Measuring Semantic Relations between Human Activities

Steven R. Wilson and **Rada Mihalcea**
University of Michigan
{steverw|mihalcea}@umich.edu

Abstract

The things people do in their daily lives can provide valuable insights into their personality, values, and interests. Unstructured text data on social media platforms are rich in behavioral content, and automated systems can be deployed to learn about human activity on a broad scale if these systems are able to reason about the content of interest. In order to aid in the evaluation of such systems, we introduce a new phrase-level semantic textual similarity dataset comprised of human activity phrases, providing a testbed for automated systems that analyze relationships between phrasal descriptions of people's actions. Our set of 1,000 pairs of activities is annotated by human judges across four relational dimensions including similarity, relatedness, motivational alignment, and perceived actor congruence. We evaluate a set of strong baselines for the task of generating scores that correlate highly with human ratings, and we introduce several new approaches to the phrase-level similarity task in the domain of human activities.

1 Introduction

Our everyday behaviors say a lot about who we are. The things we do are related to our personality (Ajzen, 1987), values (Rokeach, 1973), interests (Goecks and Shavlik, 2000), and what we are going to do next (Ouellette and Wood, 1998). While we cannot always directly observe what people are doing on a day-to-day basis, we have access to a large number of unstructured text sources that describe real-world human activity, such as news outlets and social media sites. Fiction and nonfiction writings often revolve around the things that people do, and even encyclopedic texts can be rich in descriptions of human activities. Although many common sources of text contain human activities, reasoning about these activities and their relationships to one another is not a trivial task. Descriptions of human actions are fraught with ambiguity, subjectivity, and there are multitudinous lexically distinct ways to express highly similar events. If we want to gain useful insights from these data, it should be beneficial to develop effective systems that can successfully represent, compare, and ultimately understand human activity phrases.

In this paper, we consider the task of automatically determining the strength of a relationship between two human activities,[1] which can be helpful in reasoning about texts rich with activity-based content. The relationship between activities might be similarity in a strict sense, such as *watching a film* and *seeing a movie*, or a more general relatedness, such as the relationship between *turn on an oven* and *bake a pie*. Another way to categorize a pair of activities is by the degree to which they are typically done with a similar motivation, like *eating dinner with family* and *visiting relatives*. Or, in order to uncover which other behaviors a person is likely to exhibit, it might be useful to determine how likely a person might be to do an activity given some information about previous real-world actions that they have taken.

Success on our proposed task will be a valuable step forward for multiple lines of research, especially within the computational social sciences where human behavior and its relation to other variables (e.g., personality traits, personal values, or political orientation) is a key focus. Since the language human activities is so varied, it is not enough to store exact representations of activity

[1] Throughout this paper, we use the word "activity" to refer to what a person does or has done. Unlike the typical use of this term in the computer vision community, in this paper we use it in a broad sense, to also encompass non-visual activities such as "make vacation plans" or "have a dream".

phrases that are unlikely to appear many times. It would be useful to instead have methods that can automatically find related phrases and group them based on one (or more) of several dimensions of interest. Moreover, the ability to automatically group related activities will also benefit research in video-based and multimodal human activity recognition where there is need for inference about activities based on their relationships to one another.

Reasoning about the relationships between activity phrases brings with it many of the difficulties often associated with phrase-level semantic similarity tasks. It is not enough to know that the two phrases share a root verb, as the semantic weight of verbs can vary, such as the word "go" in the phrases *go to a bar* and *go to a church*. While these phrases have high lexical overlap and are similar in that they both describe a traveling type of activity, they are usually done for different motivations and are associated with different sets of other activities. In this case, we could only consider the main nouns (i.e., "bar" and "church"), but that approach would cause difficulties when dealing with other phrases such as *sell a car* and *drive a car*, which both involve an automobile but describe dissimilar actions. Therefore, successful systems should be able to properly focus on the most semantically relevant tokens with a phrase. A final challenge when dealing with human activity phrase relations is evaluation. There should be a good way to determine the effectiveness of a system's ability to measure relations between these types of phrases, yet other commonly used semantic similarity testbeds (e.g., those presented in various Semeval tasks (Agirre et al., 2012, 2013; Marelli et al., 2014)) are not specifically focused on the domain of human activities. Currently, it is unclear whether or not the top-performing systems on general phrase similarity tasks will necessarily lead to the best results when looking specifically at human activity phrases.

