An exploration of semantic features in an unsupervised thematic fit evaluation framework

Asad Sayeed, Vera Demberg, and
Pavel Shkadzko
Computational Linguistics and
Phonetics / MMCI Cluster of Excellence
Saarland University
{asayeed,vera,pavels}@coli.uni-saarland.de

English. Thematic fit *is the extent to which an entity fits a thematic role in the semantic frame of an event, e.g., how well humans would rate "knife" as an instrument of an event of cutting. We explore the use of the SENNA semantic role-labeller in defining a distributional space in order to build an unsupervised model of event-entity thematic fit judgements. We test a number of ways of extracting features from SENNA-labelled versions of the ukWaC and BNC corpora and identify tradeoffs. Some of our Distributional Memory models outperform an existing syntax-based model (TypeDM) that uses hand-crafted rules for role inference on a previously tested data set. We combine the results of a selected SENNA-based model with TypeDM's results and find that there is some amount of complementarity in what a syntactic and a semantic model will cover. In the process, we create a broad-coverage semantically-labelled corpus.*
Italiano. *Con* **plausibilità tematica** *si intende la misura in cui una data entità può svolgere in modo adeguato un ruolo tematico all'interno del "frame" semantico associato a un evento; ad esempio come un essere umano giudica un coltello nel ruolo di strumento all'interno dell'evento tagliare. In questo contributo, abbiamo indagato l'uso di SENNA, uno strumento per l'etichettatura automatica di ruoli semantici, nella definizione di uno spazio distribuzionale finalizzato alla definizione di un modello non supervisionato di giudizi di appropriatezza semantica evento-entità. Sono stati testati diversi metodi per l'estrazione di tratti da versioni dei corpora ukWaC e BNC annotate con SENNA, e sono state identificate soluzioni che combinano diversi approcci. Alcuni dei modelli di Distributional Memory definiti hanno superato un modello esistente basato su informazione sintattica (TypeDM) codificata all'interno di regole definite manualmente per l'assegnazione di ruoli ad un insieme di dati precedentemente testati. Attraverso la combinazione dei risultati di un modello selezionato basato su SENNA con i risultati ottenuti con TypeDM, abbiamo rilevato una certa complementarità nella copertura di modelli sintattici e semantici. All'interno di questo processo, abbiamo costruito un corpus ad ampia copertura semanticamente annotato.*

1. Introduction

Can automated tasks in natural language semantics be accomplished entirely through models that do not require the contribution of semantic features to work at high accuracy? Unsupervised semantic role labellers such as that of Titov and Klementiev (2011) and Lang and Lapata (2011) do exactly this: predict semantic roles strictly from syntactic realizations. In other words, for practical purposes, the relevant and frequent semantic cases might be completely covered by learned syntactic information. For example, given

a sentence *The newspaper was put on the table*, such SRL systems would identify that *the table* should receive a "location" role purely from the syntactic dependencies centered around the preposition *on*.

We could extend this thinking to a slightly different task: thematic fit modelling. It could well be the case that the *the table* could be judged a more appropriate filler of a location role for *put* than, e.g., *the perceptiveness*, entirely due to information about the frequency of word collocations and syntactic dependencies collected through corpus data, handmade grammars, and so on. In fact, today's distributional models used for modelling of selectional preference or thematic fit generally base their estimates on syntactic or string co-occurrence models (Baroni and Lenci 2010; Ritter, Mausam, and Etzioni 2010; Ó Séaghdha 2010). The Distributional Memory (DM) model by Baroni and Lenci (2010) is one example of an unsupervised model based on syntactic dependencies, which has been successfully applied to many different distributional similarity tasks, and also has been used in compositional models (Lenci 2011).

While earlier work has shown that syntactic relations and thematic roles are related concepts (Levin 1993), there are also a large number of cases where thematic roles assigned by a role labeller and their best-matching syntactic relations do not correspond (Palmer, Gildea, and Kingsbury 2005). However, it is possible that this non-correspondence is not a problem for estimating typical agents and patients from large amounts of data: agents will most of the time coincide with subjects, and patients will most of the time coincide with syntactic objects. On the other hand, the best resource for estimating thematic fit should be based on labels that most closely correspond to the target task, i.e. semantic role labelling, instead of syntactic parsing.

Being able to automatically assess the semantic similarity between concepts as well as the thematic fit of words in particular relationships to one another has numerous applications for problems related to natural language processing, including syntactic (attachment ambiguities) and semantic parsing, question answering, and in the generation of lexical predictions for upcoming content in highly incremental language processing, which is relevant for tasks such as simultaneous translation as well as psycholinguistic modelling of human language comprehension.

Semantics can be modelled at two levels. One level is compositional semantics, which is concerned with how the meanings of words are combined. Another level is lexical semantics, which include distributional models; these latter represent a word's meaning as a vector of weights derived from counts of words with which the word occurs (see for an overview (Erk 2012; Turney, Pantel, and others 2010)). A current challenge is to bring these approaches together. In recent work, distributional models with structured vector spaces have been proposed. In these models, linguistic properties are taken into account by encoding the grammatical or semantic relation between a word and the words in its context.

DM is a particularly suitable approach for our requirements, as it satisfies the requirements specific to our above-mentioned goals including assessing the semantic fit of words in different grammatical functions and generating semantic predictions, as it is broad-coverage and multi-directional (different semantic spaces can be generated on demand from the DM by projecting the tensor onto 2-way matrices by fixing the third dimension to, e.g., "object").

