# IIITH: A Corpus-Driven Co-occurrence Based Probabilistic Model for Noun Compound Paraphrasing

**Nitesh Surtani, Arpita Batra, Urmi Ghosh and Soma Paul**
Language Technologies Research Centre
IIIT Hyderabad
Hyderabad, Andhra Pradesh-500032
{nitesh.surtaniug08, arpita.batra, urmi.ghosh}@students.iiit.ac.in, soma@iiit.ac.in

## Abstract

This paper presents a system for automatically generating a set of plausible paraphrases for a given noun compound and rank them in decreasing order of their usage represented by the confidence value provided by the human annotators. Our system implements a corpus-driven probabilistic co-occurrence based model for predicting the paraphrases, that uses a seed list of paraphrases extracted from corpus to predict other paraphrases based on their co-occurrences. The corpus study reveals that the prepositional paraphrases for the noun compounds are quite frequent and well covered but the verb paraphrases, on the other hand, are scarce, revealing the unsuitability of the model for standalone corpus-driven approach. Therefore, to predict other paraphrases, we adopt a two-fold approach: (i) Prediction based on Verb-Verb co-occurrences, in case the seed paraphrases are greater than threshold; and (ii) Prediction based on Semantic Relation of NC, otherwise. The system achieves a comparabale score of 0.23 for the isomorphic system while maintaining a score of 0.26 for the non-isomorphic system.

## 1 Introduction

Semeval 2013 Task 4 (Hendrickx et. al., 2013), "Free Paraphrases of Noun Compounds" is a paraphrase generation task that requires the system to generate multiple paraphrases for a given noun compound and rank them to the best approximation of the human rankings, represented by the corresponding confidence value. The task is an extension of Semeval 2010 Task 9 (Butnariu et al., 2010), where the participants were asked to rank the set of given paraphrases for each noun compound. Although the ranking task is quite distinct from the task of generating paraphrases, however, we have taken many insights from the systems developed for the ranking task, and have reported them appropriately in our system description.

This paper describes a system for generating a ranked set of paraphrases for a given NC. A paraphrase can be Prepositional, Verb or Verb + Prepositional. Since the prepositional paraphrases are easily available in the corpus while the occurrences of verb or verb+prep paraphrases is scarce, the task of paraphrasing becomes significant in finding out a method for predicting reliable paraphrases with verbs for a given NC. Our system implements a model that is based on co-occurrences of the paraphrases and selects those paraphrases that have a higher probability of co-occurring with a set of extracted paraphrases which are referred to as *Seed Paraphrases*. Keeping the verb-paraphrase scarcity issue in mind, we develop a two-way model: (i) Model 1 is used when the seed paraphrases are considerable in number i.e., greater than the threshold value. In this case, other verb paraphrases are predicted based on their co-occurrence with the set of extracted verb paraphrases. (ii) Model 2 is used when the size of the seed list falls below the threshold value, in which case, we make use of the prepositional paraphrases to predict the relation of the noun compound and select verbs that mostly co-occur with that relation. Our system achieves an isomorphic score of 0.23 with a non-isomorphic of 0.26 with the human generated paraphrases. The next section discusses the system.

## 2 System Description

This section of the paper describes each module of the system in detail. The first module of the system



*Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 153–157, Atlanta, Georgia, June 14-15, 2013. 

talks about the Seed data extraction using corpus search. The next module uses the seed data for predicting more verbs that would be used in paraphrasing. The third module uses these predicted verbs in template generation for generating NC Paraphrasing and the generated paraphrases are ranked in the last module.

## 2.1 Seed Data Extraction Module

We have relied mostly on the Google N-gram Corpus for extracting the seed paraphrases. Google has publicly released their web data as n-grams, also known as Web-1T corpus, via the Linguistic Data Consortium (Brants and Franz, 2006). It contains sequences of *n*-terms that occur more than 40 times on the web. Since the corpus consists of raw data from the web, certain pre-processing steps are essential before it can be used. We extract a set of POS templates from the training data, and generalize them enough to accommodate the legitimate paraphrases extracted from the corpus. The following templates are used for extracting n-gram data:

***Head-Mod N-gram:*** This template includes both the head and the modifier in the same regular expression. A corresponding 5-gram template for a NC *Amateur-Championship* is shown in Table 1.

