can belong to several possible entities in the knowledge base (e.g., *Tisch (university)* or *Tisch (filmProducer)*). Fortunately the presence of other arguments in the document i.e., "*Tom Hanks*", "*Forrest Gump*", "*Academy Awards*" can help disambiguate "*Tisch*" in the context of the document as more likely to be *Tisch (filmProducer)* than *Tisch (university)*. Indeed, in Figure 5, we can see that the network of nodes to which *Tisch (filmProducer)* belongs (i.e., those shaded in grey) is more densely connected in the envisionment than the network of nodes to which *Tisch (university)* belongs. If we hypothesize that edges between nodes (i.e., solid lines) in the constructed envisionment of Figure 5 have a generic "related to" semantics, then we can use algorithms such as PageRank for ranking the importance of nodes in the graph based on their connectedness. Starting from initial PageRank values of 1 assigned to entities in the constructed envisionment, PageRank can compute the final rank values of the nodes in the graph. We can then select for each argument in text, the entity in the knowledge base that is ranked highest for that argument as the entity that the argument most likely represents given the constructed envisionment e.g., for "*Tisch*" select *Tisch (filmProducer)* as the entity that it most likely represents.

We can conduct these two PageRank computations iteratively, using the computed rank values of one to update the envisionment graph of the other: the first PageRank computes rank values of relations, the second computes rank values of entities. We can weigh edges in the second graph (i.e., the envisionment graph of Figure 5) with the rank values of relations that link these entities. Similarly, we can weigh edges that map verbs to relations in the first graph (i.e., the envisionment graph of Figure 4) as a function of the rank values of entities expressed by the verbs' arguments; these values provide some signals as to which relation is more likely for a verb given the *types* of entities expressed by its arguments.

4.2.2 Evaluation for Temporal Scoping

In this section, I outline a method for evaluating the usefulness of the pre-condition and post-condition of change-of-state verbs for temporal scope detection (i.e., to temporally scope (find the start time and end time of) facts of events expressed by these verbs) (Wijaya et al., 2014). To start off, we first define a *Contextual Temporal Profile* (CTP) of an entity at a given time point t as a vector that contains the context within which the entity is mentioned at that time. Temporal scoping based on CTP, pre- and post-condition of change-of-state verbs is then based on these related insights: i) the context of the entity at time $t-1$ reflects the pre-condition of the entity before the event that

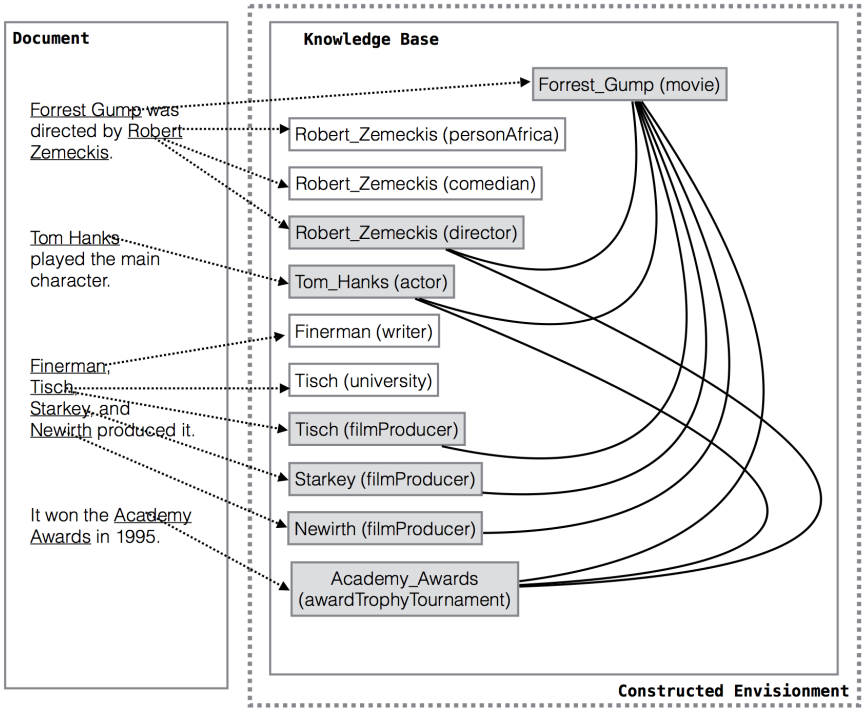

Figure 5: Constructing envisionment by linking arguments in text to entities in in knowledge base.

