PUNC: A Model of Conceptual Combination

Dermot Lynott and Mark T. Keane

Abstract. We present a model of conceptual combination, PUNC, based on the Constraint Theory of conceptual combination ([1], [2]). This model incorporates the primary constraints of the Constraint Theory; Informativeness, Diagnosticity and Plausibility in an integrated fashion, creating a cognitively plausible mechanism of interpreting novel noun-noun phrases. We detail the model, including knowledge representation, interpretation production mechanisms and the ranking of interpretations in terms of their overall goodness. We also discuss how the PUNC model improves on existing models of conceptual combination.

1 INTRODUCTION

Nominal compounds like holiday drug mule, soccer mom, laptop computer and trash cookies are pervasive and illustrate the creativity of everyday language use. Conceptual combination has long been viewed as a microcosm of the creative and generative nature of language, with new meanings being created constantly by the re-combination of words in syntactically well-formed phrases. It is the need to understand this generativity of language that has motivated several decades of cognitive science research into nominal compounds (e.g., [3], [2], [4], [5], [6], [7], [8]).

This research has thrown up a number of competing models of conceptual combination, each of which tries to capture the main empirical phenomena in the field. We briefly outline these models before concentrating on the Constraint Theory and a new algorithmic instantiation of that theory: PUNC (Producing and Understanding Novel Compounds). PUNC uses the primary constraints of the constraint theory (diagnosticity, informativeness and plausibility) to generate a set of interpretations for any noun-noun compound, ranked by their overall goodness.

2 Other Theories and Models

There are three main theoretical models of conceptual combination in the cognitive Science literature: Dual Process Theory ([8], [9]), CARIN (Competition among Relations in Nominals, [10], [11]) and the Constraint Theory ([1], [2], [12]). Each of these theories proposes different mechanisms and different interactions between the modifier and head concepts of a novel compound (i.e., the first and second words in a compound). We give an overview of these models, before giving more detail on the Constraint Theory in the next section.

The Dual Process Theory [9] proposes two mechanisms for understanding novel, noun-noun compounds. The first, scenario creation, gives rise to interpretations that are linked by a relation (e.g., a robin snake is a snake that eats robins). The second, comparison and alignment, gives rise to interpretations that involve transferring a property from the modifier to the head concept (e.g., a robin snake is a snake with a red breast). Using these two mechanisms, the dual-process view accounts for two broad classes of interpretations; relation-based and property-based. Relational interpretations involve the use of a thematic relation connecting the two words in the compound (e.g., dawn flight is a “flight that takes off at dawn”). Property interpretations assert a property of one concept of the other concept (e.g., bullet train is “a very fast train”, asserting the property of bullet of the train).

The CARIN model posits that property-based interpretations are rare, and that people are far more likely to produce interpretations that use relations. As such, CARIN predicts what interpretations people will produce based on what it terms “Modifier Relation Frequency”; in other words, how a modifier has been combined in other compounds in the past. For example, most compounds of the form a “chocolate X” mean, “an X made from chocolate” (e.g., chocolate egg, chocolate coin, chocolate bar). Therefore, future compounds of this form will be more likely to give rise to this kind of interpretation. In this way CARIN accounts for relational interpretations, while remaining silent on the origins of property interpretations.

However, both Dual-Process and CARIN accounts have drawbacks. Dual-Process requires multiple mechanisms to account for various phenomena, while the CARIN theory largely ignores an entire class of interpretation. The Constraint Theory of conceptual combination overcomes these obstacles by proposing a unitary unification mechanism to account for a wide range of data.

3 The Constraint Theory

The Constraint Theory ([1], [2]) says people understand novel, noun-noun combinations by using three primary constraints – diagnosticity, informativeness and plausibility – whose interactions dictate what

1 Department of Computer Science, University College Dublin, Belfield, Dublin 4, Ireland. Email dermot.lynott@ucd.ie
interpretations are produced. The theory has been realised in a computational model, called C3.

**Diagnosticity** relates to the salience of the features of a concept. A concept’s diagnostic features best distinguish that concept from other concepts. For example, “has wings” is a more diagnostic feature of birds than “has legs”, since many creatures have legs, but far fewer have wings. Diagnostic features facilitate the activation of the concepts involving these features. For example, using the feature “has a trunk” would call to mind “elephants” more easily than using the feature “is grey” ([13]). For this reason diagnostic features are often employed in people’s interpretations for novel compounds. For example, “a beetle that is spiky” is often produced as an interpretation for the compound *cactus beetle* since it uses a highly diagnostic feature of the concept *CACTUS*. Constraint theory proposes that interpretations of novel, noun-noun compounds will utilise diagnostic features of both the head and modifier concepts.