The usability and quality of the semantic similarity estimates produced by DM models depend not only on how the word pairs and their relations are represented, but also on the training data and the types of relations between words that are used to define the links between words in the model. Baroni and Lenci have chosen the very fast MaltParser (Nivre et al. 2007) to generate the semantic space. The MaltParser

version used by Baroni and Lenci distinguishes a relatively small number of syntactic roles, and in particular does not mark the subject of passives differently from subjects of active sentences. For our target applications in incremental semantic parsing (Sayeed and Demberg 2013), we are however more strongly interested in thematic roles (agent, patient) between words than in their syntactic configurations (subject, object).

In this paper, we produce DM models based directly on features generated from a semantic role labeller that does not directly use an underlying syntactic parse. The labelling tool we use, SENNA (Collobert et al. 2011), labels spans of text with PropBank-style semantic roles, but the spans often include complex modifiers that contain nouns that are not the direct recipients of the roles assigned by the labeler[1]. Consequently, we test out different mechanisms of finding the heads of the roles, including exploiting the syntactic parse provided to us by the Baroni and Lenci work *post hoc*. We find that a precise head-finding has a positive effect on performance on our thematic fit modeling task. In the process, we also produce a semantically labeled corpus that includes ukWaC and BNC[2].

In addition, we want to test the extent to which a DM trained directly on a role labeller which produces PropBank style semantic annotations can complement the syntax-based DM model on thematic fit tasks, given a similar corpus of training data. We maintain the unsupervised aspects of both models by combining their ratings by averaging without any weight estimation (we "guess" 50%) and show that we get an improvement in matching human judgements collected from previous experiments. We demonstrate that a fully unsupervised model based on the SENNA role-labeller outperforms a corresponding model based on MaltParser dependencies (DepDM) by a wide margin. Furthermore, we show that the SENNA-based model can compete with Baroni and Lenci's better performing TypeDM model on some thematic fit tasks; TypeDM involves hand-crafted rules over and above the finding of syntactic heads, unlike our DMs. We then investigate the differences between the characteristics of the models by mixing TypeDM and a high-performing SENNA-based model at different stages of the thematic fit evaluation process. We thus demonstrate that the SENNA-based model makes a separate contribution to thematic fit evaluation.

1.1 Thematic role typicality

Thematic roles describe the relations that entities take in an event or relation. Thematic role fit correlates with human plausibility judgments (Padó, Crocker, and Keller 2009; Vandekerckhove, Sandra, and Daelemans 2009), which can be used to evaluate whether a distributional semantic model can be effectively encoded in the distributional space.

A suitable dataset is the plausibility judgment data set by Padó (2007), which includes 18 verbs with up to twelve nominal arguments, totalling 414 verb-noun-role triples. The words were chosen based on their frequency in the Penn Treebank and FrameNet; we call this simply the "Padó" dataset from now on (see table 1). Human subjects were asked how common the nominal arguments were as agents or as patients for the verbs. We also evaluate the DM models on a data set by McRae et al. (1998), which contains thematic role plausibility judgments for 1444 verb-role-noun triples calculated over the course of several experiments. We call these "McRae agent/patient".

1 E.g., "Bob ate the donut that poisoned Mary"; "Mary" is not a recipient of the patient role of "eat", but SENNA labels it as such, as it is part of the noun phrase including "donut".

2 We provide the entire labelled corpus at `http://rollen.mmci.uni-saarland.de`. Users of the corpus should cite this paper.

Verb	Noun	Semantic role	Score
advise	doctor	agent	6.8
advise	doctor	patient	4.0
confuse	baby	agent	3.7
confuse	baby	patient	6.0
eat	lunch	agent	1.1
eat	lunch	patient	6.9

Table 1
Sample of judgements from Padó dataset.

However, these triples do contain a significant proportion of words which only very rarely occur in our training data, and will therefore be represented more sparsely. The McRae dataset is thus a more difficult data set to model than the Padó dataset.

While the first two data sets only contain plausibility judgments for verbs and their agents and patients, we additionally use two data sets containing judgments for locations (274 verb-location pairs) and instruments (248 verb-instrument pairs) (Ferretti, McRae, and Hatherell 2001) that we call "Ferretti locations" and "Ferretti instruments" respectively. We use them to see how well these models apply to roles other than agent and patient. All ratings were on a scale of 1 to 7.

Finally, we include two other data sets that come from an exercise in determining the effect of verb polysemy on thematic fit modelling (Greenberg, Demberg, and Sayeed 2015). The first, which we call "Greenberg objects", are verbs and objects with ratings (from 1 to 7) obtained from Mechanical Turk; there are a total of 480 items in this dataset. The second are 240 filler items—"Greenberg fillers"—used in the Mechanical Turk annotation that have been taken from the McRae agent/patient data and re-rated. While the Padó and McRae items used a formulation "How common is it for a *noun* to be *verb*ed?", the Greenberg data was evaluated with a statement that workers were supposed to rate: "A *noun* is something that is *verb*ed." This is intended to reduce the effect that real-world frequency has on the answers given by workers: that caviar may not be a part of most people's meals should have a minimal effect on its thematic fit as something that is eaten. In this feature exploration, we include the Greenberg ratings as another set of data points.

