**Review of Related Literature**

**Information and Relation Extraction Systems**

Over the years, there has been an increasing amount of interest in the automatic detection of semantic relations, with the goal of making computers understand text. One of the earliest works on this would be that of Hearst (1992) and Berland and Charniak (1999).

Marking the start of the automatic acquisition of relations, Hearst (1992) developed a method that automatically extracts hyponyms (IsA) from a wide variety of texts. One example of this can be seen in the phrase, *Rizzy, a dog*. It shows a hyponymy relation between the words *Rizzy* and *dog*. In extracting hyponymy relations, she used a set of frequently occurring domain-independent lexico-syntactic patterns which undoubtedly define a hyponymy relationship. Though her method has shown encouraging results, it still had some drawbacks such as the ambiguity of some relations extracted. Because her patterns were based on sample sentences in the corpora and aimed to cover as much instances of the hyponymy relation as possible, some of the outputs were indicative of other types of relation. Lastly, she went on to suggest that her method can be used to automatically acquire other types of relation such as meronymy (PartOf).

Later that decade, Berland and Charniak (1999) used a statistical approach to find meronymy (PartOf) relations from a very large corpus. As an example, the phrase *the plot of the story* signifies a meronymy relation between the words *plot* and *story*. In determining such a relation, they used a method similar to Heart’s by also using a pre-defined set of frequently occurring lexico-syntactic patterns. But instead of producing tuples which signify the relation, they focused on producing an ordered list of possible parts given a list of six seed words representing whole objects. The list includes *book, building, car, hospital, plant* and *school*. The *plant* seed word was added to the list to see if the algorithm can identify correct parts despite the ambiguity in the sense of the word. This experiment yielded accuracies lower than the five other seed words. They used statistical metrics to produce the ordered list of possible parts. Though they have stated that their comparable success against Hearst’s method was due to the large corpora that they used, they were still not able to maximize their corpora to their advantage due to the limited number of wholes and patterns used. They produced a list with an accuracy of 55% for the top 50 parts and 70% for the top 20 parts overall.

Despite their efforts, Hearst (1992) and Berland and Charniak (1999) were not able to address the problem of ambiguity in their patterns and outputs. Cases of ambiguity may occur for patterns signifying a number of semantic relations. For example, *the room of the house* shows a meronymy (part of) relation while *the room of the boy* does not. Fortunately, Badulescu, et. al. (2006) also observed this from both works thus using it as his motivation in employing another approach which automatically extracts PartOf relations.

In tackling PartOf relations, Badulescu, et. al. (2006) used a supervised, knowledge-intensive method in contrast to what has been used by Berland and Charniak (1999). They trained the algorithm with manually annotated set of positive (indicative of meronymy) and negative (not indicative of meronymy) training samples to produce a decision tree and a set of rules. Particularly, they used C4.5 decision tree learning to produce the rules. After training, they were able to produce a comprehensive set of classification rules to cover almost all subtypes of PartOf relations. They then tested the said rules using two corpora and had an overall average precision of 80.95% and recall of 75.91%.

In comparison, Berland and Charniak (1999) used a few number of words to represent whole entities which have identifiable parts in their very large corpus. In addition, they limited themselves to single word entities and concepts. Badulescu, et. al. (2006), on the other hand, used an approach which utilizes WordNet and NERD to determine single and multiple word concepts in perspective thus making his approach more general. Lastly, instead of determining the parts of a predefined whole, their work can determine if two noun concepts are indeed part of a PartOf relation through the use of their decision tree and classification rules. Badulescu, et. al. (2006) also tried to replicate the testing done by Berland and Charniak (1999) in their work but because the corpora used were different, the same conditions cannot be applied.

The aforementioned systems aimed to extract specific relations present in an English text. But such relations, IsA and PartOf, though can be easily extracted, are not the only conceptual relations there is. In lieu of this, several systems have already extracted facts and relations openly from plain-texts \cite{Agichtein:2000} \cite{Banko:2008}, web documents \cite{Alani:2003} \cite{Banko:2007}, legal documents \cite{Cheng:2008} and newspapers \cite{Muslea:1999}.