changes its state happens. ii) the context of the entity at time t reflects the event happening to the entity and the state (post-condition) of the entity at time t. iii) the *difference in context* before, at time $t-1$, and after, at time t, reflects the associated state change brought about by the event happening at time t. However an entity can undergo a multiplicity of events, hence state changes at the same time. Thus both the contexts and the differences in contexts can contain information pertaining to several events (and the associated state changes). We therefore need a way of determining which part of the context is relevant to a given state change sc_i. To this end, we generate what we refer to as an $aggregate$ state vector, $Vs(\overline{e}, sc_i)$ for a hypothetical average entity $\overline{e}$ undergoing state change sc_i. We generate $Vs(\overline{e}, sc_i)$ from the CTPs of a seed set of entities at the time they undergo state change sc_i.

Stage 1: Learning State and State Change Vectors To build CTPs for entities, we use two time-stamped corpora: the Google Books Ngram corpus (Michel et al., 2011); and the English Gigaword (Graff et al., 2003) corpus. The Google Books Ngram corpus contains n-grams for $n = 1-5$; along with occurrence statistics from over about 5 million digitized books. The English Gigaword (Graff et al., 2003) corpus contains newswire text from 1994-2008. From these corpora, we use the time granularity of a year as it is the finest granularity common to both corpora.

Definition 1 (Contextual Temporal Profile) *The Contextual Temporal Profile (CTP) of an entity e at time t, $C_e(t)$, consists of the context within which e is mentioned. Specifically $C_e(t)$ consists of uni-grams and bi-grams generated from the 5-grams(Google Books Ngram) or sentences (Gigaword) that mention e at time t.*

Notice that the CTPs can contain context units (bi-grams or uni-grams) that are simply noise. To filter the noise, we compute *tf-idf* statistics for each contextual unit and only retain the top k ranking units in $C_e(t)$. In our experiments, we used $k = 100$. We compute *tf-idf* by treating each time unit t as a document containing words that occur in the context of e (Wijaya and Yeniterzi, 2011).

Furthermore, CTPs may contain context units attributed to several events (and the associated state changes). We therefore tease apart the CTPs to isolate contexts specific to a given state change. For

this, our method takes as input a small set of seed entities, $\mathcal{S}(sc_i)$, for each type of state change. Thus for the US presidency state change that denotes the event: beginning of a US presidency, we would have seeds as follows: *(Richard_Nixon, 1969)*, *(Jimmy_Carter, 1977)*. From the CTPs of the seeds for state change sc_i, we generate an aggregate state vector, $Vs(\overline{e}, sc_i)$.

Definition 2 (Aggregate State Vector for $\overline{e}$) *The aggregate state vector of a mean entity $\overline{e}$ for state change sc_i, $Vs(\overline{e}, sc_i)$, is made up of the contextual units from the CTPs of entities in the seed set $\mathcal{S}(sc_i)$ that undergo state change sc_i. Thus, we have: $Vs(\overline{e}, sc_i) = \frac{1}{|\mathcal{S}(sc_i)|} \sum_{e,t:(e,t)\in\mathcal{S}(sc_i)} C_e(t)$.*

Thus, the state vector $Vs(\overline{e}, sc_i)$ reflects events happening to $\overline{e}$ and the state of $\overline{e}$ at the time it undergoes the state change sc_i. Additionally, we compute another type of aggregate vector, *aggregate change vector* $\triangle Vs(\overline{e}, sc_i)$ to capture the change patterns in the context units of $\overline{e}$. Context units rise or fall due to state change, as words increase or decrease in their frequency of mentions around the entity due to the event happening to the entity.

Definition 3 (Aggregate Change Vector for $\overline{e}$) *The aggregate change vector of a mean entity $\overline{e}$ for state change sc_i, $\triangle Vs(\overline{e}, sc_i)$, is made up of the change in the contextual units of the CTPs of entities in the seed set $\mathcal{S}(sc_i)$ that undergo state change sc_i. Thus, we have: $\triangle Vs(\overline{e}, sc_i) = \frac{1}{|\mathcal{S}(sc_i)|} \sum_{e,t:(e,t)\in\mathcal{S}(sc_i)} C_e(t) - C_e(t-1)$.*

The aggregate state vector $Vs(\overline{e}, sc_i)$ and the aggregate change vector $\triangle Vs(\overline{e}, sc_i)$ are then used to detect state changes.