1.2 Semantic role labelling

Semantic role labelling (SRL) is the task of assigning semantic roles such as agent, patient, location, etc. to entities related to a verb or predicate. Structured lexica such as FrameNet, VerbNet and PropBank have been developed as resources which describe the roles a word can have and annotate them in text corpora such as the PTB. Both supervised and unsupervised techniques for SRL have been developed. Some build on top of a syntactic parser, while others work directly on word sequences. In this paper, we use SENNA. SENNA has the advantage of being very fast and robust (not needing parsed text); it is able to label large, noisy corpora such as UKWAC. Without making inferences over parse trees, SENNA is able to distinguish thematic roles and identify them directly (figure 1).

SENNA uses PropBank roles which include agent (ARG0) and patient (ARG1) roles (up to ARG4 based on a classification of roles for which verbs directly subcategorize,

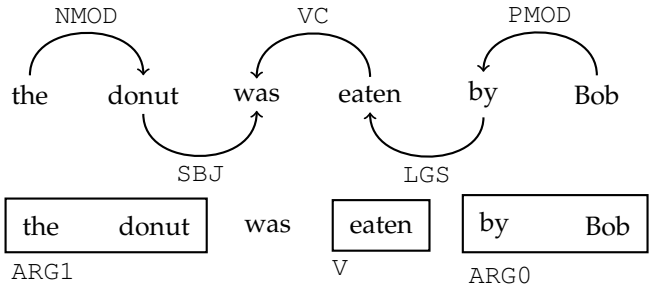

Figure 1
MaltParser dependency parse vs. SENNA semantic role labelling. SENNA directly identifies the patient role that is the syntactic subject of the passive sentence.

such as instruments and benefactives). It also includes a large number of modifier roles, such as for locations (ARGM-LOC) and temporal expressions (ARGM-TMP).

We also make use of MaltParser output in order to refine the output of SENNA—we do not exploit, as Baroni and Lenci do, the actual content of the syntactic dependencies produced by MaltParser. We explore *inter alia* the extent to which the increased precision in finding role-assignees from dependency connection information assists in producing a better match to human judgements.

2. Distributional Memory

Baroni and Lenci (2010) present a framework for recording distributional information about linguistic co-occurrences in a manner explicitly designed to be multifunctional rather than being tightly designed to reflect a particular task. Distributional Memory (DM) takes the form of an order-3 tensor, where two of the tensor axes represent words or lemmas and the third axis represents the syntactic link between them.

Baroni and Lenci construct their tensor from a combination of corpora: the UKWAC corpus, consisting of crawled UK-based web pages, the British National Corpus (BNC), and a large amount of English Wikipedia. Their linking relation is based on the dependency-parser output of MaltParser (Nivre et al. 2007), where the links consist of lexicalized dependency paths and lexico-syntactic shallow patterns, selected by hand-crafted rules.

The tensor is represented as a sparse array of triples of the form *(word, link, word)* with values as local mutual information (LMI), calculated as $O \log \frac{O}{E}$ where O is the observed occurrence count of the triple and E the count expected if we assume each element of the triple has a probability of appearing that is independent of one another. Baroni and Lenci propose different versions of representing the link between the words (encoding the link between the words in different degrees of detail) and ways of counting frequencies. Their DepDM model encodes the link as the dependency path between words, and each *(word,link,word)* triple is counted. These occurrence frequencies of triples is used to calculate LMI[3]. The more successful TypeDM model uses the same dependency path encoding as a link but bases the LMI estimates on type frequencies (counted over grammatical structures that link the words) rather than token frequencies.

3 E.g., in "Bob ate the donut", they would count *(Bob,subj,eat)*, *(donut,obj,eat)*, and *(Bob,verb,donut)* as triples.

Model	*Coverage* (%)	ρ
BagPack	100	60
TypeDM+SDDM (Malt-only)	99	59
SDDM (Malt-only)	99	56
TypeDM	100	51
Padó	97	51
ParCos	98	48
DepDM	100	35

Table 2
Comparison on Padó data, results of other models from Baroni and Lenci (2010).

Both DepDM and TypeDM also contain inverse links: if *(monster, sbj_tr eat)* appears in the tensor with a given LMI, another entry with the same LMI will appear as *(eat, sbj_tr*$^{-1}$*, monster)*.

Baroni and Lenci provide algorithms to perform computations relevant to various tasks in NLP and computational psycholinguistics. These operations are implemented by querying slices of the tensor. To assess the fit of a noun w_1 in a role r for a verb w_2, they construct a centroid from the 20 top fillers for r with w_2 selected by LMI, using subject and object link dependencies instead of thematic roles. To illustrate, in order to determine how well *table* fits as a location for *put*, they would construct a centroid of other locations for *put* that appear in the DM, e.g. *desk, shelf, account* ...

The cosine similarity between w_1's vector and the centroid represents the preference for the noun in that role for that verb. The centroid used to calculate the similarity represents the characteristics of the verb's typical role-fillers in all the other contexts in which they appear.

Baroni and Lenci test their procedure against the Padó et al. similarity judgements by using Spearman's ρ. They compare their model against the results of a series of other models, and find that they achieve full coverage of the data with a ρ of 0.51, higher than most of the other models except for the BagPack algorithm (Herdağdelen and Baroni 2009), the only supervised system in the comparison, which achieved 0.60. Using the TypeDM tensor they freely provide, we replicated their result using our own tensor-processing implementation.

3. SENNA

SENNA (Collobert and Weston 2007; Collobert et al. 2011) is a high performance role labeller well-suited for labelling a corpus the size of UKWAC and BNC due to its speed. It uses a multi-layer neural network architecture that learns in a sliding window over token sequences in a process similar to a conditional random field, working on raw text instead of syntactic parses. SENNA extracts features related to word identity, capitalization, and the last two characters of each word. From these features, the network derives features related to verb position, POS tags and chunking. It uses hidden layers to learn latent features from the texts which are relevant for the labelling task.