| | |
|---|---|
| Head <*> <*> <*>Mod | *championship* conducted for the *amateurs* |
| Head <*><*> Mod <*> | *championship* for all *amateur* players |
| Head <*>Mod <*><*> | *championship* where *amateur* is competing |

Table 1: Templates for paraphrase extraction

The paraphrases obtained from the above template are quite useful, but scarce. To overcome the issue of coverage of verb paraphrases, a loosely coupled analysis and representation of compounds can be employed, as suggested by (Li et.al, 2010). We retrieve the partial triplets from the n-gram corpus in the form of *"Head Para"* and *"Para Modifier"*.

$$(Head, Para, Mod) \longrightarrow \begin{cases} (Head, Para, ?) \\ (?, Para, Mod) \end{cases}$$

***Head Template:*** Head <*> <*>

***Mod Template:*** <*> <*> Mod; <*> Mod <*>

But the process of generating paraphrases from head and the modifier n-gram incorporates a huge amount of noise and produces a lot of irrelevant paraphrases. Therefore, these partial paraphrases are not directly used for generating the paraphrases but are instead used to diagnose the compatibility of the selected verb with the head and the modifier of the given NC in Section 2.2.2. We also extract paraphrases from ANC and BNC corpus.

## 2.2 Verb Prediction Module

This module is the heart of our system. It implements two models for predicting the verb paraphrases: a Verb Co-occurrence model and a Relation Prediction model. The decision of selection of model for verb prediction is based on the size of the seed list. If the number of seed paraphrases is above the threshold value, the verb co-occurrence model is used whereas the relation prediction model is used if it is below the threshold value.

### 2.2.1 Verb Co-occurrence Model

This model uses the seed paraphrases extracted from the corpus to predict other verb paraphrases by computing their co-occurrences. The model gains insights from the UCD-PN system (Nulty and Costello, 2010) which tries to identify a more general paraphrase by computing the co-occurrence of a paraphrase with other paraphrases. But the task of generating paraphrases has two subtle but significant differences: (i) The list of seed verb paraphrases for a given NC is usually small, with each seed verb having a corresponding probability of occurrence; and (ii) Not all the seed verbs have legitimate representation of the noun compound. Our system incorporates these distinctions in the co-occurrence model discussed below.

Using the training data at hand, we build a Verb-Verb co-occurrence matrix, a 2-D matrix where each cell (i,j) represents the probability of occurrence of $V_j$ when $V_i$ has already occurred.

$$P(V_j|V_i) = \frac{P(V_i, V_j)}{P(V_{i)}} = \frac{Count(V_i, V_j)}{Count(V_i)}$$

The verbs used in co-occurrence matrix are stored in a List A. Now, for a given test NC, the model extracts the seed list of verb paraphrases (referred as List B) from the corpus with their corresponding probabilities. The above model calculates a score for each verb in List A, by computing its co-occurrence with the verbs in List B.

$$score_{a \in A}(V_a) = \sum_{b \in B} P(V_a|V_b) * P(V_b)$$

The term $P(V_b)$ in the above equation represents the relative occurrence of the verb $V_b$ with the given NC. The relevance of this term becomes evident in the next model. The verbs achieving higher score are selected, suggesting a higher probability of co-occurrence with the seed verbs.

### 2.2.2 Semantic Relation Prediction Model

This module describes the second model of the two-way model, and is used by the system when the verbs extracted from the corpus are less than the threshold. In this model, we use prepositional paraphrases, having a pretty good coverage in the corpus, to predict the semantic relation of the compound which helps us in predicting the other paraphrases. The intuition behind using semantic class for predicting paraphrases is that they tend to capture the behavior of the noun compound and can be represented by general paraphrases.

| Noun Compound | Relation | Paraphrase Sel. | |
|---|---|---|---|
| | | Prep | Verb |
| Garden Party | Location | In, At | Held |
| Community Life | Theme | **Of**, In | Made |
| Advertising Agency | Purpose | **For**, Of, In | Doing |

Table 2: Occurrence of Prepositional Paraphrases

**Relation Annotation:** Since a supervised approach is used for identifying the semantic relation of the noun compound, we manually annotate the noun compounds with a semantic relation. We tag each noun compound with one semantic relation from the set used in (Moldovan et. al. 2004).