Stage 2: Detecting State Changes To detect events and therefore state changes happening in a previously unseen entity e_{new}, we generate its state vector, $C_{e_{new}}(t)$, and its change vector, $\triangle C_{e_{new}}(t)$ = $C_{e_{new}}(t)$ - $C_{e_{new}}(t-1)$, for every time point t. We consider every time point t in the CTP of the new entity to be a candidate for a given state change sc_i, which we seek to determine whether e_{new} goes through and at which time point. We then compare the state vector and change vector of every candidate time point t to the aggregate state and aggregate change vector of state change sc_i (we use cosine similarity). The highest ranking candidate time point (most similar to the aggregate state and aggregate change vector) is considered to be the start time of an event that has the associated state change sc_i for the new entity e_{new}.

We have conducted experiments (Wijaya et al., 2014) to obtain preliminary results of the usefulness of pre- and post-condition of change-of-state verbs for temporal scope detection. In the experiments, we seek to answer the following questions: (1) Is treating temporal scoping as state change detection in *Contextual Temporal Profile* (CTP) effective? (2) Do Contextual Temporal Profiles help improve temporal scope extraction over plain temporal profiles? We answer these questions by comparing to the state-of-the-art system in this area, our previous work CoTS (Talukdar et al., 2012), and to all the baselines against which CoTS was compared. CoTS, like CTPs, leverages temporal profiles, but does not make use of context. We evaluate on the same set of facts as CoTS: facts from the US Administration domain (US President, US Vice President, and US Secretary of State); and facts from the Academy Awards domain (Best Director and Best Picture). The number of facts per relation are as follows: US President, 9; US Vice President, 12; US Secretary of State, 13; Best Director, 14; and Best Picture, 14.

Similar to CoTS, the datasets from which the CTPs were generated are as follows: The Google Books Ngram (1960-2008) dataset (Michel et al., 2011) for the US Administration domain and the English Gigaword (1994-2008) dataset (Graff et al., 2003) for Academy Award domain.

To compute precision we used cross validation, in particular, leave-one-out cross validation due to the small number of facts per relation. We predict the *begin* time of each fact, the time the fact starts to be valid. True begin times were determined by a human annotator. This human annotated data formed the gold-standard which we used to determine Precision (P), Recall (R), and the F1 measure. All evaluations were performed at the year level, the finest granularity common to the two time-stamped datasets.

For our first experiment, we report the average precision@*k*, where *k*=1 to *n*, where *n*=47 is the number of years between 1960 to 2008 to select from. As can be seen in Figure 6, precision quickly reaches 1 for most relations. The true begin time is usually found within top *k*=5 results.

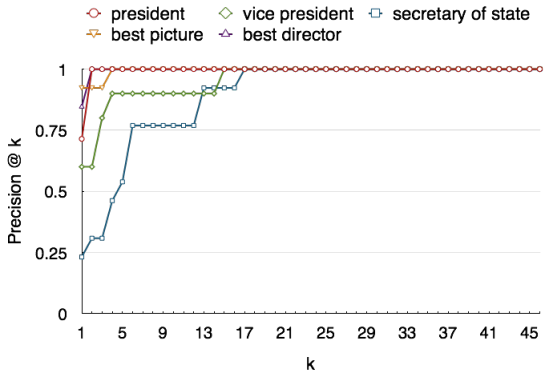

Figure 6: Precision @ k using Contextual Temporal Profiles.

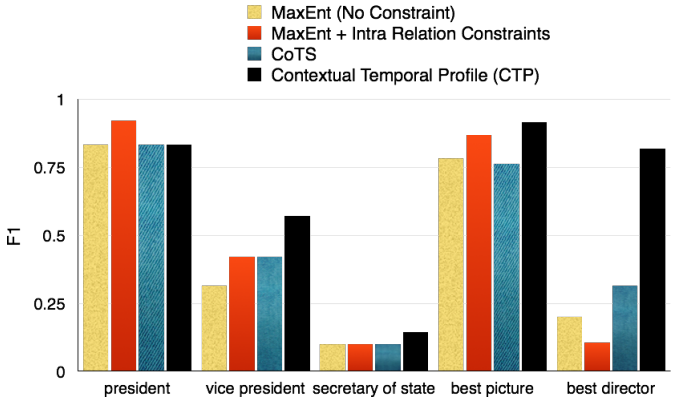

Figure 7: Comparison of F1 scores with CoTS and other baselines.