SENNA was trained on PropBank and large amounts of unlabelled data. It achieves a role labelling F score of 75.49%, which is still comparable to state-of-the-art SRL systems which use parse trees as input[4].

4. Implementation

4.1 Feature selection

We constructed our DMs from a combination of ukWaC and BNC[5] by running the sentences individually through SENNA and counting the *(assignee, role, assigner)* triples that emerged from the SENNA labelling. However, SENNA assigns roles to entire phrases, some of which include complex modifiers such as relative clauses. We needed to find a more specific focus on the assigners (always verbs, given the training data used for SENNA) and assignees; however, there are number of ways to do this, and we experimented with different types of head-finding, which is a form of feature selection for a SENNA-based DM.

4.1.1 Head-finding

Head-finding takes place over spans found by SENNA. There are two basic ways in which we search for heads, one partly dependent on a syntactic parse ("Malt-based"), one not ("linear").

Linear. The "linear" algorithm is not based on a syntactic parse, but instead on the part-of-speech tags processed in sequence. It is similar to the Magerman (Magerman 1994) head percolation heuristic. This head-finding algorithm uses a heuristic to detect the head of a noun phrase. This heuristic operates as follows: iterating over each word w, if the POS tag is nominal store it and forget any previous nominal words. At the end of the string, return the stored word. Discard the word if a possessive or other such "interrupting" item is passed. For example, in the phrase "The Iron Dragon's Daughter", the system would first store "Iron", forget "Iron" when it found the possessive "Dragon's", and return "Daughter". It is possible for it to return nothing, if the span given to it has no suitable candidate. The linear process can only identify nominal constituents; we found that adding heuristics to detect other possible role-assignees (e.g. adverbs in instrumental roles) reduced the quality of the output due to unavoidable overlaps between the criteria used in the heuristics.

Malt-based. This head-finding procedure makes use of a small amount of syntactic dependency information. The "Malt-based" head-finding heuristic is based on the MaltParser output for ukWaC and BNC that was provided by Baroni and Lenci and used in the construction of DepDM and TypeDM. In essence, it involves using the dependencies reaching the role-assigning verb. Each word directly connected to the role-assigning verb inside the SENNA span is identified as a separate role-filler for the DM. We transitively explore connections via function words such as prepositions and modals. See figure 2 for an example.

4 For example, one very recent system reaches 81.53% F-score on role-labelling (Foland Jr and Martin 2015) on in-domain data.

5 This is the same as Baroni and Lenci, except that they included Wikipedia text—we found no improvement from this and omitted it to reduce processing time.

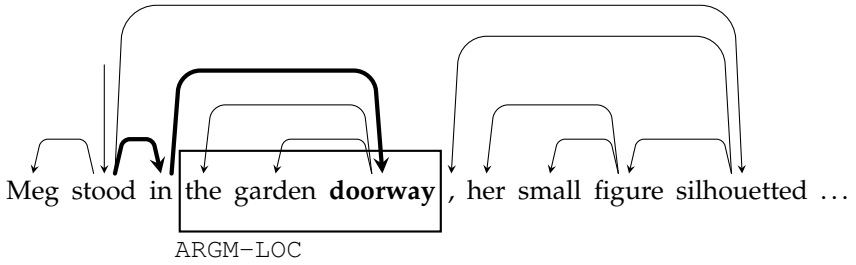

Figure 2
The Malt-based head-finding algorithm illustrated. SENNA has found "the garden doorway" and assigned it ARGM-LOC. We use the MaltParser dependency structure to find that "doorway" is the head. We skip "in" by POS tag and transitively pass over it. The first item we encounter is the head.

This heuristic is somewhat conservative. It is sometimes the case that SENNA identifies a role-filler that does not have a Malt-based dependency path. Therefore, in addition to the "Malt-only" strategy, we include two fallback strategies when a MaltParser dependency does not resolve to any item. This strategy allows us to include role-assignees that are not necessarily nominal, such as verbs in subordinate clauses receiving roles from other verbs or adverbs taking on instrumental roles.

The first fallback is based on the linear head-finding strategy. We make use of the linear strategy whenever there is no valid MaltParser dependency.

The second fallback we call "span", and it is based on the idea that even if SENNA has identified a role-bearing span of text to which MaltParser does not connect the verb direction, we can find an indirect link via another content word closer to the verb. The span technique searches for the word within the span with a direct dependency link closest to the beginning of the sentence, under the assumption that verbs tend to appear early in English sentences. If the span-exterior word is a closed-class item such as a preposition, it finds the word with the dependency link that is next closest to the beginning of the sentence. Our qualitative comparison of the linear and span fallbacks suggests that the span fallback may be slightly better, and we test this in our experiments.

4.1.2 Vocabulary selection

Using the entire vocabularies of ukWaC and BNC would be prohibitively costly in terms of resources, as there are numerous items that are *hapax legomena* or otherwise occur very rarely. Therefore, we do some initial vocabulary selection, in two ways.

The first vocabulary selection method we call "balanced" and proceeds in a manner similar to Baroni and Lenci. We choose the 30,000 most frequent nominal words (including pronouns) in COCA whose lemmas are present as lemmas in WordNet; we do the same for 6,000 verbs. The balanced vocabulary produces DMs that only contain nominal and verbal role-assignees.