To address these challenges, we introduce a new task in automatically identifying the strength of human activity phrase relations. We construct a dataset consisting of pairs of activities reportedly performed by actual people. The pairs that we have collected aim specifically to showcase diverse phenomena such as pairs containing the same verb, a range of degrees of similarity and relatedness, pairs unlikely to be done by the same type of person, and so forth. These pairs are each annotated by multiple human judges across the following four dimensions:

- **Similarity:** The degree to which the two activity phrases describe the same thing. Here we are seeking semantic similarity in a strict sense. Example of high similarity phrases: *to watch a film* and *to see a movie*.
- **Relatedness:** The degree to which the activities are related to one another. This relationship describes a general semantic association between two phrases. Example of strongly related phrases: *to give a gift* and *to receive a present*.
- **Motivational Alignment:** The degree to which the activities are (typically) done with similar motivations. Example of phrases with potentially similar motivations: *to eat dinner with family members* and *to visit relatives*.
- **Perceived Actor Congruence:** The degree to which the activities are often done by the same type of person. Put another way, does knowing that a person often performs an activity increase human judges' expectation that this person will also often do a second activity? Example of activities that might be expected to be done by the same person: *to pack a suitcase* and *to travel to another state*.

These relational dimensions were selected to cover a variety of types of relationships that may hold between two activity phrases. This way, automated methods that capture slightly different notions of similarity between phrases will potentially be able to perform well when evaluated on different scales. While the dimensions are correlated with one another, we show that they do in fact measure different things. We provide a set of benchmarks to show how well previously successful phrase-level similarity systems perform on this new task. Furthermore, we introduce several modifications and novel methods that lead to increased performance on the task.

2 Related Work

Semantic similarity tasks have been recently dominated by various methods that seek to embed segments of text as vectors into some high-dimensional space so that comparisons can be made between them using cosine similarity or other vector based metrics. While word embeddings have existed in various forms in the past (Church and Hanks, 1990; Bengio et al., 2003),

many approaches used today draw inspiration directly from shallow neural network based models such as those described in (Mikolov et al., 2013).[2] In the common skip-gram variant of these neural embedding models, a neural network is trained to predict a word given its context within some fixed window size. (Levy and Goldberg, 2014a) and (Bansal et al., 2014) extended the idea of context to incorporate dependency structures into the training process, leading to vectors that were able to better capture certain types of long-distance syntactic relationships. One of the major strengths of neural word embedding methods is that they are able to learn useful representations from extremely large corpora that can then be leveraged as a source of semantic knowledge on other tasks of interest, such as predicting word analogies (Pennington et al., 2014) or the semantic similarity and relatedness of word pairs (Huang et al., 2012).

Researchers have taken the powerful semi-supervised ability of these word embedding methods to aid in tasks at the phrase-level, as well. The most straightforward way to accomplish a phrase-level representation is to use some binary vector-level operation to compose pre-trained vector representations of individual words that belong to a phrase (Mitchell and Lapata, 2010). Other methods have sought to directly find embeddings for larger sequences of words, such as (Le and Mikolov, 2014) and (Kiros et al., 2015).

Semantic textual similarity tasks are often evaluated by computing the correlation between human judgements of similarity and machine output. The wordsim353 (Finkelstein et al., 2001) and simlex999 (Hill et al., 2016) resources provide a set of human annotated pairs of words, labeled for similarity and/or general association. Simverb-3500 (Gerz et al., 2016) was introduced to provide researchers with a testbed for verb relations, a specific yet important class of words that was less common in earlier word-level similarity data sets. SemEval has released a series of semantic text similarity tasks at varying levels of granularity, ranging from words to entire documents, such as the SICK (Sentences Involving Compositional Knowledge) dataset (Marelli et al., 2014) which is specifically crafted to evaluate the ability of systems to effectively compose individual word semantics in order to achieve the overall meaning of a sentence. While many of these evaluation sets contain human activities to some degree, they also have contain other types of words or phrases due to the way in which they were created. For example, SICK contains actions done by animals such as *follow a fish*. Similarly, Simverb-3500 contains verbs that don't necessarily describe human activities, like *chirp* and *glow*, and does not contain phrase-level activities.