For our second experiment, we compared to the F1 scores of CoTS and other baselines in (Talukdar et al., 2012). As can be seen in Figure 7, our CTPs approach gives comparable or better F1 (@k=1) than systems that use only plain temporal profiles, even when these systems are supplemented with many carefully crafted, hand-specified constraints.

We also observe that the uni-grams and bi-grams in the train CTPs and change vectors reflect meaningful events and state changes happening to the entities (Table 2). For example, after 'becoming president' and 'taking office', US presidents often see a drop in mentions of their previous (job title state) such as 'senator', 'governor' or 'vice president' as they gain the'president' state. Overall, our results show that using change-of-state detection i.e., how the pre-condition of an entity is changed to its post-condition by an event, is promising for detecting the *begin* time of the event. In its current state however, the method performs poorly on inferring *end* times as contexts relevant to an event often still mentioned with the entity even after the event long passed. For example, the entity *Al Gore* is still mentioned a lot with the bi-gram 'vice president' even after he is no longer a vice president. Prior work, CoTS, inferred *end* times by leveraging manually specified constraints, e.g., that there can only be one vice president at a time: the beginning of one signals the end of another (Talukdar et al., 2012). However such methods do not scale due to the amount of constraints that must be hand-specified. I would like to investigate how to better detect the *end* times of facts using for example, the learned temporal sequence of verbs in the proposed resource (Section 4.1.3) or by learning more precisely the pre-condition and post-condition of verbs' arguments as detailed in Section 4.1.2.

Relation	CTP Context	Unigrams and Bigrams in CTP Change Vectors
US President	was elected, took office, became president	vice president (-), by president (+), administration (+), senator (-), governor (-), candidate (-)
Best Picture	nominated for, to win, won the, was nominated	best picture (+), hour minute (-), academy award (+), oscar (+), nominated (+), won (+), star (-), best actress (+), best actor (+), best supporting (+)

Table 2: Example behavior of various contextual units (unigrams and bigrams) automatically learned in the train CTPs and change vector. The *(+)* and *(-)* signs indicate rise and fall in mention frequency, respectively.

4.2.3 Evaluation for Ontology Extension

In this section, I briefly outline a possibility of using temporal chain of verbs in the constructed resource to automatically extend the ontology of relations in knowledge bases. One approach for automatically discovering new relations in knowledge bases is by co-clustering text contexts that connect known instances of two categories in a knowledge base (Mohamed et al., 2011). This generates a candidate relation for each resulting cluster which can then be analyzed by a trained classifier to determine its semantic validity.

Temporal sequences of verbs in the proposed verbs resource can be used to provide this ontology extension approach with semantically meaningful clusters of verbs as candidate relations. For example, if we always see a group of verbs that (i) are not yet mapped to any relation in the knowledge base, (ii) connect known instances of two categories in a knowledge base, and (iii) always temporally follow/precede another group of verbs that already map to a relation in the knowledge base; we can add the former group of verbs as a candidate relation to complete the temporal chain of relations in the knowledge base. Completing knowledge bases ontology can make the knowledge graph denser and improve inference.

5 Self-Critique of Proposed Approach

Sparsity Some pre- and post-condition may not always be mentioned explicitly in sentences, or they may only be mentioned long before or long after the event that causes the state change happened. To solve this, I propose to utilize signals from several corpora of different nature such that the sparsity of one may be compensated by the redundancy of another. For example, the effect of the verb "marry" (e.g., "married", "spouse") may not be mentioned in news document the day after the marriage event happened. But they may be mentioned for several months after. The sparsity of documents with the day granularity (GigaWord) can be overcome by the redundancy of documents with the year granularity (Google Books N-gram). Another example, typical pre-condition of "marry" such as "engaged" may not be mentioned explicitly in documents but *edited* out of the document when "marry" happens. This pre-condition that is edited out when a particular event happens can be extracted, for example, from Wikipedia edit history of the entity that changes state. I propose to also utilize signals from dictionary definitions of verbs to extract their pre- and post-condition. For example, from the WordNet definition of the verb *alkalify* as "turn basic and less acidic", we can infer that the post-condition of this verb contains adjectives such as "basic" and "less acidic". Lexical resource on verbs such as VerbNet also contains useful diagnostics for detecting pre- and post-condition of verbs from the parse information and the syntactic realizations of verbs and their semantic roles in sentences. For example, the verb *deport* can appear syntactically in sentences as "Agent *deport* Theme *to* Destination" which translates semantically in VerbNet to CAUSE(Agent, Event), LOCATION(START(Event), Theme, ?Source), LOCATION(END(Event), Theme, Destination). We can therefore infer that *deport* causes LOCATION changes in its pre- and post-condition. Other lexical

resource such as FrameNet that contains sequences of causally related frames can be used to infer pre- and post-condition of verbs-expressing-frames. For example, the pre-condition of some verbs that express a frame can be constrained by/inferred from the post-condition of verbs that express the causally-preceding frame. An interesting research direction will be to integrate signals from all these different sources to come up with the overall pre- and post-condition of verbs.

