A Method for the Evaluation of the Semantic Localization of Sentence Embeddings

Lyndon White

Tuesday 28th July, 2015

1 Introduction

Various sentence embeddings are often called Semantic Vector Space Representations, such as in [1]. This paper aims to provide a method to assess the accuracy of that statement. Such a tool will allow us to assert whether the vector representation of a given method is more influenced by the structure (syntax) of the sentence embedded, or of its meaning (semantics).

Clearly, the vector sentence models are very useful and functional. As developed [illegible] have been producing highly competitive benchmarks in a number of areas.. These benchmarks, while showing the great practical use of techniques, do not look closely at the theoretical foundation.

Here a new method is presented to assess directly how accurate the methods localize sentences in semantic space. Checking if sentence which mean the same thing are located closely, or not; and whether sentences of different meanings are located distantly, or not. This will allow a better understanding of how these models work, and suggest new directions for the development in this area.

1.0.1 Sentence Embedding

Here we will define a sentence embedding as the embedding of a sentence. This is often called a phrase in the work of Sorcher et Al[2, 3, 4, 5, 6], and is one form of what is called a paragraph or document in [7]. A semantically, a sentence is the expression of a single idea. Thus by embedding sentences into vector spaces is embedding ideas into vector spaces.

1.0.2 Semantic Equivalence

Sentences are said to be semantically equivalent if they each imply the other – the relationship is that of bidirectional entailment. If we consider two sentences A and B. A is equivalent to B if A being true implies that B also must be true, and if B is true then A also must be true. This definition is closely related to that for logical equivalence. Different equivalent sentences are said to be paraphrases of each other.

The paraphrases from the Microsoft Research Paraphrase Corpus (MSRPC) were judged by the human raters to have the same high-level meaning, and to show "mostly bidirectional entailment" [8]. That is to say, that while each sentence may contain information which is not implied by the other, the core meaning of the sentences is entailed by both. [8] provides the examples of:

> Charles O. Prince, 53, was named as Mr. Weill's successor.
>
> Mr. Weill's longtime confidant, Charles O. Prince, 53, was named as his successor.

While additional information is present, each sentence implies most of the meaning of the other. Thus while not semantically equivalent, they are semantically close.

It should be noted that that semantic similarity is often defined differently for words. While semantic similarity for sentences is defined in terms of shared meaning and mutual entailment, semantic similarity for words can be defined in terms of shared properties[9]. For example: "rise" and "fall" are antonyms, but are under the aforementioned definition for word semantic similarity are very similar: they both describe an vertical change in (potentially metaphorical) position. However, the sentences "The share price is predicted to rise." and "The share price is predicted to fall." are not semantically similar sentences, as they do not imply each other - in-fact each implies that the other is false. This sentence definition of semantic equivalence can be seen to be essential in applications such as machine translation.

1.0.3 A Substitution Based Evaluation Corpus

The method proposed here to evaluate the semantic localization requires the construction of a corpus which contains semantically close and semantically distant sentences. To allow discrimination between embeddings which capture structure (syntax) or word content, and those which capture meaning, the semantically distinct sentences should have the same structure and word content except for the variations evaluated. Thus a method is required to change the meaning of sentences, without changing anything else. To change meaning at least one word must be changed. As the key action or statement of a sentence clause is given by its verb, replacing that verb with its antonym will substantially change the clauses meaning. This change will always break semantic entailment, though the pragmatic change may not always be as strong as in the share price example from above. Consider instead the sentences: "They inhaled the smog filled air." vs "They exhaled the smog filled air.". Clearly there is no bi-directional entailment between the two sentences, though pragmatically the meaning is similar – the air the characters were breathing was smog-filled. As the number of verbs changed increase though the meaning generally gets more distant, though multiple negations may apply. For example: "They exhaled the smog emptied air". Thus semantically distinct sentences are constructed. by progressively negating the verbs.

To contrast this, and see the effect of simply changing words, but not structure or meaning, a second method is used. For this evaluation, nouns are replaced with their synonyms. Thus the structure and meaning are preserved, and only the word content is changed.

1.0.4 The Validity of Substitutions

One of the basic tests for determining the part of speech a word of unknown lexical category is to substitute a word of known category in its place, if the sentence is still correct, then the unknown word may have that category[10]. For example if a sentence contains the never seen word "foobar", and substituting "hat" in its place is grammatical, it can be concluded that "foobar" is most likely also a noun. It has been suggested that this is the mechanism by which people learn the parts of speech relations. In general, substitutions of the same parts of speech are valid.