**Plausibility** requires that interpretations utilise features that have already co-occurred in past-experiences. For example, of two interpretations for the compound *angel pig*, “a pig with wings on its torso” is more plausible than “a pig with wings on its tail” for the simple reason that “wings on torsos” has been encountered in the past, whereas “wings on tails” has not. In this way, plausibility overlaps produced interpretations with prior experience (i.e., specific instances stored in memory) to judge its overall goodness.

**Informativeness** specifies that an interpretation must contain a certain amount of new information. Constraint theory predicts that uninformative interpretations will not be produced. For example, a possible interpretation for the compound *pencil bed* might be “a bed that is made of wood”. However, since pencils are traditionally made of wood such an interpretation would be rejected as uninformative.

### 3.1 The C3 Model

The C3 model ([1]) implements the three constraints of diagnosticity, plausibility and informativeness. C3 calculates the diagnosticity of a feature by comparing it to all other concepts in memory (see [1] for details). For instance, if the feature *prickly* occurred only once, in the concept CACTUS, then it would have a high diagnosticity score for that concept. C3 uses the resulting diagnosticity scores to construct partial interpretations for a compound using the diagnostic features of both concepts. To compute plausibility, these partial interpretations are then compared to existing concepts, using their level of overlapping semantic features as a measure of plausibility. An interpretation is scored as being completely plausible if it overlaps entirely with some stored instance in the knowledge base. During this stage these partial interpretations are also elaborated with additional properties from relevant concepts in the knowledge base, but only if these additions increase the plausibility of the interpretation.

Finally, **informativeness** is implemented as a post-hoc filtering process in C3. Once interpretations have been produced they are examined to ascertain whether they are informative or not. Interpretations that do not contain a requisite amount of new information are rejected. Informativeness is therefore a binary affair with new information either being present or absent.

For a given compound C3 generates approximately 4,000 unique interpretations ([1]) with the entire process taking from several hours to days. Obviously it would not be feasible to examine every single interpretation generated, so for simulation purposes a threshold on the overall goodness is usually set so that only the top 10 interpretations are output once processing is complete.

### 4 The PUNC Model

The PUNC model is an implementation of the Constraint Theory that seeks to improve upon C3 by reducing the amount of processing required, while still producing interpretations that parallel those produced by people. PUNC retains the central constraints of diagnosticity, informativeness and plausibility, but implements them differently to the C3 model. We detail the knowledge represented in PUNC and how this knowledge can be meshed to generate interpretations. We also discuss how these changes improve PUNC’s performance compared to existing models.

#### 4.1 Knowledge Representations in PUNC

PUNC uses a simply hierarchy to represent the most diagnostic knowledge associated with each concept. This knowledge encoded includes diverse information such as diagnostic features, functions, roles, and relations to other concepts. For example, the concept CACTUS is represented by diagnostic features such as “has spikes”, “grows in the desert”, “can conserve water”. Since CACTUS also inherits features from CREATURE will have the feature “can eat things”, so specific instances of creatures e.g., BEETLE, DOG, SNAKE, will also be able to eat things. Each feature is weighted by its importance relative to the concept, so “has spikes” is weighted as being the most important feature of CACTUS (e.g., a weighting of 1), while “grows in the desert” is weighted as being of slightly less importance (e.g., a weighting of 3). The PUNC knowledge base was coded blind (i.e., without reference to specific compounds or interpretations) so current weightings currently represent intuitive values for each feature of a concept.

Features that are inherited from a parent concept are not as diagnostic to the child concept (e.g., “can photosynthesise” is less diagnostic of CACTUS than it is of PLANT). Therefore, inherited features are always
4.2 How PUNC Produces Interpretations

The input to PUNC is a noun-noun combination and the output is a list of possible interpretations for that combination, ranked in terms of their overall goodness. The model produces multiple interpretations for each combination; interpretations that parallel those produced by people. PUNC provides a diverse selection of interpretations, using a parsimonious mechanism that reflects both the speed and efficiency with which people understand these compounds, and the diversity of the interpretations they produce.