Noise – several events can happen to the seed entity at any particular time, causing several state changes; how to ensure that the state change that we extract is the one corresponding to the event of interest? To solve this, I propose to use only thematically relevant document for learning pre- and post-conditions. For example, to learn the pre- and post-conditions of event such as marriage, only use documents that are relevant to the marriage event. Using a dozen or so of seed entities to select documents relevant for learning pre- and post-condition of the verb of interest is a starting point. Other method for selecting relevant document, for example topic-based clustering can be utilized. A better and cleaner way of learning pre- and post-condition vectors can also be utilized. For example, using the work of persona identification (Bamman et al., 2013; Bamman and Smith) to detect the *persona* of the verb's argument (i.e., what the argument can do/can be done to it, its attributes and its properties), before and after the event expressed by the verb happened.

Empirical Results I stress, however, that the extracted pre- and post-conditions of a change of state verb is going to be supported by statistics from corpora. They are therefore empirical and may in no way be exact – they express only *typically* what pre- and post-conditions of this change of state verb are. The variety of events to which speakers and writers can use a single verb is wide (made especially wide by figurative and metaphorical uses). The pre- and post-conditions that we learn represent perhaps only one particular (the majority) supported use of the verb in the corpora. I also do not make any claim on the necessity or sufficiency of these conditions. For example, "married" maybe a sufficient and necessary post-condition of "marry", but "single" is neither sufficient nor necessary pre-condition of "marry". Similarly, I do not make any claim on the necessity or sufficiency of the learned temporal sequence between verbs. "divorce" is for example, only optionally *after* "marry" i.e., people are not *required* to divorce after they are "married".

Dependency on the Argument The pre- and post-conditions of a change of state verb may depend lexically on both the verb and the argument that changes state. For example, the verb "elect" causes different change of states when applied to different arguments: president vs. vice president, for example. To solve this, we can generalize the different pre- and post-condition *attributes* and *properties* of these arguments by mapping them to their category types and therefore relation values in the knowledge base. For example, both "president" and "vice president" can be mapped to the values of the relation HASJOBPOSITION. We can then generalize that the verb "elect" causes the change in the HASJOBPOSITION relation.

6 Proposed Timeline

Fall 2014 Construct the verb resource containing the mapping of verbs to relations, pre- and post-conditions of change-of-state verbs' arguments, and temporal sequences of verbs

Spring 2015 Build and evaluate the constructed resource for reading natural language text, temporal scoping and ontology extension

Summer 2015 Write and defend thesis

References

Eneko Agirre and Oier Lopez De Lacalle. Clustering wordnet word senses. In *RANLP*, volume 260, pages 121–130, 2003.

David Bamman and Noah A Smith. Unsupervised discovery of biographical structure from text.

David Bamman, Brendan O'Connor, and Noah A. Smith. Learning latent personas of film characters. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*

(Volume 1: Long Papers), pages 352–361, Sofia, Bulgaria, August 2013. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/P13-1035.

J. Callan, M. Hoy, C. Yoo, and L. Zhao. Clueweb09 data set. *boston.lti.cs.cmu.edu*, 2009.

Nathanael Chambers and Daniel Jurafsky. Unsupervised learning of narrative event chains. In *ACL*, pages 789–797. Citeseer, 2008.

Timothy Chklovski and Patrick Pantel. Verbocean: Mining the web for fine-grained semantic verb relations. In *EMNLP*, volume 2004, pages 33–40, 2004.

William Croft. *Syntactic categories and grammatical relations: The cognitive organization of information*. University of Chicago Press, 1991.

David R Dowty. *Word meaning and Montague grammar: The semantics of verbs and times in generative semantics and in Montague's PTQ*, volume 7. Springer, 1979.

Anthony Fader, Stephen Soderland, and Oren Etzioni. Identifying relations for open information extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1535–1545. Association for Computational Linguistics, 2011.

Charles J Fillmore. The case for case. 1967.