As synonyms and antonyms make up the same parts of speech as their base form, it is grammatically valid to substitute for the original. Substituting a synonym is not expected to change the meaning of the sentence, thus such a substitution creates a new sentence which is semantically equivalent. Indeed this has been proposed as the very definition of synonymy ([11] attributes this definition to Leibniz), though there has been some debate over the nature and degree of this relation[11]. It has been noted in a limited study that subjects could identify whether an adjective or its antonym belong in blanked out section of a sentence[12]. This suggests that identical sentences up to an antonym swap in adjectives, may not occur in natural English. Never-the-less, such a swap is still a part-of-speech matched substitution and thus is grammatical.

2 Background and Motivation

2.1 Semantic Tasks

Many of the key tasks in natural language processing and generation have strong semantic components:

- Translation
- Sentiment Recognition
- Indexing and Retrieval
- Summarization
- Automatic Paraphrasing[1]

2.2 Existing Corpora

2.2.1 Paraphrase Corpora

Closely related to the corpus use for evaluation here are paraphrase corpora such as MSRP. The paraphrase corpora and paraphrase detection tasks present sentences and possible paraphrases. The goal is to determine weather a sentence is a paraphrase of another or not. This implicitly requires working out if the sentences are semantically equivalent. However the corpora are not control to present syntactically similar sentences, thus can not be use to assess to prevalence of syntactic vs semantic information in the vector encoding.

The Paraphrases for Plagiarism corpus (P4P)[13] does have useful information on the types of changes and relations between the compared texts, but is not a sentence orientated corpus. P4P compares short paragraphs, giving more context. While this results in a deeper and more meaningful paraphrasing task, the multiple sentences complicates the evaluation, and renders some embedding methods ([2, 3, 4, 5, 6]) unable to directly process the result. As with other paraphrasing corpora it also does not constrain the structure of sentences.

2.2.2 Finding Structural Matches in Large Corpora

Inspection of large corpora will reveal that structurally identical sentences are rare. While Large corpora contain many thousands of sentences, very few have the same structure, or even same order of POS tags. For example, MASC[14] (Manually Annotated Sub-Corpus of the Open American National Corpus) contains 34,535 sentences, only 4,230 share a set and ordering of POS tags with another. Further: if trivial sentences with 5 or fewer tokens are excluded there are just 634. If sentences where 30% or more of the words are proper nouns, dates, times, or numbers are also excluded (thus removing sentences such as "'Posted by : tomemos | Sunday , 09 May 2010 at 09:16 AM'), only 316 sentences remain. Only a subset of these will be structurally identical when considered under a parse tree.

Thus it is not surprising that MSRPC, which only contains 7800 sentences, contains 5 structurally identical sentences in two structural classes, other than paraphrases or differing only numerically. Even these sentences are highly similar in topic and word usage. Class 1: {'Schering-Plough shares fell 72 cents to close at $ 18.34 on the New York Stock Exchange .', 'Wal-Mart shares rose 16 cents to close at $ 58.28 on the New York Stock Exchange .', 'Xerox shares rose 2 cents to close at $ 11.45 on the New York Stock Exchange .'}, Class 2: {'Jason Giambi capitalized with an RBI single to center .', 'Randall Simon followed with an RBI single to right .'}.

Similarly, there is only one structural match between MSRPC and the MASC corpus: MSRPC contains: "I'm never going to forget this day." which is a structural match for the sentence: "I'm not going to allow that question." from MASC. Thus it is not viable to find naturally occurring sentences of the same structure, but different meaning.

2.2.3 Lexical Substitution Corpora

.

CoInCo (Concepts in Context) is a manually created all-words lexical substitution corpus[15]. Approximately 2500 sentences from a subset of MASC (Manually Annotated Subcorpus of the Open American National Corpus), were given to human annotator, who were tasked with listing all the word synonyms that could replace words. CoInCo only handles synonym substitutions, though an effort could be made to automatically extend it to antonyms (and other lexical relations), by mapping substitution lemmas from CoInCo on to WordNet synsets. According to [15] CoInCo is the only large scale, manually created corpus of its type, though other single word substitution corpora exist.

2.2.4 What is required

To create a corpus capable of assessing the degree of semantic information, as compared to syntactic and word content information, a paraphrase corpus is extended through lexical substitutions maintaining the syntactic structure. The automatically created corpus discussed here, combines features of existing corpora, to make it suited to this fine-grained approach.

