Computing Semantic Similarity between Skill Statements for Approximate Matching

Feng Pan
USC Information Sciences Institute
Marina del Rey, CA 90292
pan@isi.edu

Robert G. Farrell
IBM T. J. Watson Research Center
Hawthorne, NY 10532
robfarr@us.ibm.com

Abstract

This paper explores the problem of computing text similarity between verb phrases describing skilled human behavior for the purpose of finding approximate matches. Four parsers are evaluated on a large corpus of skill statements extracted from an enterprise-wide expertise taxonomy. A similarity measure utilizing common semantic role features extracted from parse trees was found superior to an information-theoretic measure of similarity and comparable to the level of human agreement.

1 Introduction

Knowledge-intensive industries need to become more efficient at deploying the right expertise as quickly and smoothly as possible, thus it is desired to have systems that can quickly match and deploy skilled individuals to meet customer needs. The searches in most of the current matching systems are based on *exact matches* between skill statements. However, exact matching is very likely to miss individuals who are very good matches to the job but didn't select the exact skills that appeared in the open job description.

It is always hard for individuals to find the perfect skills to describe their skill sets. For example, an individual might not know whether to choose a skill stating that refers to "maintaining" a given product or "supporting" it or whether to choose a skill about maintaining a "database" or about maintaining "DB2". Thus, it is desirable for the job search system to be able to find *approximate matches*, instead of only exact matches, between available individuals and open job positions. More specifically, a skill similarity computation is needed to allow searches to be expanded to related skills, and return more potential matches.

In this paper, we present our work on developing a skill similarity computation based upon semantic commonalities between skill statements. Although there has been much work on text similarity metrics (Lin, 1998a; Corley and Mihalcea, 2005), most approaches treat texts as a bag of words and try to find shared words with certain statistical properties based on corpus frequencies. As a result, the *structural information* in the text is ignored in these approaches. We will describe a new semantic approach that takes the structural information of the text into consideration and matches skill statements on corresponding *semantic roles*. We will demonstrate that it can outperform standard statistical text similarity techniques, and reach the level of human agreement.

In Section 2, we first describe the skill statements we extracted from an enterprise-wide expertise taxonomy. In Section 3, we describe the performance of a standard statistical approach on this task. This motivates our semantic approach of matching skill statements on corresponding semantic roles. We also compare and evaluate the performance of four natural language parsers (the Charniak parser, the Stanford parser, the ESG parser, and MINIPAR) for the purpose of our task. An inter-rater agreement study and evaluation of

Proceedings of NAACL HLT 2007, pages 572–579,
Rochester, NY, April 2007.

our approach will be presented in Section 4. We end with a discussion and conclusion.

2 Skill Statements

An expertise taxonomy is a standardized, enterprise-wide language and structure to describe job role requirements and people capabilities (skill sets) across a corporation. In the taxonomy we utilize for this study, skills are associated with job roles. The taxonomy has 10667 skills. Each skill has a title, for example, "Advise BAAN eBusiness ASP." We refer to this title as the *skill statement*.

The official taxonomy update policies require that skill statements be verb phrases using one of 18 valid skill verbs (e.g., *Advise, Architect, Code, Design, Implement, Sell, and Support*).

3 Computing Semantic Similarities between Skill Statements

In this section, we first explain a statistical information-theoretic approach we used as a baseline, and show examples of how it performs for our task. The error analysis of this approach motivates our semantic approach that takes the structural information of the text into consideration. In the remainder of this section, we describe how we extract semantic role information from the syntactic parse trees of the skill statements. Four natural language parsers are compared and evaluated for the purpose of our task.

3.1 Statistical Approach

In order to compute semantic similarities between skill statements, we first adopted one of the standard statistical approaches to the problem of computing text similarities based on Lin's information-theoretic similarity measure (Lin 1998a). Lin defined the commonality between A and B as

$$I(common(A, B))$$

where *common*(*A, B*) is a proportion that states the commonalities between A and B and where the amount of information in proposition *s* is

$$I(s) = -\log P(s)$$

The similarity between A and B is then defined as the ratio between the amount of information needed to state the commonality of A and B and the information needed to fully describe A and B:

$$Sim(A, B) = \frac{\log P(common(A, B))}{\log P(description(A, B))}$$

In order to compute *common*(*A,B*) and *description*(*A,B*), we use standard bag-of-words features, i.e., unigram features -- the frequency of words computed from the entire corpus of the skill statements. Thus *common*(*A,B*) is the unigrams that both skill statements share, and *description*(*A,B*) is the union of the unigrams from both skill statements.