Snowball \cite{Agichtein:2000}, an open relation extraction system, employed a novel strategy in generating patterns and extracting relational tables from plain-text documents, specifically newspaper articles. A training phase is done with minimal training samples from human users. The seed patterns are then used to extract new patterns and relation tuples. As part of its extraction process, the system statistically evaluates the newly generated patterns and tuples and retains only the reliable ones in the new iteration. The large-scale evaluation provides Snowball with a methodology to produce high-quality patterns. However, the system can only produce relational tables involving named-entities accurately labeled by Alembic, a third-party named-entity tagger employed by Snowball. An example of a relational table would be for ORGANIZATION and LOCATION pairs. Such a table can contain the pairs *Microsoft-Edmond* and *Boeing-Seattle* which shows that the organizations *Microsoft* and *Boeing* can be found in *Edmond* and *Seattle*, respectively. Though it is only correct to extract such relations, there are still those which do not only involve a couple of named-entities. Relations involving world states like that between *morning* and *go to school*, clearly shows that a relation can also be between named-entities and phrases. This scenario poses another limitation of Snowball which is similar to \cite{Berland:1999}. Another shortcoming of Snowball would be that it can only extract relations between two named-entities which is not always the case for conceptual relations.

Taking a different path in relation extraction systems, the Artequakt project \cite{Alani:2003} focused on the domain of artists’ biographies and extracted conceptual relations in order to automatically generate biographical accounts of artists. In comparison to previous systems, this one did not use any pre-determined extraction patterns per se and neither did it learn extraction patterns as a pre-process. Instead, the system just had a list of pre-determined ontology relations that it wants to extract along with its pair of concepts. In the whole process, the Artequakt project made use of third-party tools such as the Apple Pie Parser for syntactic analysis or part-of-speech tagging, GATE for entity recognition and WordNet to supplement GATE and to aid in actual relation extraction.

In extracting the relations, the unstructured web documents first goes through an entity recognition tool (GATE). WordNet is also used to supplement in case GATE fails to recognize any named-entity. The document then goes through the actual extraction phase wherein it gets decomposed into paragraphs and sentences. The part-of-speech of each word in a sentence is then labeled. After this, the main components of a sentence such as the subject, verb and object are identified. The system then uses the verb and entity pairs in each sentence and matches them with a corresponding ontology relation and concept pairs. In case of any linguistic variation, WordNet is used to increase the chance of matching with ontology relations and concepts. In its initial experiment, 50 web documents describing 5 artists were used. Promising results were shown as the system was able to extract at most 3 thousand unique conceptual relations with 85% precision and 42% recall on the average. Its low average recall was due to the varying cardinality of some relations. A high recall is preferred for relations with multiple cardinalities like *places\_visited* while high precision is more preferred for relations with a single cardinality like that of *birth\_place*.

Though this work has driven away from the usual use of templates in order to extract their target relations, it still boasts of its portability. The use of ontology relations instead of painstakingly specifying every single template for each target relation takes away the need to force-fit a relation extraction system to a specific domain.

In 2007, Banko et. al. was able to develop an open information extraction system named TextRunner. It processes a corpus of heterogeneous web documents in a single pass without any human intervention. Though this system does not focus heavily on solving the problems faced by previous systems like portability but rather focus on the scalability of RE systems to the web, its novel contributions can still be considered a solution to such problems.

In developing the system, Banko et. al. (2007) used the problems of automation, corpus homogeneity and scalability as motivations. This led to the development of some novel components such as the single pass extractor, self-supervised classifier, synonym resolution and query interface. The single pass extractor tags the sentences with their part-of-speech tags and noun-phrase chunks. Through the self-supervised classifier, it then checks for every pair of noun phrases that are not too far apart and determines whether or not there is a relationship between them. But before this can be done, the classifier has to be trained with positive and negative samples before it can accurately decide which among the noun phrase pairs has a relationship.

Since TextRunner \cite{Banko:2007} does not have a pre-determined set of relations unlike previous works, there is a high chance that the system extracts different tuples representing only one relation. To solve this problem, the system used Resolver to cluster the extracted tuples into sets of synonymous relations and entities.