3 Creation of Corpus

3.1 Base Paraphrase Corpus

The Microsoft Research Paraphrase Corpus (MSRPC)[8] is used as a base source of sentences, and of the ground truth for their semantic equality. The sentences with paraphrases provided are combined from the training and testing sets. This gives 7,800 sentences for evaluation. Of these 311 are repeated – that is they have multiple different paraphrases specified – no special handling is done for these cases. The sentences are modified by the procedure described below to create semantically equivalent and semantically distinct versions.

3.2 WordNet Lookup

The second primary resource required is a source of synonyms and antonyms for the words being modified. For this WordNet 3.0[16] is used, via the NLTK[17] interface. WordNet organizes the words in to sets of synonymous words called synsets. WordNet only stores the base word form i.e. lemma for each word, but the Morphy tool is provided to lookup the lemma for the words inflicted forms[18]. Thus WordNet can be used to find the replacement words need for the modifications of the sentences.

3.3 Method

3.3.1 Tokenization

The first step is to tokenize the sentence. This was achieved using the NLTK[17] Treebank Word Tokenizer. This tokenizer is based on regular expressions. It splits the sentences in to words, punctuation elements, and also separates contractions: “don’t” becomes “do n’t”. Tokenization is a fairly simple task, accomplished by these regular expressions.

3.3.2 POS Tagging and Restriction of Auxiliary Verbs

The second step is to tag the the word tokens with their parts of speech (POS). The Stanford POS Tagger[19] was used via NLTK[17] interface, to accomplish this. The Stanford POS Tagger, has a 97.24% accuracy on the Penn Treebank Wall Street Journal test set[19]. It is one of the best POS taggers available.

Though it does still make some mistakes, for example, in the sentence: “Cadets were ticketted for drinking alcohol.” drinking is mistaken for a noun, when it should be a verb.

The sentences are tagged with the Penn Treebank tagset[20]. This tagset contains 37 POS tags. Of interest to this work are the tags for nouns and verbs. Valid noun tags for transformation are NN and NNS, which covers singular, mass and plural nouns. The proper noun tags NNP and NNPS are not valid for transformation and are excluded. Valid verbs are marked those marked with all verb tags (VB,VBD,VBG, VBN, VBZ, VBP), except for models (MD) and auxiliaries.

Model and auxiliary verbs are normally inverted by inserting a not after the verb, or equivalently a contraction n’t.[21] This is forbidden by the guideline of not changing the structure in the generated sentences. Thus model verbs and auxiliaries are excluded.

While the POS tagger captures models with the MD tag, the other auxiliary verbs are not caught. [20] states they are to be handles as other verbs. Some of them also have non-auxiliary senses, for example “has” in “he has gone” is an auxiliary[21], but in “He has a dog” it is not. WordNet also does not differentiate auxiliaries from other verbs, and so in both cases suggests that antonym for “has” is “lacks”. To avoid any confusion of this sort auxiliary verbs are blocked using a blacklist. This blocks changes to: "be", "am", "are", "is", "was", "were", "being", "can", "could", "do", "did", "does", "doing", "have", "had", "has", "having", "may", "might", "must", "shall", "should", "will", and "would".

WordNet uses much simpler parts of speech tags, as it only considered lemmas. WordNet POS tags are: noun, verb, adverb and adjective. The Penn Treebank POS tags can be simplified down to them. Further more, the additional information captured in the Penn Treebank Tags, is sufficient to allow recover the full form an a lemma generated from WordNet. (see section 3.3.6).

3.3.3 Phrase Detection

Certain sets of words are best treated as a single unit, this paper, as in [18], will call these collocations, they are also sometimes referred to as continuous dictionary phrases. WordNet Version 3.0 contains 64,331 such collocations.

Consider the word sequences: "chief financial officer", and "police officer".

A synset exist containing "chief_finial_officer" and "CFO", another exist containing: "police_officer", "officer" and "policeman". If collocations were handled as word sequences "chief financial officer" could have the officer replaced, to get: "chief financial policeman", or even: "police officer" could become "police police officer".

If collocations were handled as words, "policeman" could become "police officer". This adds a word, violating the constraint of not changing the sentence's structure. To avoid all these issues entirely, we forbid the substituting for any words in a phrase, as well as forbidding substituting a phrase for any words.