The words are stemmed first so that the words with the same root (e.g., *managing* & *management*) can be found as commonalities between two skill statements. A stop-word list is also used so that the commonly used words in most of the documents (e.g., *the, a*) are not used as features. A formal evaluation of this approach will be presented in Section 4 where the similarity between 75 pairs of skill statements will be evaluated against human judgments, but we discuss some examples here.

In order to see how to improve Lin's statistical similarity measure, we examine sample skill statement pairs which achieve high similarity scores from Lin's measure but were rated consistently as dissimilar by human subjects in our evaluation. Here are two examples:

1. Advise Business Knowledge of *CAD functionality* for FEM
   Advise on Business Knowledge of *Process* for FEM
2. Advise on *Money Market*
   Advise on *Money Center Banking*

In these two examples, although many words are shared between the two pairs of skill statements (*Advise Business Knowledge of ... for FEM* for the first pair; *Advise on Money* for the second pair), they are not similar to human judges. We conjecture that this judgment of dissimilarity is due to the differences between the key components of the skill statements (*CAD functionality* vs. *Process* in the first pair; *Money Market* vs. *Money Center Banking* in the second pair).

This kind of error is common for most statistical approaches to the problem, where common information is computed without considering the structural information in the text. From the above examples, we can see that the similarity computation would be more accurate if the verb phrases *match on corresponding semantic roles*, instead of

matching words from any location in the skill statements. By identifying semantic roles, we can provide more weights to those semantic roles critical for our task, i.e., the key components of the skill statements.

3.2 Identifying and Assigning Semantic Roles

The following example shows the kind of semantic roles we want to be able to identify and assign.

> [**action** Apply] [**theme** Knowledge of [**concept** IBM E-business Middleware]] to [**purpose** PLM Solutions]

In this example, "Apply" is the "action" of the skill; "Knowledge of IBM E-business Middleware" is the "theme" of the skill, where the "concept" semantic role (*IBM E-business Middleware*) specifies the key component of the skill requirement and is the most important role for skill matching; "PLM Solutions" is the "purpose" of the skill.

Our goal was to extract all such semantic role patterns for all the skill statements, and match on corresponding semantic roles. Although there exists some automatic semantic role taggers (Gildea and Jurafsky, 2002; Giuglea and Moschitti, 2006), most of them were trained on PropBank (Palmer et. al., 2005) and/or FrameNet (Johnson et. al., 2003), and perform much worse in other corpora (Pradhan et. al., 2004). Our corpus is from a very different domain (information technology) and there are many domain-specific terms in the skill statements, such as product names, company names, and company-specific nomenclature for product offerings. Given this, we would expect poor performance from these automatic semantic role taggers. Moreover, the semantic role information we need to extract is more detailed and deeper than most of the automatic semantic role taggers can identify and extract (e.g., the "concept" role embedded within the "theme" role).

We developed a specialized parser that extracts semantic role patterns from each of the 18 skill verbs. This semantic role parser can achieve a much higher performance than the general-purpose semantic role taggers. The inputs needed for the semantic role parser are syntactic parse trees generated by a natural language parse of the original skill statements.

3.3 Preprocessing for Parsing

We first used the Charniak parser (2000) to parse the original skill statements. However, among all the 10667 skill statements, 1217 were not parsed as verb phrases, leading to very poor performance. After examining the error cases, we found that abbreviations are used widely in the skill statements. For example,

> Advise Solns Supp Bus Proc Reeng for E&E Eng Procs

These abbreviations made the system unable to determine the part of speech of some words, resulting in incorrect parses. Thus, the first step of the preprocessing was to expand abbreviations.

There were 225 valid abbreviations already identified by the expertise taxonomy team. However, we found many abbreviations that appeared in the skill statements but were not listed there. Since most abbreviations are not words found in a dictionary, in order to find the abbreviations that appear frequently in the skill statements, we first found all the words in the skill statements that were not in WordNet (Miller, 1990). We then ranked them based on their frequencies, and manually identified high frequency abbreviations. Using this approach, we added another 187 abbreviations to the list (a total of 412).