In evaluating the system, a corpus of 9 million web documents was used. And with that, TextRunner was able to extract approximately 7.8 million well-formed tuples. Human reviewers evaluated some 400 randomly selected extracted tuples and determined that they were 80.4% correct. The system was then further compared to the performance of another traditional IE system, KnowItAll. After using a set of ten high-frequency relations, there were more correct relations extracted by TextRunner than KnowItAll.

In trying to improve TextRunner \cite{Banko:2007}, Banko and Etzioni (2008) developed new systems in order to conduct a survey on the differences of open and traditional relation extraction. In these systems, the Conditional Random Fields model was used to label instances of a relation between all possible entity pairs. This is already an improvement from the Naïve-Bayes classifier used by TextRunner which chooses tokens between entities heuristically and only predicts whether these indicated a relationship or not. Conditional Random Fields, on the other hand, is an undirected graphical model used to model multiple interdependent variables.

O-CRF, the new open relation extraction system, performs a self-supervised training as with TextRunner. It uses independent heuristics and applies them to the PennTreebank in order to obtain labeled relational tuples which are then described with features. Such features include part-of-speech tags, regular expressions, context words and the combination of features six words to the left and six words to the right of the labeled word. The context words used here include only closed classes like prepositions and determiners. Function words like verbs and nouns are not utilized as context words. The labeled relational tuples are then used to train the CRF. In extracting relations, O-CRF first does a single pass over the corpus and uses phrase chunking to identify entities. The CRF is then used to identify and label the relations occurring between entity pairs. As with TextRunner, O-CRF is also beset with duplicate relations. This was solved by applying the Resolver algorithm to predict if two relation strings refers to the same thing.

In order to make comparisons, R1-CRF, a system applying the same CRF model was developed. But this time, the traditional relation extraction paradigm is utilized. Though the same graphical model is used, there were some tweaks in order to comply with the traditional paradigm. A relation is given in advance and instead of training the CRF unsupervised, hand-labeled positive and negative samples are used. And unlike O-CRF, R1-CRF can use context words besides closed classes.

After evaluation, O-CRF showed 88.3% precision and 45.2% recall. These show promising results in using open relation extraction. However, the usage of such a paradigm will only be essential if the number of relations is big or unknown. This is also essential for extraction jobs concerning massive corpora. On the other hand, traditional relation extraction is more suitable for extraction jobs with a small number of target relations.

**Semantic Relations**

The interest in the automatic extraction of semantic relations in text has become one of the growing interests among researchers in the NLP community. And in recent years, a number of them applied different classification techniques on various domains. This, however, led to a variety of disjoint classification schemes which later on became a nuisance to the advancement of the field.

Way back in \citeyearNP{Mann:1987}, \citeA{Mann:1987} presented Rhetoric Structure Theory which describes major features of the organization of natural text. This descriptive theory is used linguistically to characterize the structure of natural text in terms of relations between parts of the text. It is a hierarchical structure which identifies both the transition point of a relation and the items related. Though it can be used for large corpora, its scope is limited to monologues only. Dialogues and spoken text, which are present in stories, are not handled by RST.

The relations in RST are mainly classified into two: nuclear-satellite and multinuclear. The nuclear-satellite relations can still be further classified as presentational or subject matter relations. Presentational relations are those which aim to increase inclination in the reader. An example of this would the Evidence relation which aims to increase the belief of the reader on the nucleus of the relation. Other than that, Motivation, Justify, and Background, among others, are also considered as presentational relations. Subject matter relations, on the other hand, aims to make the reader recognize the relation. Such relations include Condition, Circumstance, Elaboration, Purpose and Volitional cause, among others.

Years after RST, \citeA{Knott:1994} conceptualized a set of coherence relations. But instead of treating relations as constructs used to describe a text, relations were thought of as constructs with psychological reality. Using this as motivation, \citeA{Knott:1994} developed a bottom-up methodology to define a set of relations using cue phrases which is a concrete linguistic indicator of a relation in a text. Unlike most theorists who define relations between entities in a sentence, the relations described in this work are mostly those between the sentences of a text, thus implicit in nature. Such coherence relations are sometimes made explicit through the use of cue phrases like \emph{for example} and \emph{before}. The relations based on the cue phrases are divided into seven classes, namely: sequence, situation, causal/purpose, similarity, contrast/violated expectation/choice, clarifying and interruption.