The second vocabulary selection method we call "prolific", and it involves using the top 50,000 most frequent words (by type) in the corpus itself, regardless of part of speech. However, as our DMs are evaluated with POS-labelled lexical items (the POS tags we use are coarse: simply nouns, verbs, adverbs, and so on), this can evolve into a "real" vocabulary that is somewhat larger that 50,000, as many word types represent multiple parts of speech (e.g., "fish" is both a verb and a noun).

Some of our features involve a parameter such as vocabulary size. We choose reasonable values for these and avoid parameter searching in order for the tensors to remain as unsupervised as possible.

4.2 From corpus to DMs

The process of constructing DMs from the above proceeds as follows:

1. The corpus is first tokenized and some character normalization is performed, as the ukWaC data is collected from the Web and contains characters that are not accepted by SENNA. We use the lemmatization performed via MaltParser and provided by Baroni and Lenci.
2. Each sentence is run through SENNA and the role-assigning verbs with their role-assigned spans are collected. There is a very small amount of data loss due to parser errors and software crashes.
3. One of the head-finding algorithms is run over the spans: either linear-only, Malt-only, Malt-based with linear fallback, and Malt-based with span fallback. These effectively constitute separate processed corpora.
4. A table of counts is constructed from each of the head-finding output corpora, the counts being occurrences of *(assigner, role, assignee)* triples. The assigners and assignees are filtered by either balanced or prolific vocabularies.
5. This table of counts is processed into LMI values and the inverse links are also created. Triples with zero or negative LMI values are removed. This produces the final set of DM tensors.

In terms of choosing links, our implementation most closely corresponds to Baroni and Lenci's DepDM model over MaltParser dependencies. The SENNA-based tensors are used to evaluate thematic fit data as in the method of Baroni and Lenci described above.

5. Experiments

We ran experiments with our tensor (henceforth SDDM) on the following sources of thematic fit data: the Padó dataset, agents/patients from McRae, instrumental roles from Ferretti et al. (2001), location roles from Ferretti et al., and objects from Greenberg et al. (2015), both experimental items and fillers. We also concatenated all the datasets together and evaluated them as a whole. For each dataset, we calculated Spearman's ρ with respect to human plausibility judgments. We compared this against the performance of TypeDM given our implementation of Baroni and Lenci's thematic fit query system. We then took the average of the scores of SDDM and TypeDM for each of these human judgement sources and likewise report ρ.

During centroid construction, we used the ARG0 and ARG1 roles to find typical nouns for subject and object respectively. The Padó data set contains a number of items that have ARG2 roles; Baroni and Lenci map these to object roles or subject roles depending on the verb[6]; our SENNA-based DM can use ARG2 directly. For the

6 They mapped ARG2 for verbs like "ask" and "tell" to subject roles for "hit" to object roles.

Head-finding	Padó	McRae agent/patient	Ferretti loc.	Ferretti inst.
Linear	51	27	12	19
Malt	56	27	13	27
Malt+linear	52	28	13	23
Malt+span	54	27	16	23

Head-finding	Greenberg objects	Greenberg fillers	All items
Linear	42	19	29
Malt	40	16	31
Malt+linear	44	20	31
Malt+span	40	17	30

Table 3
Spearman's ρ values (x100) with SDDM variants by head-finding algorithm with the balanced vocabulary.

instrument role data, we mapped the verb-noun pairs to PropBank roles ARG2, ARG3 for verbs that have an INSTRUMENT in their frame, otherwise ARGM-MNR. We used "with" as the link for TypeDM-centroids; the same PropBank roles work with SENNA. For location roles, we used ARGM-LOC; TypeDM centroids are built with "in", "at", and "on" as locative prepositions.

Using the different DM construction techniques from section 4, we arrive at the following exploration of the feature space:

1. We use the balanced vocabulary and vary the technique. We test the linear and Malt-only head-finding algorithms, and we test the Malt-based head-finding with the linear and span fallbacks.
2. We use the balanced vocabulary with the linear head-finding algorithm.
3. We then use the prolific vocabulary and test the linear and Malt-only techniques and the Malt-based technique with the span fallback.
4. Finally, we average the cosines from Baroni and Lenci's TypeDM with the Malt-only technique to explore the differences in what is encoded by a SENNA-based tensor from a fully MaltParser-based one.

6. Results and discussion

For all our results, we report coverage and Spearman's ρ. Spearman's ρ is calculated with missing items (due to absence in the tensor on which the result was based) removed from the calculation.

Our SENNA-based tensors are taken directly from SENNA output in a manner analogous to Baroni and Lenci's construction of DepDM from MaltParser dependency output. Both of them do much better than the reported results for DepDM (see Table 2) and two of the Malt-based SDDM variants (Malt-only and Malt+Span) do better than TypeDM on the Padó data set.

Head-finding	Padó	McRae agent/patient	Ferretti loc.	Ferretti inst.
Linear	51	26	12	13
Malt	52	24	15	14
Malt+span	50	25	19	12
Head-finding	**Greenberg objects**	**Greenberg fillers**	**All items**	
Linear	43	18	27	
Malt	38	14	26	
Malt+span	40	16	27	

Table 4
Spearman's ρ values (x100) for SDDM with the prolific vocabulary.

6.1 Varying the head-finding algorithm

The results of these experiments are summarized in Table 3. We find that particularly for the Padó dataset and the instrument dataset, the Malt-only DM tensor is best-performing and exceeds the linear head-finding by a large margin. Some of this improvement is possibly due to the fact that our tensors can handle ARG2 directly; however, the biggest gain is realized for the Malt-only process. On the other hand, the Malt-only tensor does relatively poorly on the Greenberg dataset, both the experimental objects and the fillers.