The two most common types of interpretation produced by people are described as property-based and relation-based. Property-based interpretations occur when a property of the modifier is transferred to the head (e.g., bee beetle as “a yellow and black striped beetle”). Relation-based interpretations occur when some relation links the head and modifier concepts (e.g., cookery magazine as “a magazine that is about cookery”). Property-based interpretations account for 30-50% of interpretations produced, with relation-based interpretations accounting for the same amount again ([1]). Other types of interpretations that occur with less frequency include conjunctions (e.g., pet fish as “something which is both a pet and a fish”) and known-concept interpretations, where the interpretation refers to an object that already exists (e.g., pencil bed as “a pencil case”). Through the mechanism described below, PUNC is able to generate various types of interpretation, in a manner that reflects how people perform the process of conceptual combination.

### 4.2.1 Meshing Knowledge to Produce Interpretations

PUNC generates different types of interpretations through meshing the available knowledge from modifier into the knowledge of the head. When knowledge is meshed it can give rise to possible interactions between the head and modifier concepts, or to the transfer of some aspects of the modifier concept to the head. Each concept consists of a collection of features (e.g., properties, roles, etc.) that describes it and how it relates to other concepts. PUNC compares the features of the modifier concept to those of the head to establish whether elements of both can be meshed.

A simplified description of the knowledge represented for the concepts CACTUS and BEETLE is given in (i) and (ii). We will use these descriptions to illustrate how knowledge is meshed to create interpretations.

- **CACTUS**: is spiky, is found in deserts, can conserve water, is green, can be eaten, can photosynthesise
- **BEETLE**: is black, has 6 legs, has antennae, can eat things, can be eaten.

Since each feature encoded has an associated diagnosticity weighting, PUNC processes features in descending order of diagnosticity. Firstly, PUNC examines each feature in the modifier to see if it can be meshed with the head’s features to create interpretations. Secondly, it examines if the modifier itself can be used to fulfil some role in the knowledge represented in the head concept. As a starting point, PUNC uses the knowledge of the head concept as the core for each interpretation that is produced. This core is then altered or augmented by whatever new features are introduced from the modifier.

For the compound *cactus beetle*, PUNC firstly looks at the modifier’s representation (see i) and attempts to create a unique interpretation using each piece of knowledge. For example, the feature “is spiky” from CACTUS is taken and compared to the features of the head concept. If this feature is considered informative (i.e., if it does not already exist in the head concept), then it can be used to create an interpretation. The “is spiky” knowledge from CACTUS is then meshed with the existing representation of BEETLE forming a representation for the interpretation, which contains all of the diagnostic features of the head concept. The newly incorporated information (“is spiky”) is elevated to being the most diagnostic feature of this new representation, and as such can be used to create a gloss of the new representation of “a beetle that is spiky”. This interpretation distinguishes this beetle from other types of beetle. In this way PUNC outputs both a representation of the interpretation and a text description of the interpretation. In the same way that PUNC produces “a beetle that is spiky”, it will also generate the interpretations “a beetle that is found in deserts” and “a beetle that can conserve water”.

In some cases the knowledge being meshed from the modifier can conflict with knowledge that already exists in the head’s representation. For example, when PUNC compares the knowledge “is green” from CACTUS to the Knowledge of BEETLE, there is a clash, because “is black” is represented as being the colour of beetles. In incorporating this knowledge from the modifier PUNC overrides the information that was in the head to create a representation for “a beetle that is green”.

4.2.1 Meshing Knowledge to Produce Interpretations
When the feature “can be eaten” from the modifier is compared to the head, PUNC meshes this with the knowledge from BEETLE that it “can eat things”, as there is a reciprocal relationship between these two pieces of knowledge. This produces a representation for the interpretation “a beetle that eats cactus”. On the other hand, when PUNC encounters the knowledge “can be eaten” in the both concepts, it does not create the interpretation “a beetle that can be eaten”; as that feature is already present in BEETLE and so the resulting interpretation would be uninformative.

At this point, PUNC examines whether there are features in the head concept that can incorporate the modifier itself to form an interpretation (as opposed to specific features of the modifier concept, as above). For example, there may be actions that the head concept can perform on other concepts (e.g., eats, plays, hunts etc.). For cactus beetle there are no features that allow this (see Table 1 for a list of interpretations produced by PUNC for the compound cactus beetle). However, if the compound were cactus magazine, MAGAZINE contains the feature “can be about something”. PUNC meshes this feature with the modifier itself to produce a representation for “a magazine that is about cacti”. Similarly, cactus magazine could also be “a magazine that is made from cactus” as magazines can be made from different things. Incorporating the modifier itself in this way gives rise to a variety of interpretations depending on the features present in the head.