From the error cases, we also found that many words were mistagged as proper nouns, For example, "Technically" in

> Advise Technically for Simulation

was parsed as a proper noun. We realized the reason for this error was that all the words, except for prepositions, are capitalized in the original statements and the parser tends to tag them as proper nouns. To solve this problem, we changed all the capitalized words to lower case, except for the first word and the acronyms (words that have all letters capitalized, e.g., IBM). After applying these two steps of preprocessing, we parsed the skill statements again. This time, more than 200 additional skill statements were parsed as verb phrases after the preprocessing.

When we examined the error cases more closely, we found the errors occur mostly when the skill verbs can be both a noun and a verb (e.g., *design*, *plan*). In those cases, the parser may parse the entire statement as one noun phrase, instead of a verb phrase. In order to disambiguate such cases,

we added a subject ("Employees") to all the skill statements to convert them into full sentences. After applying this additional step of preprocessing, we parsed the skill statements again. This time, only 28 skill statements were not parsed as sentences containing verb phrases, a significant improvement. The remaining errors were due to the use of some words as skill verbs, e.g., "architect"[1], not recognized as verbs by the parser.

3.4 Parser Evaluation and Comparison

While the Charniak parser performed well in our initial verb phrase (VP) test, we decided to compare the Charniak parser's performance with other parsers. For this evaluation, we compared it with the Stanford parser, the ESG parser, and MINIPAR.

The **Stanford parser** (Klein and Manning, 2003) is an unlexicalized statistical syntactic parser that was trained on the same corpus as the Charniak parser (the Penn TreeBank). Its parse tree has the same structure as the Charniak parser.

The **ESG** (English Slot Grammar) **parser** (McCord, 1980) is a rule-based parser based on the slot grammar where each phrase has a head and dependent elements, and is also marked with a syntactic role.

MINIPAR (Lin, 1998b), as a dependency parser, is very similar to the ESG parser in terms of its output. It represents sentence structures as a set of dependency relationships between head words.

Since our purpose is to use the syntactic parses as inputs to extract semantic role patterns, the correctness of the bracketing of the parses and the syntactic labels of the phrases (e.g., NP, VP, and PP) are the most important information for our purposes, whereas the POS (Part-Of-Speech) labels of individual words (e.g., nouns vs. proper nouns) are not that important (also, there are too many domain-specific terms in our data). Thus, our evaluation of the parses is only on the correctness of the bracketing and the syntactic labels of the phrases, not the correctness of the entire parse. For our task, the correctness of the prepositional phrase attachment is especially important for extracting accurate semantic role patterns (Gildea and Jurafsky, 2002). For example, for the sentence

> Apply Knowledge of IBM E-business Middleware to PLM Solutions.

the correct bracketing should be

> Apply [Knowledge [of [IBM E-business Middleware]]] [to [PLM Solutions]].

Thus the parser needs to correctly attach "of IBM E-business Middleware" to "Knowledge" and attach "to PLM Solutions" to "Apply", not "Knowledge".

To evaluate the performance of the parsers, we randomly picked 100 skill statements from our corpus, preprocessed them, and then parsed them using the four different parsers. We then evaluated the parses using the above evaluation measures. The parses were rated as correct or incorrect. No partial score was given. Figure 1 shows the evaluation results. The error analysis reveals four major sources of error for all the parsers, most of which are specific to the domain we are working on:

(1) Many domain specific terms and acronyms. For example, "SAP" in "Employees advise on *SAP* R/3 logistics basic data." was always tagged as a verb by the parsers.
(2) Many long noun phrases. For example, "Employees perform *JD edwards foundation suite address book*."
(3) Some specialized use of punctuation. For example, "Employees perform business transportation consultant-logistics.sys."
(4) Prepositional phrase attachment can be difficult. For example, in "Employees apply IBM infrastructure knowledge *for IDBS*", "for IDBS" should attach to "apply", but many parsers mistakenly attach it to "IBM infrastructure knowledge".