As for the fallback variants of the Malt-based tensor, the span fallback reflects some of the behaviour of the Malt-only tensor, although it does particularly well at the location dataset. In contrast, the linear fallback does well on the Greenberg data. It also appears that all the tensors have roughly the same effectiveness when run on all the datasets together. These observations suggest that there are tradeoffs relative to the "application" of the tensor. The Greenberg data pulls down the performance of the Malt-based and Malt+span tensors most acutely; it should be noted that the main difference with the Padó data is the question that was asked as well as its presentation via Mechanical Turk[7]. On the whole, the fallbacks appear to have a moderating effect on the Malt-based tensor, reducing ρ on Padò and Ferretti instruments but increasing it on some of the other data sets.

6.2 Prolific vocabulary

In table 4, we see that by comparison to table 3, the larger prolific vocabularies do not assist much, and in fact hurt overall. The only improvement we see is in the Malt+span version, which does better than the balanced-vocabulary tensors on locations.

The balanced vocabulary produces tensors with a vocabulary size of 36,000, but the prolific vocabulary allows for considerable variation depending on how many forms have multiple realizations as open-class parts-of-speech, which is very common in English. The Malt-only prolific DM has 68,178 vocabulary items, 84,903 with the span fallback, and the linear-only has 89,979. As simply adding vocabulary and thus expanding the scope of feature selection does not appear to differentiate these tensors,

7 That the Greenberg data is only objects doesn't seem to make much difference here. The Malt-only tensor on Padó objects alone yields a ρ of 48 while the linear-only tensor yields 42—the linear-only tensor is considerably worse on objects for the Padó dataset.

System	Padó	McRae agent/patient	Ferretti loc.	Ferretti inst.
TypeDM	53	33	23	36
SDDM (Malt-only)	56	27	13	28
TypeDM+SDDM	59	34	21	39
TypeDM/SDDM correlation	65	54	26	30

System	Greenberg objects	Greenberg fillers	All items
TypeDM	53	31	38
SDDM (Malt-only)	41	16	31
TypeDM+SDDM	51	26	38
TypeDM/SDDM correlation	66	68	54

Table 5
Spearman's ρ values (x100) for TypeDM and averaging of TypeDM with the Malt-only SDDM variant.

the influence of less frequent items becomes more apparent—and their influence is not necessarily positive.

6.3 Combining with TypeDM

6.3.1 Cosine averaging

Table 5 contains the result of averaging the cosine scores produced by TypeDM[8] with those of two SDDM variants. The variant we try is the Malt-only tensor, because it exceeds TypeDM's score on Padó on its own. Averaging its cosine scores with TypeDM over the Padó data set provides a further boost. A small improvement occurs with the McRae dataset, but the instruments also show a further increase. However, the Malt-only tensor reduces performance on locations and the Greenberg datasets, and it makes no difference on the all-items dataset.

So why does the Malt-only tensor reduce ρ on locations and the Greenberg data? To analyse this, we calculated Spearman's ρ values on a per-verb basis in the locations data set for TypeDM and for Malt-only SDDM. Since each verb in this dataset has 5-10 nouns, the ρ values will not by themselves be highly reliable, but they can provide some hints for error analysis. Taken individually, the majority of verbs appear to improve with the Malt-based tensor. These seem to include verbs such as "act", "confess", "worship" and "study".

The Malt-only SDDM tensor has a relatively high but not total correlations with TypeDM in terms of cosine, especially apparent in the all-items dataset. These values suggest that even when their correlations with human judgements are similar, they only partly model the same aspects of thematic fit. The correlations for the Greenberg data set are the highest, while the correlations for the locations data set are the lowest, and these are the worst-performing when the cosines are averaged. This suggests that the cosine-averaging process is most beneficial when the correlation between the models is

8 Baroni and Lenci used a version of the Pado data that erroneously swapped the judgments for some ARG0 vs. ARG1. Our repair of this error caused a small upward shift in the TypeDM results (from ρ=51 to 53), but should not cause DepDM (not made publicly available) to catch up.

DM variant	Vocabulary	Above-zero LMI values
Linear	balanced	36,071,848
Malt	balanced	22,284,150
Malt+linear	balanced	36,046,090
Malt+span	balanced	26,139,198
Linear	prolific	62,970,314
Malt	prolific	35,575,476
Malt+span	prolific	42,581,704
TypeDM	N/A	131,369,458

Table 6
The number of above-zero LMI values in each SDDM variant, giving an idea of the relative dimensionality of vectors in each DM.

within an "intermediate" range—too much or too little inter-model correlation means that the differences between the two are adding noise, not signal.

These distinctions are usually more apparent in the less-frequent dimensions. The Baroni and Lenci's thematic fit evaluation process uses the top 20 highest-LMI role-fillers for a given verb/role combination. We compared the dimensions of the centroids constructed from these top 20 between TypeDM the SDDM and found little to distinguish them qualitatively; the most "frequent" dimensions remain most frequent regardless of technique. Once again, we find that the "long tail" of smaller dimensions is what distinguishes these techniques from one other, but not necessarily the size of that long tail, as we can see from table 6. Aside from TypeDM, which is much larger, most of the variation in DM size has little overall relation to the performance of the DM; the best competitor to TypeDM (or contributor, when the results are combined) is the Malt-only tensor, and it is the smallest.