These collocations are detected using a sliding window of width 3 and 2 words across the sentences. The words in the window are then checked to see if they form a collocations known to WordNet. If they do then they are blocked from substitution. This blocks all such collocations of up to length 3.

While this blocks all such continuous phrases it does not handle other kinds. Several other kinds of phrases have been distinguished as having distinct meaning. Such as the skip-bigrams considered by [22]. The necessity of avoiding substitutions with these is less clear. As they are not considered as clear single lexical entity (unlike the continous dictionary phrases), they may be sufficiently handled by word sense disambiguation.

3.3.4 Word Sense Disambiguation

Word sense disambiguation is used to select the correct sense of the word being substituted for – so that it synonym or antonym is the same sense. This is used for example to ensure that the synonym of "bank", as in a financial institution, is not "shore" as in the edge of a body of water[9]. While several methods were considered for performing the word sense disambiguation, it was resolve to use the simple Most Frequent Sense (MFS).

The MFS is a naïve method for determining word sense. It functions by always choosing the word sense that occurs most often – without regard for context, beyond the POS tag of the word. This method is almost certain to make some mistakes, however more sophisticated algorithms have been shown to offer little improvement over it.

BabelNet[23] is a multilingual extension of WordNet that has seen significant use as a target for the evaluation of word sense disambiguation methods. In the English BabelNet WSD subtask at [24], the MFS obtained a F1 score of 66.5%, the best competing algorithm scores 68.6%. Since then, new methods have improved that to [25] 71.5% – which exceeds MFS by 5%. On a similar task at [26], no entry exceeded the MFS F1 score of 67.5%. As simple MFS remains one of the most competitive methods it is used in this system.

3.3.5 Substitution Generation

Noun Synonyms As discussed above, all words sorted in WordNet are stored in synsets[16]. A synset contains many lemmas all with the same semantic meaning. Any given word may have many word-senses, each word-sense for that word belongs to a different synset. All the lemmas within a synset are synonymous. The synonyms of a noun are all the lemmas form the correct synset.

Verb Antonyms Verbs, as nouns are stored in synsets. WordNet stores antonyms on a per-lemma basis. The possible antonyms for a given verb are thus the antonym of the lemma, and also of the any synonyms – lemmas from the same synset. This extension increases the number of antonym lemmas available for any base lemma.

3.3.6 Un-lemmatizing

As the words generated from WordNet are lemmatized, this process must be reversed to restore them to their grammatically correct from for the context. For example, the verbs, "rise", "rose", and "rising" are all mapped to the lemma "rise", from this lemma we generate the antonym: "fall", to put it into the context of the first word, it needs to be mapped to "fall", "fell", and "falling" respectively. Similarly, for nouns lemmatization removes plurals. The needed information for both is captured in the parts of speech tags.

The Penn Treebank POS tags captures the information required to go from a lemma to the correct form. This can be used with the heuristic conjugation and pluralization methods from the Pattern.en library [27] to correct the generated lemma to the appropriate form. The mappings used are shown in figure 1. It can be noted that no processing is done on the Verb, non-3rd person singular present case (VBP), this is because with the exception of the various forms of the verb "be", which are excluded earlier as and auxiliary/model, the non-3rd person singular present is always the same as the base form of the verb[28, p. 84]. These rules have been found to be generally sufficient to ensure grammatical text.

To ensure against any failure in the heuristic rules based un-lemmatization the validity of the generated and un-lemmatized word is checked against the "british-english-insane" collection of words from the Spell Checker Orientated Word Lists (SCOWL)[29]. Any generated words which fail this test are discarded.

During this step, the initial letter capitalization of any generated word is matched to that of the base word. While no method evaluated below makes use of capitals they are preserved for ease of future comparisons with methods which do.

[1]

[1] Pattern.en Verb Conjugation methods use are described in detail in the documentation

Sentence	base word	generated word	POS	POS meaning[20]	Pattern.en method to un-lemmatize
The share price will rise.	**rise**	**fall**	VB	Verb, base form	–
The share price rose.	rose	fell	VBD	Verb, past tense	`conjugate(lemma, tense=PAST)`
The share price is rising.	rising	falling	VBG	Verb, gerund or present participle	`conjugate(lemma, tense=PRESENT, aspect=PROGRESSIVE)`
The share price normally rises.	rises	falls	VBZ	Verb, present tense, 3rd person singular	`conjugate(lemma, tense=PRESENT)`
The share price has risen.	risen	fallen	VBN	Verb, past participle	`conjugate(lemma, tense=PAST, aspect=PROGRESSIVE)`
The share prices rise.	**rise**	**fall**	VBP	Verb, non-3rd person singular present	–
The car is hot.	**car**	**automobile**	NN	Noun, singular or mass	–
The cars are hot.	cars	automobiles	NNS	Noun, plural	`pluralize(lemma)`

Figure 1: POS Tags for various forms of words. The WordNet lemma is in **bold**.