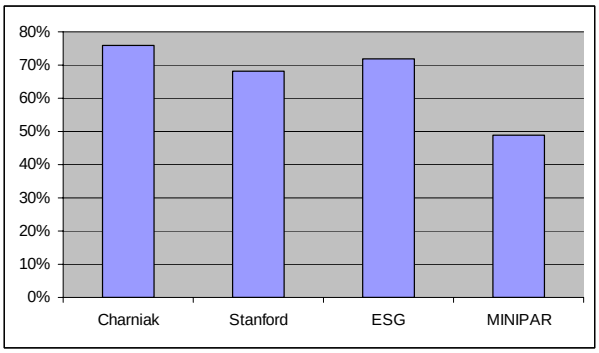

Figure 1. An Evaluation of Four Parsers on the Task of Parsing Human Skill-related Verb Phrases

We noticed that MINIPAR performed much worse compared with the other parsers. The main

[1] "Architect" has no verb sense in WordNet and many other dictionaries, but it does have a verb sense in the Oxford English Dictionary (http://dictionary.oed.com/).

reason is that it always parses the phrase "VERB knowledge of Y" (e.g., "Employees *apply knowledge of web technologies*.") incorrectly -- the parse result always mistakenly attaches "of Y" (e.g., *of web technologies*) to the VERB (e.g., *apply*), not "knowledge". Since there were so many *of* phrases in the test set and in the corpus, this kind of error significantly reduced the performance for our task. These kinds of errors on prepositional phrase attachment in MINIPAR were also mentioned in (Pantel and Lin, 2000).

From the evaluation and comparison results we can see that the Charniak parser performs the best for our task among all the four parsers. This result is consistent with a more thorough evaluation (Swanson and Gordon, 2006) on a different corpus with a set of different target verbs, which showed the Charniak parser performed the best among three parsers (including the Stanford parser and MINPAR) for labeling semantic roles. We note that although the ESG parser performed a little worse than the Charniak parser, its parses contain much richer syntactic (e.g., subject, object) and semantic (e.g., word senses) slot-filling information, which can be very useful to many natural language applications.

3.5 Extracted Semantic Role Patterns

From the parse trees generated by the Charniak parser, we first automatically extracted patterns for each of the 18 skill verbs (e.g., "Advise on NP for NP"), and then we manually identified the semantic roles. For example, the semantic role patterns identified for the skill verb "Advise" are:

- Advise [Theme] (for [Purpose])
- Advise (technically) on/about [Theme] (for [Purpose])
- Advise clients/customers/employees/users on/regarding [Theme]

The corpus also contains embedded sub-semantic-role patterns, for example, for the "Theme" role we extracted the following sub-patterns:

- (application) knowledge of/for [Concept]
- sales of [Concept]
- (technical) implementation of [Concept]

We have extracted and identified a total of 74 such semantic role patterns from the skill statements.

4 Evaluation

In order to evaluate the two approaches (semantic role parsing and statistical) to computing semantic similarity of skill statements in our domain, we first conducted an experiment to evaluate how humans agree on this task, which also provides us with an upper bound accuracy for the task.

4.1 Inter-Rater Agreement and Upper Bound Accuracy

To assess inter-rater agreement, we randomly selected 75 skill pairs from the expertise taxonomy. Since random pairs of verbs would have little or no similarity, we selected skill pairs that share the same job role, or same secondary or primary job category, or from across the entire expertise taxonomy.

These 75 skill pairs are then given to three raters to independently judge their similarities on a 5 point scale from 1 as very similar to 5 as very dissimilar. Since this 5 point scale is very fine-grained, we also converted the judgments to a more coarse-grained measure -- binary judgment: 1 and 2 count as similar; 3-5 as not similar.

The metric we used is the kappa statistic (Carletta, 1996), which factors out the agreement that is expected by chance:

$$\kappa = \frac{P(A) - P(E)}{1 - P(E)}$$

where $P(A)$ is the observed agreement among the raters, and $P(E)$ is the expected agreement, i.e., the probability that the raters agree by chance.

Since the judgment on the 5 point scale is ordinal data, the weighted kappa statistic is used to take the distance of disagreement into consideration (e.g., the disagreement between 1 and 2 is smaller than that between 1 and 5).