**Prep-Rel and Verb-Rel Co-occurrence:** A Prep-Rel co-occurrence matrix similar to Verb-Verb co-occurrence matrix discussed in last subsection. This 2-D matrix consists of co-occurrence probabilities between the prepositional paraphrases and the semantic relation of the compound, where each cell (i,j) represents the probability of occurrence of preposition $P_j$ with relation $R_i$. This matrix is used as a model to identify semantic relation using prepositional paraphrases extracted from the corpus.
The Verb-Relation co-occurrence matrix is used to predict the most co-occurring verbs with the identified relation. Each cell (i,j) in the matrix represents the probability of the verb $V_j$ co-occurring with relation $R_i$.

**Relation Extraction:** Research focusing on semantic relation extraction has followed two directions: (i) Statistical approaches to using very large corpus (Berland and Charniak (1999); Hearst (1998)); and (ii) Ontology based approaches using hierarchical structure of wordnet (Moldovan et. al., 2004). We employ a statistical model based on the Preposition-Relation co-occurrence for identifying the relation. The model is quite similar to the one used in Section 2.2, but it is here that the model reveals its actual power. Since two or more relations can be represented by same set of prepositional paraphrases, as *Theme* and *Purpose* in Table 2, it is important to take into account the probabilities with which the extracted prepositions occur in the corpus. In Table 2, the NC *Community Life (Theme)* occurs frequently with preposition '*of*' whereas the NC *Advertising Agency (Purpose)* is mostly represented by preposition '*for*' in the corpus. The term $P(P_p)$ in the equation below captures this phenomenon and classifies these two NCs in their respective classes.

$$score_{r \in R}(r) = \sum_{p \in P} P(r|P_p) * P(P_p)$$

The relation with the highest score is selected as the semantic class of the noun compound. A set of verbs highly co-occurring with that class are selected, and their compatibility with the corresponding noun compound is judged from their occurrences with the partial head and the modifier paraphrases as discussed in Section 2.1. The above classifier performs moderately and classifies a given NC with 42.5% accuracy. We have also tried the Wordnet based Semantic Scattering model (Moldovan et. al., 2004), trained on a set of 400 instances, but achieved an accuracy of 38%, the reason for which can be attributed to the small training set. Since the accuracy of identifying the correct relation is low, we select some paraphrases from the 2nd most probable relation, as assigned by the probabilistic classifier.

## 2.3 Paraphrase Generator Module

After predicting a set of verb for a test noun compound, we use the following templates to generate the paraphrases:

*a) Head VP Mod*
*b) Head VP PP Mod*
*c) Head [that|which] VP PP Mod*

The paraphrases that are extracted from the corpus are also cleaned using the POS templates extracted from the training data.

### 2.4 Paraphrase Ranker Module

Motivated by the observations from Nulty and Costello (2010) that "people tend to use general, semantically light paraphrases more often than detailed, semantically heavy ones", we perform ranking of the paraphrases in two steps: (i) Assigning different weights to different type of paraphrases, i.e. a light weight prepositional paraphrases achieving higher score than the verb paraphrases; and (ii) Ranking a more general paraphrase with the same category higher. A paraphrase *A* is more general that paraphrase *B* (Nulty and Costello, 2010) if

$$P(A|B) > P(B|A)$$

For a list of paraphrases *A* generated for a given compound, each paraphrase *b* in that list is scored using the below eq., where more general paraphrase achieves a high score and is ranked higher.

$$score(b) = \sum_{a \in A} P(b|a)$$

The seed paraphrases extracted from the corpus are ranked higher than the predicted paraphrases.