3.3.7 Indefinite Article Correction

A word substituted for may have been preceded by an indefinite article. Depending on the vowel sound at the start of the next word, it may no longer be appropriate[28, p. 1618]. This case is detected and rectified, again making use of the Pattern.en library[27]. This correctly handles phonetic cases such as "an honest man" and "a unit of measure". Case is again preserved. This is the final step in the generation of semantically altered sentences.

3.4 Sequential addition of substitutions

For each sentence, for each verb/noun with a valid substitution that substitution is made progressively. Starting with the base sentence, first one word is changed, and the new sentence added to the corpus, than another and so forth until valid substitutions are exhausted. This is applied separately for nouns and verbs to generate the two test sets. These progressive additions allow investigation into whether simple word content affects reported distances, as well as into the effects of double negation. Examples of this can be seen in the next section.

4 Final Corpus Size

Of the 7800 base sentences with paraphrases, the above method generated many modified sentences for evaluation. The precise counts are in the table below.

Number of substitutions	Nouns Synonyms Sentences	Verb Antonym Sentences	Overlap
1	7168	3061	2829
2	5503	590	449
3	3395	82	46
4	1747	12	4
5	813	2	0
6	309	0	0
7	104	0	0
8	22	0	0
9	9	0	0

5 Examples of Constructed Evaluation Sentences

http://www.clips.ua.ac.be/pages/pattern-en#conjugation

Change	**Sentence**	**Wiki PV-DM Concatenting Model Distance**
Base Sentence	However , other unions including the powerful CGT remained opposed to the reform and demanded the government begin fresh negotiations with them .	0.00
Para-phrase	The powerful CGT and other unions remained opposed to the plans , however , and demanded the government renegotiate the reform with them .	**0.35**
1 Noun Synonym	However , other **brotherhoods** including the powerful CGT remained opposed to the reform and demanded the government begin fresh negotiations with them .	*0.27*
2 Noun Synonym	However , other **brotherhoods** including the powerful CGT remained opposed to the reform and demanded the **authorities** begin fresh negotiations with them .	*0.25*
3 Noun Synonym	However , other **brotherhoods** including the powerful CGT remained opposed to the reform and demanded the **authorities** begin fresh **dialogues** with them .	0.43
1 Verb Antonym	However , other unions including the powerful CGT remained opposed to the reform and **obviated** the government begin fresh negotiations with them .	*0.36*
2 Verb Antonym	However , other unions including the powerful CGT remained opposed to the reform and **obviated** the government **end** fresh negotiations with them .	*0.36*
3 Verb Antonym	However , other unions including the powerful CGT **changed** opposed to the reform and **obviated** the government **end** fresh negotiations with them .	*0.43*
4 Verb Antonym	However , other unions **excluding** the powerful CGT **changed** opposed to the reform and **obviated** the government **end** fresh negotiations with them .	*0.46*

6 Evaluation

A given model may be evaluated on how semantically close, or distantly it places sentences which are semantically equivalent, or not. As discussed, replacing a noun with its synonym produces a semantically equivalent sentence – both the original sentence, and the modified sentence entail the other. Conversely, replacing a verb with its antonym generally will produce a semantically different sentence, often the the modified sentence will entail the converse of the original and visa-verse. As a baseline for a reasonable distance under a model, the distance to the gold standard, semantically equal, paraphrase is computed.

6.0.1 Scoring

By comparing the distance from the base sentence embedding to the paraphrase and modified sentence embeddings, the success of the modal at semantic localization is evaluated. It is assessed on keeping close semantically similar sentences, and distant semantically distinct sentences. A semantically different sentence must be further away that a semantically equivalent one, thus the sentences modified by replacing verbs with their antonyms must be more distant than the semantically equal paraphrase. Conversely, the semantically identical similar form should be no further away than the paraphrase. By counting the portion of the evaluation corpus is correctly placed under this definition, the models semantic semantic accuracy with respect to fine-grained changed is scored.

6.1 The Models

For demonstration purposes, several models are evaluated below.