Several recent works have raised concerns over the standard evaluation approaches used in semantic textual similarity tasks. One potential issue is the use of inadequate metrics depending on the task that a practitioner is interested in tackling. While the Pearson correlation between human-judged similarity scores and predicted outputs is often used, this type of correlation can be misleading in the presence of outliers or nonlinear relationships (Reimers et al., 2016). Remiers et al. propose a framework for selecting a metric for semantic text similarity tasks, which we take into consideration when selecting our evaluation metric. Additionally, correlation with human judgments does not always give a good indication of success on some downstream applications, the human ratings themselves are somewhat subjective, and statistical significance is rarely reported in comparisons of word embedding methods (Faruqui et al., 2016). However, our goal in this work is not to evaluate the overall quality of distributional semantic models, but to find a method that has high utility in the domain of human activity relations, and so we do rely on comparisons with human judges as a means of assessment.

3 Data Collection and Annotation

One potential source of data containing people's self-reported descriptions of their activities is social media platforms, but these data are noisy and require preprocessing steps that, being imperfect, may propagate their own errors into the resulting data. In order to get a set of cleaner activities that people might actually talk about doing, we directly asked Amazon Mechanical Turk (AMT) workers to write short phrases describing five activities that they had done in the past week. We collected data from 1,000 people located in the United States for a total of 5,000 activities. The activity phrases were then normalized by converting them to their infinitive form (without a preceding "to"), correcting spelling errors, removing punctuation, and converting all characters to lowercase.

[2] It is worth noting that (Levy and Goldberg, 2014b) show that these embeddings are actually implicitly factorizing a shifted version of a more traditional PMI word-context matrix, which is similar to the word co-occurrence matrix factorization approach used in (Pennington et al., 2014)).

Activity	Prompt	User Selection
pay the phone bill	an activity that is EXTREMELY SIMILAR	pay one's student loan bill
play softball	an activity that is SOMEWHAT SIMILAR	go bowling
take a bath	an activity that uses the SAME VERB	take care of one's ill spouse
smoke	an activity that is RELATED, but not necessarily SIMILAR	get sick and go to the doctor
go out for ice cream	an activity that is NOT AT ALL SIMILAR	cash a check

Table 1: Examples of activity/prompt pairs and the corresponding activities that were selected by the annotators given the pair.

After removing duplicate entries (about 2,000) and any phrases referring specifically to doing work on AMT (e.g., those containing the tokens mTurk or Turking, about 150 cases), we were left with a set of 2,909 unique activity phrases.

We acknowledge that this methodology introduces some bias since the workers all come from the United States, and it is therefore likely that our set of activity phrases describe things that are more commonly done by Americans than people from other regions. Furthermore, primacy and recency effects (Murdock Jr, 1962) may bias the types of items listed toward things done in the morning or just before logging onto the AMT platform. Based on this, we expect that our set of activities is not necessarily a representative sample of everything that people might do, but they are still descriptions of actual activities that real humans have done and are useful for our task.

3.1 Forming Pairs of Activities

Next, we sought to create pairs of activities that showcase a variety of relationship types, including varying degrees of similarity and relatedness. To achieve this, we turned to another group to human annotators. After reading through a document which oriented them to the task, the annotators were given the full list of activities in addition to a subset of randomly selected activity phrases. Each of these phrases was randomly paired with one of several possible prompts (see Table 1 for examples) which instructed the annotators how they should select a second activity phrase from the complete list in order to form a pair. Each prompt was sampled an equal number of times in order to make sure that the final set of pairs exhibited various types of relationships to the same degree. All annotators had access to a searchable copy of the full list, but the order of the activities was shuffled each time in order to avoid potential bias from the annotators selecting phrases near the top of the list, and a new shuffled version of the list was given after every 25 pairs created. While a suitable second activity phrase was not always present (e.g., no phrase in our dataset matches "an activity that uses the SAME VERB" as *choreograph a dance*), it is not crucial that all of these pairs fit the prompts exactly since these are only intended to approximate various phenomena, and the final annotations will be done without the knowledge of the prompts used to generate the pairs. In total, 12 unique annotators created 1,000 pairs of phrases.