6.3.2 Centroid candidate selection

There are at least two means by which one form of DM tensor could outperform another on a thematic fit task. One of them is via the respective "semantic spaces" their vectors inhabit—the individual magnitudes of the dimensions of the vectors used to construct role-prototypical centroids and test them against individual noun vectors. The other means is by the candidate nouns that are used to select the vectors from which the centroids are constructed. In this section, we investigate how these factors interact. Since the same LMI calculation is used for both the construction of vector dimensions as well as being the ranking criterion for candidate nouns within a single DM, are these factors actually dependent on one another?

In order to answer this question, we tested the result of using the top 20 candidates of one tensor for the construction of centroids using the vectors of another. Specifically, we took the TypeDM candidates and used them to construct Malt-only SDDM centroids. We then took cosines of those centroids with the Malt-only SDDM noun vectors for each dataset. We call this result $\text{SDDM}_{\text{TypeDM}}$. We also ran this process *vice versa*, and we call that result $\text{TypeDM}_{\text{SDDM}}$.

In table 7, we observe that using TypeDM vectors with SDDM candidates had a small overall deleterious effect on the TypeDM results except on the one dataset for which Malt-only SDDM outperformed TypeDM—the Padó dataset. It had a large negative effect on Ferretti instruments. On the other hand, using SDDM vectors with

System	Padó	McRae agent/patient	Ferretti loc.	Ferretti inst.
TypeDM	53	33	23	36
SDDM (Malt-only)	56	27	13	28
$\text{TypeDM}_{\text{SDDM}}$	56	32	19	21
$\text{SDDM}_{\text{TypeDM}}$	48	25	19	45
Avg. Jaccard index	38	38	29	14

System	Greenberg objects	Greenberg fillers	All items
TypeDM	53	31	38
SDDM (Malt-only)	41	16	31
$\text{TypeDM}_{\text{SDDM}}$	49	28	36
$\text{SDDM}_{\text{TypeDM}}$	50	29	33
Avg. Jaccard index	48	48	42

Table 7
Spearman's ρ values (x100) for TypeDM, SDDM (malt-only), and the candidate-swapped results. We also include the average Jaccard index (x100) of overlap between the candidate nouns for each dataset.

TypeDM candidates hurt SDDM's performance on Padó, but improved its performance considerably on both Greenberg datasets and enormously on instruments—the best instruments results so far.

What could account for these differences? One thing to note is that the SDDM balanced vocabulary is still considerably larger than that of TypeDM, so some SDDM candidates for centroid construction would not have corresponding vectors in TypeDM. This would mean that the $\text{TypeDM}_{\text{SDDM}}$ centroids thus constructed would be the sum of less than 20 vectors. Greenberg et al. (2015) show that the number of vectors chosen for the centroid does not have a drastic influence on performance of the centroid beyond 10. For the cosines calculated over the Padó dataset, only an average of 7.6% of the candidate nouns obtained from Malt-only SDDM were not found in TypeDM. However, it does appear to reduce ρ in several of the datasets, but only the Ferretti instruments score falls drastically.

We tested the overlap of candidate nouns between TypeDM and the Malt-only SDDM. That is, for every verb-role pair, we found the top 20 candidate nouns for each tensor and used the Jaccard index (size of intersection divided by size of union) between them as a measure of overlap. For each dataset, we report the average Jaccard index. What we find is that the average Jaccard indices are never more than 50%—the intersections are always much smaller than the unions. What stands out is that Ferretti instruments, which experiences the largest changes due to swapping noun candidates, also has by far the lowest Jaccard index.

To illustrate this, we took at look at the verb "call". In the instruments dataset, to call with paper or to call with a radio is rated poorly by humans (2.5/7 each), whereas to call with a telephone or a voice is given very high ratings (6.9 and 6.9 respectively). $\text{TypeDM}_{\text{SDDM}}$ does poorly on this: calling with paper is rated much higher (39%) than calling with a voice or a telephone (24% and 31%). $\text{SDDM}_{\text{TypeDM}}$ does well, giving 4% ratings to calling with paper and radio and 16% and 24% ratings to telephone and voice (the relative ranking is what matters to ρ, not the absolute cosines). The overlap between the top 20 noun candidates of TypeDM and SDDM is very poor, with a Jaccard index of only 8%.

Qualitatively, TypeDM chooses much better typical instruments of "call", such as "message" and "enquiry". However, $\text{SDDM}_{\text{TypeDM}}$ still outperforms TypeDM alone on instruments. The centroid from $\text{SDDM}_{\text{TypeDM}}$ still consists of statistics collected for the Malt-only SDDM. In other words, the vectors of SDDM produce better results than TypeDM's vectors for instruments after we apply TypeDM's typical noun candidates.

It thus appears that candidate selection and centroid construction are separable from one another, and that while TypeDM seems to produce better noun candidates for some of the datasets, Malt-only SDDM's semantic space can sometimes be superior for the thematic fit task.

6.4 Coverage

All the datasets presented here have a coverage in the above 95% range over all items.

7. Conclusions

In this work, we constructed a number of DM tensors based on SENNA-annotated thematic roles in the process of probing the feature space for their use in thematic fit evaluation. We find that combining the output of SENNA with MaltParser dependency link information provides a boost in thematic fit performance in some well-studied datasets such as the Padó data (over and above TypeDM) and the Ferretti instrument data, but other feature selections provide improvements in the Ferretti location data.