Charles John Fillmore. *The grammar of'hitting'and'breaking'*. 1968.

John Rupert Firth. *Papers in Linguistics 1934-1951: Repr*. Oxford University Press, 1961.

David Graff, Junbo Kong, Ke Chen, and Kazuaki Maeda. English gigaword. *Linguistic Data Consortium, Philadelphia*, 2003.

Karin Kipper, Hoa Trang Dang, and Martha Palmer. Class-based construction of a verb lexicon. In *AAAI/IAAI*, pages 691–696, 2000.

Judith A Langer. The process of understanding: Reading for literary and informative purposes. *Research in the Teaching of English*, pages 229–260, 1990.

Ni Lao, Tom Mitchell, and William W Cohen. Random walk inference and learning in a large scale knowledge base. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 529–539. Association for Computational Linguistics, 2011.

Beth Levin. *English verb classes and alternations: A preliminary investigation*. University of Chicago press, 1993.

Dekang Lin and Patrick Pantel. Discovery of inference rules for question-answering. *Natural Language Engineering*, 7(04):343–360, 2001.

Jean-Baptiste Michel, Yuan Kui Shen, Aviva Presser Aiden, Adrian Veres, Matthew K Gray, Joseph P Pickett, Dale Hoiberg, Dan Clancy, Peter Norvig, Jon Orwant, et al. Quantitative analysis of culture using millions of digitized books. *science*, 331(6014):176–182, 2011.

George A Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine J Miller. Introduction to wordnet: An on-line lexical database*. *International journal of lexicography*, 3 (4):235–244, 1990.

Thahir Mohamed, Estevam Hruschka, and Tom Mitchell. Discovering relations between noun categories. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1447–1455, Edinburgh, Scotland, UK., July 2011. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/D11-1134.

Ekaterina Ovchinnikova. *Integration of world knowledge for natural language understanding*, volume 3. Springer, 2012.

Nick Riemer. *Introducing semantics*. Cambridge University Press, 2010.

Herbert Rubenstein and John B Goodenough. Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633, 1965.

Josef Ruppenhofer, Michael Ellsworth, Miriam RL Petruck, Christopher R Johnson, and Jan Scheffczyk. Framenet ii: Extended theory and practice, 2006.

Roger C Schank and Robert P Abelson. *Scripts, plans, goals, and understanding: An inquiry into human knowledge structures*. Psychology Press, 2013.

Stephen Soderland, Brendan Roof, Bo Qin, Shi Xu, Oren Etzioni, et al. Adapting open information extraction to domain-specific relations. *AI Magazine*, 31(3):93–102, 2010.

P. P. Talukdar and K. Crammer. New regularized algorithms for transductive learning. *ECML*, 2009.

Partha Pratim Talukdar, Derry Wijaya, and Tom Mitchell. Coupled temporal scoping of relational facts. In *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining (WSDM)*, Seattle, Washington, USA, February 2012. Association for Computing Machinery.

Xuerui Wang and Andrew McCallum. Topics over time: a non-markov continuous-time model of topical trends. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 424–433. ACM, 2006.

Hila Weisman, Jonathan Berant, Idan Szpektor, and Ido Dagan. Learning verb inference rules from linguistically-motivated evidence. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 194–204. Association for Computational Linguistics, 2012.

Derry Wijaya, Partha Pratim Talukdar, and Tom Mitchell. Acquiring temporal constraints between relations. In *Proceedings of the Conference on Information and Knowledge Management (CIKM 2012)*, Hawaii, USA, October 2012. Association for Computing Machinery.

Derry Wijaya, Partha Pratim Talukdar, and Tom Mitchell. Pidgin: ontology alignment using web text as interlingua. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*, pages 589–598. ACM, 2013.

Derry Wijaya, Ndapa Nakashole, and Tom Mitchell. Ctps: Contextual temporal profiles for time scoping facts via entity state change detection. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2014.

Derry Tanti Wijaya and Philip Gianfortoni. Nut case: What does it mean?: Understanding semantic relationship between nouns in noun compounds through paraphrasing and ranking the paraphrases. In *Proceedings of the 1st international workshop on Search and mining entity-relationship data*, pages 9–14. ACM, 2011.

Derry Tanti Wijaya and Reyyan Yeniterzi. Understanding semantic change of words over centuries. In *Proceedings of the 2011 international workshop on DETecting and Exploiting Cultural diversiTy on the social web*, pages 35–40. ACM, 2011.