3.2 Annotating Activity Pairs

All of the activity phrase pairs were uploaded to AMT in order to be labeled. For each pair, ten workers were asked to rate the similarity, relatedness, motivational alignment, and perceived actor congruence on a 5-point Likert-type scales (a total of 40,000 annotated data points). The workers were given a set of instructions that included descriptions of the four types of relationships with examples, including cases in which a pair might be related but not similar, motivationally aligned but not similar, etc. By asking the same set of people to label all four relational dimensions for a given pair, we hoped to make them cognizant of the differences between the scales.

The first three relationships were prompted for using the form: "To what degree are the two activities similar/related/of the same motivation?" and were coded as 0 (e.g., for responses of "not at all similar") and the integers 1-4 with 4 representing the strongest relationship. Perceived actor congruence was solicited for using the form: "Person A often does *activity 1*, while person B rarely does *activity 1*. Who would you expect to do *activity 2* more often?" with choices ranging from "Most likely Person B" to "Most likely Person A." Perceived actor congruence ranges from -2 to 2 and has the lowest score when Person B is chosen and the highest when Person A is chosen. A score of 0 on this scale means that judges were unable to determine whether Person A or Person B would be more likely to perform the action being asked about (i.e., *activity 2*). Each individual Human Intelligence Task (HIT) posted to AMT required an annotator to label 25 pairs so that we could reliably

Activity 1	Activity 2	SIM	REL	MA	PAC
go jogging	lift weights	1.67	2.22	2.89	1.11
read to one's kids	go to a bar	0	0	0	-1.29
take transit to work	commute to work	3.38	3.5	3.38	0.5
make one's bed	organize one's desk	0.58	1.29	1.57	0.71

Table 2: Sample activity phrase pairs and average human annotation scores given for the four dimensions: Similarity (SIM), Relatedness (REL), Motivational Alignment (MA) and Perceived Actor Congruence (PAC). SIM, REL, and MA are on a 0-4 scale, while PAC scores can range from -2 to 2.

	SIM	REL	MA	PAC
SIM	1.000	.962	.928	.735
REL		1.000	.932	.776
MA			1.000	.738
PAC				1.000

Table 3: Spearman correlations between the four relational dimensions: Similarity (SIM), Relatedness (REL), Motivational Alignment (MA) and Perceived Actor Congruence (PAC).

compute agreement, and a worker could complete as many HITs as they desired.

To remove potential spammers (annotators seeking quick payment who do not follow the task instructions), we first eliminated all annotations by any AMT workers who left items blank or selected the same score for every item for any of the four relationships in any of their completed HITs. Then, inter-annotator agreement was computed by calculating the Spearman correlation coefficient ρ between each annotator's scores and the average scores of all other AMT workers who completed the HIT, excluding those already thrown out during spammer removal. We then removed any annotations from workers whose agreement scores were more than three standard deviations below the mean agreement score for the HIT under the assumption that these workers were not paying attention to the pairs when selecting scores.

The final scores for each pair were assigned by taking the average AMT worker score for each relationship type. Some sample activities and their ratings are shown in Table 2. Averaged across all four relationship types, there is a good level of inter-annotator agreement at $\rho = .720$ (recomputed after spammer removal). The highest levels of agreement were found for similarity and relatedness ($\rho = .768$ for both), which is to be expected as these are somewhat less subjective than motivational alignment ($\rho = .745$) and perceived actor congruence ($\rho = .620$). These agreement scores can be treated as an upper bound for performance on this task; achieving a score higher than these would mean that an automated system is as good at ranking activity phrases as the average human annotator.