The inter-rater agreement results for both the fine-grained and coarse-grained judgments are shown in Table 1. In general, a kappa value above 0.80 represents perfect agreement, 0.60-0.80 represents significant agreement, 0.40-0.60 represents moderate agreement, and 0.20-0.40 is fair agreement (Chklovski and Mihalcea, 2003). We can see that the agreement on the fine-grained judgment is moderate, whereas the agreement on the coarse-grained (binary) judgment is significant.

	Fine-Grained	Coarse-Grained
Kappa	0.412	0.602

Table 1. Inter-Rater Agreement Results.

From the inter-rater agreement evaluation, we can also get an *upper bound accuracy* for our task, i.e., human agreement without factoring out the agreement expected by chance (i.e., $P(A)$ in the kappa statistic). The average $P(A)$ for the coarse-grained (binary) judgment is 0.81, and that constitutes the upper bound accuracy for our task.

4.2 Evaluation of the Statistical Approach

We use the 75 skill pairs as test data to evaluate our semantic similarity approach against human judgments. Considering the reliability of the data, only the coarse-grained (binary) judgments are used. The gold standard is obtained by majority voting from the three raters, i.e., for a given skill pair, if two or more raters judge it as similar, then the gold standard answer is "similar", otherwise it is "not similar".

We first evaluated Lin's statistical approach described in Section 3.1. Among 75 skill pairs, 53 of them were rated correctly according to the human judgments, that is, 70.67% accuracy. The error analysis shows that many of the errors can be corrected if the skills are matched on their corresponding semantic roles. We then evaluated the utility of the extracted semantic role information to see whether it can outperform the statistical approach.

4.3 Evaluation of Semantic Role Matching Approach

For simplicity, we will only report on evaluating semantic role matching on the "concept" role that specifies the key component of the skills, as introduced in Section 3.2.

There are at least two straightforward ways of performing semantic role matching for the skill similarity computation: 1) match on the entire semantic role; 2) match on the head nouns only. But both have their drawbacks: the first approach is too strict and will miss many similar skill statements; the second approach may not only miss the similar skill statements, e.g.,

Perform [Web Services *Planning*][2]
Perform [Web Services *Assessment*]

but also misclassify dissimilar ones as similar, e.g.,

Advise about [Async Transfer Mode (ATM) *Solutions*]
Advise about [CTI *Solutions*]

In order to solve these problems, we used a simple matching criterion from Tversky (1977). The similarity of two texts t_1 and t_2 is determined by:

$$\text{Similarity}(t_1, t_2) = \frac{2 \times (\#\text{ common features between } t_1 \text{ and } t_2)}{\#\text{ total features in } t_1 \text{ and } t_2}$$

This equation states that *two texts are similar if shared features are a large percentage of the total features.* We set a threshold of 0.5, requiring that at least 50% of the features be shared. We apply this criterion to the text contained in the "concept" role.

The words in the calculation are preprocessed first: abbreviations are expanded, stop-words are excluded (e.g., *the* and *of* don't count as shared words), and the remaining words are stemmed (e.g., *manager* and *management* are counted as shared words), as was done in our previous information-theoretic approach. Words connected by punctuation (e.g., *e-business, software/hardware*) are treated as separate words. For example,

Advise on [*Field*/*Force* Management] for Telecom
Apply Knowledge of [Basic *Field Force* Automation]

The shared words between the two "concept" roles (bracketed) are "Field" and "Force", and their shared percentage is (2*2)/7 = 57.14% > 50%, so they are similar.

We have also evaluated this approach on our test set with the 75 skill pairs. Among 75 skill pairs, 60 of them were rated correctly (i.e., 80% accuracy), which significantly outperforms the statistical approach, and is very close to the upper bound accuracy, i.e., human agreement (81%), as shown in Figure 2.

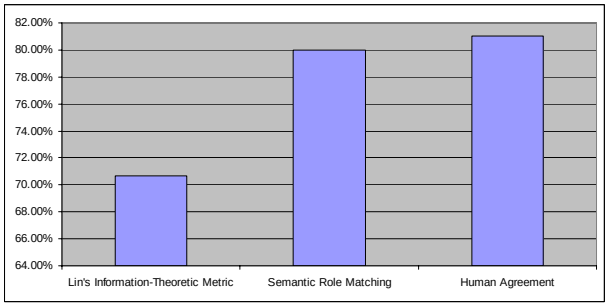

Figure 2. Evaluation on Semantic Similarity between Skill Statements

[2] The "concept" role is identified with brackets, and the head nouns are italic.