Distance Distance may be calculated though any method the model specifies. The All the model chosen below, use the cosine distance (or equivalently cosign similarity). This is given by $d(\tilde{u},\tilde{v}) = 1 - \frac{\tilde{u}\cdot\tilde{v}}{\|\tilde{u}\|_2\,\|\tilde{v}\|_2}$, this is between 0, and 2 and is proportional to the cosine of the angle between the embeddings $\tilde{u}$ and $\tilde{v}$.

6.1.1 Unfolding Recursive Auto-Encoder (U-RAE)

The Unfolding Recursive Auto-Encoder is a autoencoder based method. It functions by using the same network to recursively pairwise combine embedded representations, following the parse tree[3]. It's optimization target is to be be able to reverse (unfold) the merges and produce the original sentence. The central folding layer - where the whole sentence is collapsed to a single embedding vector is the representation.

In this evaluation we make use of the pretrained network the authors of [3] have graciously made available[2], full information is available in that paper. It

[2] http://www.socher.org/index.php/Main/DynamicPoolingAndUnfoldingRecursiveAutoencodersForParaphraseDetection

is initialized on the unsupervised Collobert and Weston word embeddings[30], and training on a subset of 150,000 sentences from the gigaword corpus. In the evaluation below the dynamic pooling layer is not used.

6.1.2 Doc2Vec Models

Two new methods, commonly refereed to a doc2vec are described in [7]. For both, we evaluate using the GenSim implementation[31] from the current develop branch.

Both are trained on approximately 12 million sentences from 500 randomly selected wikipedia articles. In both the window size was set to 8 words, and the vectors were of 300 dimensions.

PV-DM Distributed Memory Paragraph Vectors (PV-DM) Doc2Vec document embeddings are based on an extension of Continuous Bag-of-Words word-embedding model[32]. It is trained using a sliding window of words to predict the next word. The softmax predictor network is feed a word-embedding for each word in the window, and an additional embedding vector which is reused for all words in the sentence (called the paragraph vector in original paper). These input embeddings can be concatenated or averaged, in the results show below they were concatenated. During training both word and sentence vectors are allows to vary, in evaluation, the word vectors are locked and the sentence vector trained until convergence.

PV-DBOW Distributed Bag of Words version of Paragraph Vectors (PV-DBOW), is based on the Skip-gram model for word-embeddings, also from [32]. In PV-DBOW a sentence vector is used as the sole input to a neural net. That network is tasked with predicting the words in the sentence.

6.1.3 Baseline: Bag of Words

The traditional bag of words model is presented as a baseline. There is a dimension in each vector for the count of each token, including punctuation. In bag of words, there is a direct relationship between the number of words in-common, the sentence length, and the distance.

7 Results and Discussion

7.1 Noun Synonym Distance vs Paraphrase Distance

Replacing one or more nouns with their synonyms does not break logical entailment. Thus it is expected that a model locating in purely semantic space would locate the modified sentences at least as close to the original sentence, as the paraphrase was.

Number of Changes	**U-RAE**	**PV-DM**	**PV-DBOW**	**BOW**	Number of Evaluation Cases
1	95%	91%	90%	100%	7168
2	91%	84%	82%	99%	5503
3	87%	76%	75%	95%	3395
4	83%	68%	67%	85%	1747
5	79%	58%	57%	66%	813
6	75%	52%	50%	50%	309
7	75%	36%	39%	34%	104
8	73%	36%	27%	18%	22
9	89%	56%	33%	11%	9

Table 1: Portion of the evaluation cases where the distance from the base sentence to the sentence modified by noun substitution was less than, or equal to the distance from the base sentence to the paraphrase. Larger is better.

Unsurprisingly, the BOW model does very well for small number of changes, but is overtaken by the other models as the number of changes increases. This shows that the models do, to some extent recognise the lack of semantic change from the substitution. It can be seen that the U-RAE, outperfroms the PV-DM, which outperforms the PV-DBOW model. This order does correspond the significance each model places on structure, which may be significant.

7.2 Verb Antonym Distance vs Paraphrase Distance

Replacing one of more verbs with their antonyms generally breaks logical entailment. Thus it is expected that a model locating in purely semantic space would place the modified sentences at more distantly than the paraphrases were placed to the original.