In the domain of medicine, \citeA{Rosario:2001} defined a classification scheme for two-word noun compounds. Though their data was from MedLine, a collection of biomedical journals, the classes and relations defined in the study was made as general as possible. To be more specific, there was more granularity than those in case frames but the relations were also more general than the ones classified in traditional information extraction systems. In their classification scheme, there were actually 38 relations divided into 12 classes. General relations are also mixed with domain-specific ones. Examples of general relations include time of, frequency, instrument, object, topic and location, among others while those domain-specific ones include defect in location, person/center that treats, defect, research on and bind, among others.

\citeA{Rosario:2002} continued the study on semantic relations for noun compounds. But this time, a different classification scheme was used. Instead of their previous two-level hierarchy, they used the MeSH hierarchy which is a multi-level lexical hierarchy of classifying relations for noun compounds with 15 classes at the topmost level. Each of the 15 topmost classes corresponds mainly to a specific medical terminology or field like Anatomy, Biology, etc. This scheme presents classes which are more granular and more specific to the medicine field.

In \citeyearNP{Nastase:2003}, \citeauthor{Nastase:2003} presented a classification scheme for noun-modifier pairs in base noun phrases. This scheme is a two-level hierarchy classification of semantic relations for noun-modifier pairs. The hierarchy has 5 top-level classes and 30 bottom-level classes. Its 5 superclasses include causality, temporality, spatial, participant, and quality. Causality relations are mainly those which show cause-effect relations. For example, the base noun phrase *cold virus* will have a cause relationship between them since the head word *virus* caused the modifier *cold*. But other than the usual cause and effect relations, there is also the purpose relation which exists whenever the head word is meant for the modifier. Such is the case for the base noun phrase *concert ground* where the head word *ground* has the purpose of having a *concert*. Temporality relations, on the other hand, express time. One example is the frequency relation which holds whenever the head word occurs every time the modifier occurs. This is evident in the base noun phrase *weekly mass*. Spatial relations pertain to having the nature of space. Such is the case for *outgoing call* which shows a direction relation. Participant relations, unlike previous superclasses, include relations similar to semantic roles. One example of this would be the agent role which exists when the modifier performs the head word. The base noun phrase *fan boycott* signifies such a relation since *fan* performs the *boycott.* Lastly, the quality relations are those specifying content, manner and type, among others.

The same year, \citeA{Alani:2003} used a classification scheme very specific to the domain of artists’ biographies. The ontology was derived from the CIDOC Conceptual Reference Model ontology and further modified by adding classes and relations needed to represent pieces of information appropriate for artists. Examples of such relations include date of birth, place of birth and inspired by, among others. These ontology relations are then utilized in generating artist biographies.

Instead of concentrating on classifying semantic relations for noun compounds/base noun phrases, \citeA{Moldovan:2004} specified a scheme in classifying relations for a range of phrases. This includes 35 classes of relations spanning at various syntactic levels. They were mostly derived from the list of relations specified in previous researches. However, it only contains the most frequently used relations in a large corpus. Some of the relations include possession, temporal, part-whole, is-a, cause, purpose, frequency, stimulus, manner and location, among others.

Concentrating more on the field of story generation, \citeA{Nakasone:2006} developed a storytelling ontology model using RST \cite{Mann:1987}. The ontology was made as generic as possible since most storytelling ontology models were defined and constrained by the way the events were linked and the nature of the narratives. Instead of constraining the model with such notions, the solution was more focused on how the narratives were organized and communicated to readers. Since the domain of the model is story generation, the ideas and events are to be focused on the concept of a conflict. Hence, the RST relations utilized were categorized into two: Conflict or Resolution relations. Conflict relations describe how the current state of the story is changed. Such relations include Contrast, Solutionhood, Elaboration, Consequence and Sequence. Resolution relations, on the other hand, describe how to understand the current state of the story. Examples of this type of relation include Background, Cause, Purpose and Result, among others.