The difference between this approach and Lin's information content approach is that this computation is local -- no corpus statistics is used. Also, using this approach, it is easier to set an intuitive threshold (e.g., 50%) for a classification problem (e.g., *similar or not* for our task). With this approach, however, there are also cases that are mistagged as similar, for example,

Apply Knowledge of [Basic Field *Force Automation*]
Advise on [Sales *Force Automation*]

Although "Field Force Automation" and "Sales Force Automation" seem similar on their surface form, they are two quite different concepts. Deeper domain knowledge (such as an ontology) is needed to distinguish such cases.

5 Discussion

We have also investigated several approaches to improving the semantic role text similarity measure we described. One approach is to also consider similarities between skill verbs. In this example:

Implement *Domino Mail Manager*
Develop for *Domino Mail Manager*

although the key components of the skill statements (*Domino Mail Manager*) are the same, their skill verbs are different (*implement* vs. *develop for*). The skills required for "implementing" a system or software product are usually different from those required for "developing for" the same system or software product. This example shows that a semantic similarity computation between skill verbs is required to distinguishing such cases.

Many approaches to the problem of word/concept similarities are based on taxonomies, e.g., WordNet. The simplest approach is to count the number of nodes on the shortest path between two concepts in the taxonomy (Quillian, 1972). The fewer nodes on the path, the more similar the two concepts are. The assumption for this shortest path approach is that the links in the taxonomy represent uniform distances. However, in most taxonomies, sibling concepts deep in the taxonomy are usually more closely related than those higher up. Different approaches have been proposed to discount the depth of the concepts to overcome the problem. Budanitsky and Hirst (2006) thoroughly evaluated six of the approaches (Hirst and St-Onge, Leacock and Chodorow, Jiang and Conrath, Lin, Resnik, Wu and Palmer), and found that Jiang and Conrath (1997) was superior to the other approaches based on their evaluation experiments.

For our task, we compared two approaches to computing skill verb similarities: shortest path vs. Jiang and Conrath. Since the words are compared based on their specific senses, we first manually assigned one most appropriate sense for each of the 18 skill verbs from WordNet. We then used the library developed by Pedersen et al. (2004) to compute their similarity scores.

Table 2 shows the top nine pairs of skill verbs with the highest similarity scores from the two approaches. We can see that the two approaches agree on the top four pairs, but disagree on the rest in the list. One intuitive example is the pair "Lead" and "Manage" which is ranked the 5th by the Jiang and Conrath approach but ranked the 46th by the shortest path approach. It seems that the Jiang and Conrath approach matches better with our human intuition for this example. While we didn't compare these results with human performance, in general most of the similar skill verb pairs listed in the table don't look very similar for our domain. This may be due to the fact that WordNet is a general-purpose taxonomy -- although we have already selected the most appropriate sense for each verb, their relationship represented in the taxonomy may still be quite different from the relationship in our domain. A domain-specific taxonomy for skill verbs may improve the performance. The other reason may be due to the structure of WordNet's verb taxonomy, as mentioned in (Resnik and Diab, 2000), which is considerably wider and shallower than WordNet's noun taxonomy. A different verb lexicon, e.g., VerbNet (Kipper et al., 2000), can be explored.

Shortest Path		**Jiang and Conrath**	
Apply	Use	Apply	Use
Design	Plan	Design	Plan
Apply	Implement	Apply	Implement
Implement	Use	Implement	Use
Analyze	Apply	**Lead**	**Manage**
Analyze	Perform	Apply	Support
Analyze	Support	Support	Use
Analyze	Use	Apply	Sell
Perform	Support	Sell	Use
…	…	…	…

Table 2. Top Similar Skill Verb Pairs

6 Conclusion

In this paper, we have presented our work on a semantic similarity computation for skill statements in natural language. We compared and evaluated four different natural language parsers for our task, and matched skills on their corresponding semantic roles extracted from the parse trees generated by one of these parsers. The evaluation results showed that the skill similarity computation based on semantic role matching can outperform a standard statistical approach and reach the level of human agreement.