Each interpretation that is produced by PUNC has an overall goodness score calculated, based on the diagnosticity of the features used to produce the interpretation, and the level of plausibility of the interpretation (as used by the C3 model). This is discussed further in section 4.3.

Using the above mechanism that meshes knowledge of the modifier and head concepts, PUNC can produce a variety of interpretation types that have been described in the literature, such as property-based (e.g., “a hat that has yellow and black stripes”), relation-based (e.g., “a beetle that eats sugar”) and conjunctions (e.g., “a bird that is also a pet”). These interpretations account for the vast majority of interpretations produced by people and PUNC provides an efficient mechanism for generating them.

4.3 Improvements on other models

Overall, PUNC is more efficient than C3 as it reduces the amount of processing necessary to produce a set of psychologically plausible interpretations.

4.3.1 Diagnosticity

[1] specified that employing differential levels of knowledge accessibility might improve model performance by limiting the amount of knowledge that needs to be drawn into the interpretation process. However, the C3 model specifies that all knowledge is equally and directly accessible from memory, whereas PUNC encodes diagnostic information of concepts in descending order of importance for each concept. It has been shown that diagnostic features are more available to people than non-diagnostic features ([16], [17]). By processing the highly diagnostic features of the head and modifier first, PUNC produces “better” interpretations first, with poorer interpretations generally occurring later in the processing stages. This more closely reflects how people produce compounds, since communicative goals generally require people to produce good interpretations first [18].

4.3.2 Informativeness

C3 generates all possible interpretations and then decides whether they are informative or not. PUNC only generates interpretations that are considered informative in the first place. It does this by considering whether information is informative before creating an interpretation, rather than performing a post-hoc test as C3 does. For example, in the compound beetle tar the concepts of TAR and BEETLE both contain the information that they are black. PUNC checks if this information is already present in the head concept and if so, rejects the use of this information in specifying an interpretation. While the end results may be the same for PUNC and C3 in this case, the means of getting there is not. PUNC offers a much more succinct solution to the pragmatic concern of informativeness.

4.3.3 Plausibility

PUNC looks at plausible interactions between the concepts involved, which contributes to the overall goodness of an interpretation. For example, in compound ballet mother, BALLET is represented as a dance, and dances can be performed. The concept MOTHER inherits from both WOMAN and PERSON and so “performs” is one possible feature that a person can have. Because these features of the two concepts mesh perfectly, the interpretation of “a mother who performs ballet” is deemed highly plausible. On the other hand, if the compound being interpreted was ballet dog; “dog that performs ballet” would not receive a high a plausibility scoring since dogs do not typically perform ballet, although the interpretation is still possible. Different interactions between the concepts give rise to different levels of plausibility for an interpretation.

When it comes to plausibility, PUNC address the plausibility level of an interpretation when it is being produced (as explained above). In contrast, C3 takes the less economical approach of producing a partial representation for an interpretation and calculating its plausibility based on the extent of its overlap with all stored instances in memory. It then attempts to enhance the overall plausibility of an interpretation by incorporating extra features from the head and modifier concepts, so as to increase the size of the overlap with stored instances. PUNC avoids retroactively assessing the plausibility in two ways. Firstly, PUNC considers the type of interaction between the two concepts at the
point of producing an interpretation. Secondly, it uses the knowledge of the head concept as the core for each interpretation produced. This means that its overlap with existing concepts is maximised from the outset.

In obviating the need for post-hoc checks, PUNC provides an efficient mechanism for integrating plausibility into the interpretation process.

4.3.4 Order of produced interpretation

PUNC processes the information of the head and modifier concepts in descending order of diagnosticity. This means that PUNC is more likely to produce better interpretations first. By the time the least diagnostic features of a concept are used to form an interpretation it is liable that these interpretations will not be considered as good as previous ones, although a high plausibility score may elevate their overall goodness score. On the other hand, C3 generates all possible interpretations and so does not prioritise the order in which they are produced. PUNC’s approach is therefore a closer reflection of how people produce compounds, who following pragmatic considerations ([18]), tend to produce better interpretations first.