And just recently, \citeA{Hendrickx:2009} developed a system which does a multi-way classification of semantic relations between a pair of nominals. But this time, instead of classifying all possible semantic relations, the focus was just on nine mutually exclusive domain-independent semantic relations with enough exhaustive coverage. The list includes Cause-Effect, Instrument-Agency, Product-Producer, Content-Container, Entity-Origin, Entity-Destination, Component-Whole, Member-Collection and Communication-Topic.

**Knowledge Representations**

Common sense knowledge acquisition is not new in the Natural Language Processing field. Over the years, several knowledge repositories or databases have been developed like WordNet, VerbNet, Cyc, FrameNet and ConceptNet. These repositories contain entries ranging from syntactic to semantic in nature. Though most, if not all, contain semantic relations, there are certainly differences on the relations they contain and how they are represented.

Begun in 1984, CYC aims to formalize common sense knowledge into a logical framework. It stores knowledge of every day concepts, objects and events in axioms. The assertions are both manually and automatically done by knowledge engineers at Cycorp assuming that they are already known in the world. In representing the assertions, a first-order predicate calculus, named CycL, with an extension of some second-order features is used. The knowledge base is partitioned into “microtheories” which are a bundle of assertions. Some microtheories are partitioned based on their common assumptions while some are partitioned based on a specific domain and level of detail. This mechanism allows Cyc to infer faster by focusing on a specific microtheory. Each time an inference is made, new assertions may be added into the knowledge base (source: cyc.com).

One of the forerunners and arguably the most popular among knowledge bases is WordNet. It is a general purpose semantic knowledge base started in 1985 at Princeton University. Its database consists of words, mostly nouns, verbs and adjectives. Each entry is structured into senses and associated using a small number of semantic relations such as the synonym, is-a and part-of relations. These relations are represented in WordNet as a semantic network with each word as a node and the relations as edges.

In \citeyearNP{Fillmore:1998}, \shortciteA{Fillmore:1998} developed FrameNet, a lexical resource containing frame-semantic descriptions of each English lexical item (noun, adjective and verb). The semantic domains that FrameNet covers are the following: health care, chance, perception, communication, transaction, time, space, body, motion, life stages, social context, emotion and cognition. The whole lexical database is composed of a lexicon, the frame database and the annotated example sentences. Each lexical entry contains some usual information like part-of-speech as well as formulas which describe how elements of a semantic frame can be recognized. FrameNet, as what was previously stated, also defines the argument structure of each entry in the lexicon through roles but instead of using case-roles or thematic roles, each argument is given a role name relative to a certain concept. The data structures used to represent the lexical entries along with their semantic frames were implemented using SGML.

VerbNet \cite{Kipper:2000} is another repository of semantic information but unlike WordNet, Cyc and ConceptNet, this repository is more focused on verbs and their semantics. It is primarily a verb lexicon using Levin verb classes to represent the lexical entries. As its semantic information, the lexical resource relates each verb’s thematic roles and semantic predicates with syntactic frames and restrictions.

Though VerbNet has semantic information included in its verb lexical entries, it still differs from what WordNet, Cyc and ConceptNet has. The verb lexicon stores semantic roles and not semantic relations. Note that they are two different things though they are both semantic in nature. Semantic roles exist between a verb and its arguments while semantic relations may exist between any parts of speech.

Combining the structure of WordNet and the semantic richness of Cyc, ConceptNet \cite{Liu:2004b} is a large-scale common sense knowledge database aimed to optimize practical inferences over real-world texts. It adopted the semantic network knowledge representation of WordNet and included 17 additional relations such as EffectOf, SubEventOf and CapableOf. This will provide a richer semantic network compared to what WordNet already has. However, there are still differences on the relations they contain. In WordNet, relations are more formal and is assumed to always happen while in the case of ConceptNet, it relations are more informal and defeasible. This means that since ConceptNet is geared towards a more practical inference, its relations may not always happen. One example would be the part-of relation between dog and pet. A dog will always be a canine but not a pet.

Having a set of only 20 relations is not much of an advantage over Cyc since it provides more than 20 and with more detail. However, compared to the use of CycL as a knowledge base representation, ConceptNet’s semantic network representation makes it easier to make practical inferences.