The linking thematic roles used to construct these tensors are not further augmented by hand-crafted inference rules making them similar to Baroni and Lenci's DepDM. All of them easily exceed DepDM on the Padó data set. When used in combination with TypeDM in an unsupervised score averaging process, we find that the fit to human judgements improves for some datasets and declines for other data sets, particularly the Greenberg data. On the whole, we find that the SDDM tensors encode a different part of linguistic experience from the explcitly syntax-based TypeDM in the fine structure of dimensions they contain. Using the semantic space of SDDM with the prototypical role-filler candidate noun selection of TypeDM improves the performance of SDDM on some data sets, particularly instruments, showing that candidate selection and vector component calculation can be strategically separated.

This work made use of Baroni and Lenci's thematic fit evaluation process just as they describe it. However, future work could include testing out the augmented versions of this algorithm that involve clustering the vectors that go into centroid formation to produce multiple centroids reflecting verb senses (Greenberg, Sayeed, and Demberg 2015). A further item of future work would be to understand why the Greenberg data works better with the linear head-finding (as opposed ot the Malt-based head-finding), despite its overall similarity to the Padó data.

References

Baroni, Marco and Alessandro Lenci. 2010. Distributional memory: A general framework for corpus-based semantics. *Computational Linguistics*, 36(4):673–721.

Collobert, Ronan and Jason Weston. 2007. Fast semantic extraction using a novel neural network architecture. In *Annual meeting-association for computational linguistics*, volume 45, page 560.

Collobert, Ronan, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 12:2493–2537.

Erk, Katrin. 2012. Vector space models of word meaning and phrase meaning: A survey. *Language and Linguistics Compass*, 6(10):635–653.

Ferretti, Todd R, Ken McRae, and Andrea Hatherell. 2001. Integrating verbs, situation schemas, and thematic role concepts. *Journal of Memory and Language*, 44(4):516–547.

Foland Jr, William R and James H Martin. 2015. Dependency-based semantic role labeling using convolutional neural networks. In *Lexical and Computational Semantics (* SEM 2015)*.

Greenberg, Clayton, Vera Demberg, and Asad Sayeed. 2015. Verb polysemy and frequency effects in thematic fit modeling. In *Proceedings of the 6th Workshop on Cognitive Modeling and Computational Linguistics*, pages 48–57, Denver, Colorado, June. Association for Computational Linguistics.

Greenberg, Clayton, Asad Sayeed, and Vera Demberg. 2015. Improving unsupervised vector-space thematic fit evaluation via role-filler prototype clustering. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics Human Language Technologies (NAACL HLT)*.

Herdağdelen, Amaç and Marco Baroni. 2009. BagPack: A general framework to represent semantic relations. In *Proceedings of the Workshop on Geometrical Models of Natural Language Semantics*, pages 33–40, Athens, Greece, March. Association for Computational Linguistics.

Lang, Joel and Mirella Lapata. 2011. Unsupervised semantic role induction via split-merge clustering. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1117–1126, Portland, Oregon, USA, June. Association for Computational Linguistics.

Lenci, Alessandro. 2011. Composing and updating verb argument expectations: A distributional semantic model. In *Proceedings of the 2Nd Workshop on Cognitive Modeling and Computational Linguistics*, CMCL '11, pages 58–66, Stroudsburg, PA, USA. Association for Computational Linguistics.

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

Magerman, David M. 1994. *Natural Lagnuage Parsing as Statistical Pattern Recognition*. Ph.D. thesis, Stanford University.

McRae, Ken, Michael J Spivey-Knowlton, and Michael K Tanenhaus. 1998. Modeling the influence of thematic fit (and other constraints) in on-line sentence comprehension. *Journal of Memory and Language*, 38(3):283–312.

Nivre, Joakim, Johan Hall, Jens Nilsson, Atanas Chanev, Gülsen Eryigit, Sandra Kübler, Svetoslav Marinov, and Erwin Marsi. 2007. Maltparser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(2):95–135.

Ó Séaghdha, Diarmuid. 2010. Latent variable models of selectional preference. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 435–444, Stroudsburg, PA, USA. Association for Computational Linguistics.

Padó, Ulrike. 2007. *The integration of syntax and semantic plausibility in a wide-coverage model of human sentence processing*. Ph.D. thesis, Universitätsbibliothek.

Padó, Ulrike, Matthew W. Crocker, and Frank Keller. 2009. A probabilistic model of semantic plausibility in sentence processing. *Cognitive Science*, 33(5):794–838.

Palmer, Martha, Daniel Gildea, and Paul Kingsbury. 2005. The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106.

Ritter, Alan, Mausam, and Oren Etzioni. 2010. A latent dirichlet allocation method for selectional preferences. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 424–434, Stroudsburg, PA, USA. Association for Computational Linguistics.

Sayeed, Asad and Vera Demberg. 2013. The semantic augmentation of a psycholinguistically-motivated syntactic formalism. In *Cognitive modeling and computational linguistics (CMCL 2013)*, pages 57–65, Sofia, Bulgaria, 8 August.

Titov, Ivan and Alexandre Klementiev. 2011. A bayesian model for unsupervised semantic parsing. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 1445–1455, Portland, Oregon, USA, June. Association for Computational Linguistics.

Turney, Peter D, Patrick Pantel, et al. 2010. From frequency to meaning: Vector space models of semantics. *Journal of artificial intelligence research*, 37(1):141–188.

Vandekerckhove, Bram, Dominiek Sandra, and Walter Daelemans. 2009. A robust and extensible exemplar-based model of thematic fit. In *EACL 2009, 12th Conference of the European Chapter of the Association for Computational Linguistics, Proceedings of the Conference, Athens, Greece, March 30 - April 3, 2009*, pages 826–834.