4.3.5 Knowledge Represented

The knowledge represented in PUNC is very different from other models that employ a set of finite relations that can be used to combine a compound’s concepts. For example, the CARIN model ([4], [10]), uses a finite set of relations such as LOCATION, USE, MADE OF. This not only restricts such a model to relation-based interpretations, but meanings that are in fact quite different are formed using the same underlying relation. For example, the “for” relation can be used in a variety of interpretations that have very different meanings (e.g., compare “a magazine for workers” to “a treatment for backs”). PUNC’s use of diverse knowledge allows for more specific interpretations, yet with greater variation in the types of interpretation produced.

4.3.6 Model Performance for Produced Interpretations

By integrating the constraints of diagnosticity, informativeness and plausibility PUNC efficiently generates a list of possible interpretations for compounds. While a model such as C3 produces thousands of interpretations for a single compound, even the interpretations that receive the highest goodness scores may not reflect the interpretations produced by people [1]. This can be partly attributed to limitations of the size of the knowledge base, but this problem applies to any computational model. In PUNC, however, the set of interpretations produced for a given compound generally seems quite sensible and informative. For example, Table 1 shows interpretations produced for the compound cactus beetle. While the most highly ranked interpretations seem to be better candidates (e.g., “a beetle that is spiky”), the lower ranked interpretations, while still possible, appear less good (e.g., “a beetle that can photosynthesis”).

Preliminary testing has been carried out to establish whether the interpretations that PUNC produces do in fact reflect those produced by people, and whether the interpretations most frequently produced by people are considered better interpretations by PUNC’s ranking.

To examine PUNC’s performance we used two sets of interpretations for novel, noun-noun compounds; one from [1] (e.g., whale seal, viper slug) and one set from [19] (e.g., carrot bomb, bee hat). In both of these studies, people were asked to provide what they thought would be plausible interpretations for a list of novel compounds. Participants’ responses were collated and ranked by their frequency of production.

We took the same set of compounds that were presented to those participants (39 in total) and input them into the PUNC model. For each compound PUNC returned an ordered set of interpretations and representations for each interpretation, which we could then compare to participants’ responses.

We found that in 77% of cases the most frequently produced interpretation by people was produced by PUNC as the highest ranking interpretation. For example, for the compound plate paper, the interpretation that was ranked highest by PUNC was “paper that is used to make plates”, which matched the most frequently produced interpretation by the participants. For some compounds, PUNC’s highest ranking interpretation was not the interpretation produced most often by people, but was still among the set of interpretations people produced. For example, bee

---

2 Goodness Score = (Plausibility / 10) * ((Head Diagnosticity * 0.55) + (Modifier Diagnosticity * 0.45))
hat meaning “a hat that is worn by a bee” was considered the best interpretation by PUNC, but it was only the third most frequently produced interpretation by participants. If we compare all of the interpretations produced by PUNC that were also produced by people, there is a strong correlation between PUNC’s goodness score and the frequency of production of participants’ interpretations ($r = -0.74$, $N = 261$).3 These initial findings are promising, but more stringent testing is required to establish whether PUNC can be used as a viable model of conceptual combination.

5. DISCUSSION

Overall, the PUNC model offers an efficient model of conceptual combination, and implementation of the Constraint Theory, while still reflecting a diversity of interpretation types that other models lack (see [4]). By meshing diagnostic features of concepts in a way that obviates the need to consider all possible permutations of information, together with the constraints of informativeness and plausibility, PUNC constructs sets of interpretations that reflect not only those produced by people, but also their relative goodness. This increased efficiency seems to better parallel the speed people manifest in their interpretation of novel word combinations. Additionally, [20] have incorporated the additional constraint of compound familiarity to PUNC’s existing constraints, which has been shown to be an important factor in explaining people’s response times to noun-noun compounds.

As an implementation of the constraint theory of conceptual combination, PUNC makes an important leap from a largely computational level implementation to a more algorithmic-level consideration of conceptual combination. Future work will also consider how the PUNC system can be used to predict the compounds that people produce given particular object descriptions.

ACKNOWLEDGMENTS

The work presented here has been part-funded by a grant from the Irish Research Council for Science, Engineering and Technology.

REFERENCES


3 The negative correlation is due to lower goodness scores in PUNC referring to better interpretations, while lower scores in participant’s responses refer to poorer interpretations.