## 3 Algorithm

This section presents the implementation of the overall system.

```
// Training Phase – Build Co-occurrence Matrices
Verb_Co-occur = 2-D Matrix
Prep-Rel_Co-occur = 2-D Matrix
Verb-Rel_Co-occur = 2-D Matrix
Verb_List = Verb List extracted from training corpus

// Testing – Extract paraphrases with probabilities
Ext_Verb = List of extracted verb paraphrase
VProb = Probability of each Ext_Verb
Ext_Prep = List of extracted prepositional paraphrases
PProb = Probability of each Ext_Prep

Prob_Verb = List // Verbs with their selection score
Prob_Rel = List // Relations with their selection score
Threshold = 3 // Verb threshold for two-way model

if count( Ext_Verb ) > Threshold
   Candidate_Verbs = {Verb_List } - { Ext_Verbs }
   foreach Candidate_Verbs V_i :
      Prob_Verb[V_i] = 0
      foreach Ext_Verb V_j :
         Prob_Verb[V_i] += Verb_Co-occur [V_i][V_j] *
                           VProb[V_j]
else
   foreach Prep-Rel_Co-occur as rel :
      Prob_Rel[rel] = 0
      foreach Ext_Prep as prep :
         Prob_Rel[rel] += Prep-Rel_Co-occur[rel][prep]
                          * PProb[prep]
      Rel=select highestProb(Prob_Rel)
      Prob_Verb = Verb-Rel_Co-occur[Rel]

sort(Prob_Verb)
Verb_Predicted = select top(N)
Paraphrase = generate_paraphrase(verb_predicted)
rank(Paraphrase)
```

## 4 Results

The set of generated paraphrases are evaluated on two metrics: a) Isomorphic; b) Non-isomorphic. In the isomorphic setting, the test paraphrase is matched to the closest reference paraphrases, but the reference paraphrase is removed from the set whereas in non-isomorphic setting, the reference paraphrase which is mapped to a test paraphrase can still be used for matching other test paraphrases. Table 3 presents the scores of the 3 participating teams who have submitted total of 4 systems.

| Systems | Isomorphic | Non-Isomorphic |
|---|---|---|
| SFS | 0.2313 | 0.1794 |
| *IIITH* | *0.2309* | *0.2583* |
| MELODI-Pri | 0.1298 | 0.5484 |
| MELODI-Cont | 0.1357 | 0.536 |

Table 3: Results of the submitted systems

Our system achieves an isomorphic score of 0.23, just below the SFS system maintaining a score of 0.26 for the non-isomorphic system. The two variants of MELODI system get a high score for the non-isomorphic metric but low scores for isomorphic metric as compared to other systems.

## 5 Conclusion

We have described a system for automatically generating a set of paraphrases for a given noun compound, based on the co-occurrences of the paraphrases. The system describes an approach for handling those 38% cases (calculated for optimum threshold value) of NCs where it is not convenient to predict the verbs using their co-occurrences with the seed verbs, because the size of the seed list is below a threshold value. For other cases, the verb co-occurrence model is used to predict the verbs for NC paraphrasing. The optimum value of threshold parameter investigated from experiments is found to be 3, showing that atleast 3 verb paraphrases are necessary to capture the concept of a NC.

## References


Matthew Berland and Eugene Charniak. 1999. *Finding parts in very large*. In Proceeding of ACL 1999

T. Brants and A. Franz. 2006. *Web 1T 5-gram Version1*. Linguistic Data Consortium

Cristina Butnariu, Su Nam Kim, Preslav Nakov, Diarmuid O S´ eaghdha, Stan Szpakowicz, and Tony-Veale. 2010. *Semeval-2 task 9: The interpreta-tion of noun compounds using paraphrasing verbs and prepositions.* In Proceedings of the 5th SIGLEX Workshop on Semantic Evaluation

Cristina Butnariu, Su Nam Kim, Preslav Nakov, Diarmuid O S´eaghdha, Stan Szpakowicz, and Tony-Veale. 2013. *Semeval'13 task 4: Free Paraphrases of Noun Compounds*. In Proceedings of the International Workshop on Semantic Evaluation, Atlanta, Georgia

Marti Hearst. 1998. *Automated Discovery of Word-Net relations.* In An Electronic Lexical Database and-Some of its Applications. MIT Press, Cambridge MA

Mark Lauer. 1995. *Designing Statistical Language-Learners: Experiments on Noun Compounds*. Ph.D. Thesis, Macquarie University

Guofu Li, Alejandra Lopez-Fernandez and Tony Veale. 2010. *UCD-Goggle: A Hybrid System for Noun Compound Paraphrasing*. In Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval-2), Uppsala, Sweden

Dan Moldovan, Adriana Badulescu, Marta Tatu, Daniel Antohe, and Roxana Girju. 2004. *Models for the Semantic Classification of Noun Phrases*. In Proceedings of the HLT-NAACL-04 Workshop on Computational Lexical Semantics, pages 60–67, Boston, MA

Paul Nulty and Fintan Costello. 2010. *UCD-PN: Selecting general paraphrases using conditional probability.* In Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval-2), Uppsala, Sweden