Number of Changes	**U-RAE**	**PV-DM**	**PV-DBOW**	**BOW**	Number of Evaluation Cases
1	6%	8%	9%	0%	3061
2	10%	13%	15%	1%	590
3	21%	22%	20%	1%	82
4	17%	17%	58%	0%	12
5	0%	0%	0%	100%	2

Table 2: Portion of the evaluation cases where the distance from the base sentence to the sentence modified by verb antonym substitution was greater than, the distance from the base sentence to the paraphrase. Larger is better.

As is expected the BOW model performs very poorly – it has no mechanism to recognize the additional significance of changing the verb to its opposite. The neural models do perform better, though still poorly. Indicating that some level of logical significance may being induced into the sentence embeddings. It is not clear whether double and quadrupedal negatives impacted the distance, though the decrease in relative distance between three and four substitutions for the U-RAE and PV-DM suggests that may be the case.

7.3 Verb Antonym Distance vs Noun Synonym Distance

The above metrics compared the modified sentences of different entailment to paraphrases with differing word content and structure. To compare solely on the modification to the key words, the distance between the noun synonym and verb antonym modified sentences can be compared. As the noun synonym modified sentences are semantically equivalent to the base sentences, it is expected that they be closer to the base sentence than the semantically different verb antonym sentences.

Number of Changes	**U-RAE**	**PV-DM**	**PV-DBOW**	**BOW**	Number of Evaluation Cases
1	61%	49%	50%	1%	2829
2	58%	50%	43%	4%	449
3	43%	39%	43%	7%	46
4	0%	25%	75%	25%	4

Table 3: Portion of the evaluation cases where the distance from the base sentence to the sentence modified by verb antonym substitution was greater than, the distance from the base sentence to the sentence modified by noun synonym substitution. Larger is better.

The BOW distance for a verb change and a noun change are under most circumstances exactly equal. The exception to this is when the change occurs in a word which results in the number of instances of the same word changing, eg if a sentence previously used the word "share" twice and one of those uses is changed to "part" then the distance is different to if a non-repeated word was changed to another non-repeated word.

8 Conclusion

A method what presented, to evaluate the semantic localization of sentence embedding models. Semantically equivalent sentences are those which exhibit bidirectional entailment – they each imply the truth of the other. Paraphrases are mostly semantically equivalent. Replacing a noun with its synonym creates another sentence which is semantically equivalent to the original. Replacing a verb with its antonym creates a new sentence which is not. By comparing the distances of the generated sentences, and paraphrases from the original sentence, the relationship between semantic closeness and embedding distance can be seen.

The models evaluated using this method show that they are substantially more permanent the a naive bag of words approach there is still significant room for improvement. While these models perform very well at related practical tasks, this new method highlights some of their limitations. It suggests that calling the vector space the models embed into a syntactic space is misleading. The space clearly incorporates elements of syntax and word choice, as well as meaning. This result is not surprising and indeed some papers (including [3]) do refer to the space this way. The new method does make its truth substantially clearer.

8.1 Motivating better use of Semantic Resources in embedding creation

[33] is a improved method over word2vec for determining word embeddings making use of semantic knowledge, potential exists to extend in into the document domain through the doc2vec models presented in [7].

In [1], dependency trees are used instead of the constituency tree used in the original URAE, because of its improved invariance to syntactical changes. This may, by decreasing the impact of syntax create models with a greater emphasis on semantic placement.

References

[1] M. Iyyer, J. Boyd-Graber, and H. D. III, "Generating sentences from semantic vector space representations," in *NIPS Workshop on Learning Semantics*, 2014.

[2] R. Socher, J. Bauer, C. D. Manning, and A. Y. Ng, "Parsing with compositional vector grammars," in *ACL*, 2013.

[3] R. Socher, E. H. Huang, J. Pennington, A. Y. Ng, and C. D. Manning, "Dynamic pooling and unfolding recursive autoencoders for paraphrase detection," in *Advances in Neural Information Processing Systems 24*, 2011.

[4] R. Socher, C. C. Lin, C. Manning, and A. Y. Ng, "Parsing natural scenes and natural language with recursive neural networks," in *Proceedings of the 28th international conference on machine learning (ICML-11)*, pp. 129–136, 2011.

[5] R. Socher, C. D. Manning, and A. Y. Ng, "Learning continuous phrase representations and syntactic parsing with recursive neural networks," in *Proceedings of the NIPS-2010 Deep Learning and Unsupervised Feature Learning Workshop*, pp. 1–9, 2010.

[6] R. Socher, J. Pennington, E. H. Huang, A. Y. Ng, and C. D. Manning, "Semi-supervised recursive autoencoders for predicting sentiment distributions," in *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2011.