The extracted semantic role information can also be incorporated into the standard statistical approaches as additional features. One way is to give higher weights to those semantic role features deemed most important. This approach has achieved a high performance for a text categorization task when combining extracted keywords with the full text (Hulth and Megyesi, 2006).

We have shown that good results can be achieved for a domain-specific text matching task by performing a simple word-based feature comparison on corresponding structural elements of texts. We have shown that the structural elements of importance can be identified by domain-specific pattern analysis of corresponding parse trees. We believe this approach can generalize to other domains where phrases, sentences, or other short texts need to be compared.

Acknowledgements

The majority of this work was performed while the first author was a summer intern at IBM T. J. Watson Research Center in Hawthorne, NY. Thanks to Yael Ravin and Jennifer Lai for supporting this work, Brian White for his help on the software, Michael McCord for assistance with the IBM ESG parser, and the IBM Expertise Taxonomy team for letting us use their data.

References

A. Budanitsky and G. Hirst. 2006. Evaluating WordNet-based Measures of Lexical Semantic Relatedness. *Computational Linguistics*. 32(1):13-47.

J. Carletta. 1996. Assessing agreement on classification tasks: the kappa statistic. *Computational Linguistics*, 22(2):249–254.

E. Charniak. 2000. A maximum-entropy-inspired parser. In *Proceedings of NAACL*.

T. Chklovski and R. Mihalcea. 2003. Exploiting Agreement and Disagreement of Human Annotators for Word Sense Disambiguation. In *Proceedings of RANLP*.

D. Gildea and D. Jurafsky. 2002. Automatic labeling of semantic roles. *Computational Linguistics*, 28(3): 245 – 288.

A. Giuglea and A. Moschitti. 2006. Semantic Role Labeling via FrameNet, VerbNet and PropBank. In *Proceedings of COLING-ACL*.

A. Hulth and B. B. Megyesi. 2006. A Study on Automatically Extracted Keywords in Text Categorization. In *Proceedings of COLING-ACL*.

J. J. Jiang and D. W. Conrath. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings of ROCLING X*.

C. Johnson, M. Petruck, C. Baker, M. Ellsworth, J. Ruppenhofer, and C. Fillmore. 2003. *Framenet: Theory and practice*. Berkeley, California.

K. Kipper, H. T. Dang, M. Palmer. 2000. Class-Based Construction of a Verb Lexicon. In *Proceedings of AAAI*.

D. Klein and C. D. Manning. 2003. Accurate Unlexicalized Parsing. In *Proceedings of ACL*.

D. Lin. 1998a. An information-theoretic definition of similarity. In *Proceedings of ICML*.

D. Lin. 1998b. Dependency-based evaluation of MINIPAR. In *Proceedings of the Workshop at LREC on The Evaluation of Parsing Systems*.

M. C. McCord. 1980. Slot grammars. *Computational Linguistics*, 6: 31-43.

G. A. Miller. 1990. WordNet: an On-line Lexical Database. *International Journal of Lexicography 3(4)*.

M. Palmer, D. Gildea, and P. Kingsbury. 2005. The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1).

P. Pantel and D. Lin. 2000. An unsupervised approach to prepositional phrase attachment using contextually similar words. In *Proceedings of ACL*.

T. Pedersen, S. Patwardhan, and J. Michelizzi. 2004. Wordnet::similarity - measuring the relatedness of concepts. In *Proceedings of AAAI*, Intelligent Systems Demonstration.

S. Pradhan, W. Ward, K. Hacioglu, J. Martin, and D. Jurafsky. 2004. Shallow Semantic Parsing using Support Vector Machines. In *Proceedings of HLT/NAACL*.

M. R. Quillian. 1972. Semantic Memory, Semantic Information Processing. *Semantic information processing*, Cambridge.

P. Resnik and M. Diab. 2000. Measuring verb similarity. In *Proceedings of COGSCI*.

R. Swanson and A. S. Gordon. 2006. A Comparison of Alternative Parse Tree Paths for Labeling Semantic Roles. In *Proceedings of COLING/ACL*.

A. Tversky. 1977. Features of Similarity, *Psychological Review*, vol. 84, no. 4, pages 327-352.