[7] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pp. 1188–1196, 2014.

[8] W. B. Dolan and C. Brockett, "Automatically constructing a corpus of sentential paraphrases," in *Third International Workshop on Paraphrasing (IWP2005)*, Asia Federation of Natural Language Processing, 2005.

[9] C. Manning and H. Schütze, *Foundations of Statistical Natural Language Processing*. MIT Press, 1999.

[10] A. Radford, *Minimalist Syntax: Exploring the Structure of English*. Cambridge Textbooks in Linguistics, Cambridge University Press, 2004.

[11] G. A. Miller and W. G. Charles, "Contextual correlates of semantic similarity," *Language and cognitive processes*, vol. 6, no. 1, pp. 1–28, 1991.

[12] W. G. Charles and G. A. Miller, "Contexts of antonymous adjectives," *Applied Psycholinguistics*, vol. 10, pp. 357–375, 9 1989.

[13] A. Barrón-Cedeño, M. Vila, M. A. Martí, and P. Rosso, "Plagiarism meets paraphrasing: Insights for the next generation in automatic plagiarism detection," *Computational Linguistics*, vol. 39, no. 4, pp. 917–947, 2013.

[14] R. J. Passonneau, C. Baker, C. Fellbaum, and N. Ide, "The masc word sense sentence corpus," in *Proceedings of LREC*, 2012.

[15] G. Kremer, K. Erk, S. Padó, and S. Thater, "What substitutes tell us-analysis of an" all-words" lexical substitution corpus," in *Proceedings of EACL*, 2014.

[16] G. A. Miller, "Wordnet: a lexical database for english," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[17] S. Bird, E. Klein, and E. Loper, *Natural language processing with Python*. " O'Reilly Media, Inc.", 2009.

[18] R. I. Tengi, *WordNet: an electronic lexical database, The MIT Press, Cambridge, Massachusetts*, ch. Design and implementation of the WordNet lexical database and searching software, p. 105. 1998.

[19] K. Toutanova, D. Klein, C. D. Manning, and Y. Singer, "Feature-rich part-of-speech tagging with a cyclic dependency network," in *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pp. 173–180, Association for Computational Linguistics, 2003.

[20] B. Santorini, "Part-of-speech tagging guidelines for the penn treebank project (3rd revision)," 1990.

[21] A. Radford, *English syntax: An introduction*. Cambridge University Press, 2004.

[22] W. Yin and H. Schütze, "An exploration of embeddings for generalized phrases," *ACL 2014*, p. 41, 2014.

[23] R. Navigli and S. P. Ponzetto, "Babelnet: Building a very large multilingual semantic network," in *Proceedings of the 48th annual meeting of the association for computational linguistics*, pp. 216–225, Association for Computational Linguistics, 2010.

[24] R. Navigli, D. Jurgens, and D. Vannella, "Semeval-2013 task 12: Multilingual word sense disambiguation," in *Second Joint Conference on Lexical and Computational Semantics (* SEM)*, vol. 2, pp. 222–231, 2013.

[25] P. Basile, A. Caputo, and G. Semeraro, "An enhanced lesk word sense disambiguation algorithm through a distributional semantic model," in *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, (Dublin, Ireland), pp. 1591–1600, Dublin City University and Association for Computational Linguistics, August 2014.

[26] A. Moro and R. Navigli, "Semeval-2015 task 13: Multilingual all-words sense disambiguation and entity linking," *Proceedings of SemEval-2015*, 2015.

[27] T. De Smedt and W. Daelemans, "Pattern for python," *The Journal of Machine Learning Research*, vol. 13, no. 1, pp. 2063–2067, 2012.

[28] R. Huddleston and G. Pullum, *The Cambridge Grammar of the English Language*. Cambridge textbooks in linguistics, Cambridge University Press, 2002.

[29] K. Atkinson, "Scowl (spell checker orientated word lists)," 2011.

[30] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proceedings of the 25th international conference on Machine learning*, pp. 160–167, ACM, 2008.

[31] R. Řehůřek and P. Sojka, "Software Framework for Topic Modelling with Large Corpora," in *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, (Valletta, Malta), pp. 45–50, ELRA, May 2010. `http://is.muni.cz/publication/884893/en`.

[32] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[33] M. Yu and M. Dredze, "Improving lexical embeddings with semantic knowledge," in *Association for Computational Linguistics (ACL)*, pp. 545–550, 2014.