SIM	REL	Activity 1	Activity 2
↑	↑	call one's mom	call dad
↑	↓	-	-
↓	↑	rake leaves	mow the lawn
↓	↓	go for a run	shop at a thrift store
SIM	**MA**	**Activity 1**	**Activity 2**
↑	↑	check facebook	check twitter
↑	↓	drive to missouri	go on a road trip
↓	↑	write a romantic letter	kiss one's spouse
↓	↓	cut firewood	trim one's beard
SIM	**PAC**	**Activity 1**	**Activity 2**
↑	↑	make a cherry pie	bake a birthday cake
↑	↓	have dinner with friends	eat by oneself
↓	↑	go to the gym	take a shower
↓	↓	read a novel	go to a party
REL	**MA**	**Activity 1**	**Activity 2**
↑	↑	gamble	go to the casino
↑	↓	go swimming	clean the pool
↓	↑	clean out old email	vacuum the house
↓	↓	study abstract algebra	go to the state fair
REL	**PAC**	**Activity 1**	**Activity 2**
↑	↑	eat cereal	eat a lot of food
↑	↓	homeschool one's child	drive one's child to school
↓	↑	cut the grass	talk to neighbors
↓	↓	eat at a restaurant	cook beans from scratch
MA	**PAC**	**Activity 1**	**Activity 2**
↑	↑	go to the dentist	brush one's teeth
↑	↓	take the train to work	drive to work
↓	↑	walk one's dog	walk to the store
↓	↓	read	watch football all day

Table 4: Activity pairs from our dataset highlighting stark differences between the four relational dimensions. For each dimension, ↑ refers to phrases rated at least one full point above the middle value along the Likert scale, while ↓ indicates a score at least one full point below the middle value. No pairs with high similarity and low relatedness exist in the data.

3.3 Relationships Between Dimensions

While the four relationship types being measured are correlated with one another (Table 3.2), there were certainly cases in which humans gave different scores for each relationship type to the same pair which shed light on the nuanced differences between the dimensions. (Table 4). Therefore, it is not necessarily the case that the best method for capturing one dimension is also the most corre-

lated with human judgements across all four dimensions. However, it appears that similarity, relatedness, and motivational alignment are more highly correlated with one another than perceived actor congruence.

4 Methods

To determine how well automated systems are able to model humans' judgements of similarity, relatedness, motivational alignment, and perceived actor congruence, we evaluate a group of semantic textual similarity systems that are either commonly used or have shown state-of-the-art results. Each method takes two texts of arbitrary length as input and produces a continuous valued score as output. All of the methods are trained on outside data sources and many have been proposed as generalized embeddings that can be successful across many tasks. The methods we assess fall into three different categories: Composed Word-level Embeddings, Graph-based Embeddings, and Phrase-level Embeddings.

Activity Phrase Pre-processing. For the first two classes of methods, we experiment with several variations in the set of words being passed to the model as input in order remove the influence of potentially less semantically important words. We do not apply these pre-processing approaches to the phrase-level embedding methods since those methods are designed specifically to operate on entire phrases (as opposed to the bag-of-words view that the other methods take). The five variations of each phrase we consider are:

Full: The original phrase in its entirety.

Simplified: Starting with the Full phrase, we remove several less semantically relevant edges from a dependency parse[3] of the phrase, including the removal of determiners, coordinating conjunctions, adjectival modifiers, adverbs, and particles. This step is somewhat similar to performing stopword removal. For example, this filtering step would result in the bag of words containing "clean", "living" and "room" for full phrase: *clean up the living room*.

Simplified - Light Verbs: Starting with the Simplified set of words, we remove the root verb of the activity if it is not the only word in the Simplified phrase and if it belongs to the following list of semantically light verbs (Kearns, 1988): "go", "make", "do", "have", "get", "give", "take", "let", "come", and "put". This means that we would convert the phrase *go get a tattoo* to just *get a tattoo*, but *read a novel* would retain its verb and become *read novel* (i.e., it will remain equivalent to the Simplified variation).

Simplified - All Verbs: To compare against the effect of removing light verbs, this approach takes the Simplified phrase and removes the root verb unless the Simplified phrase only contains that one word. Performing this filtering step would convert the phrase *cook a sausage* to simply *sausage*.

Core: This method seeks to reduce the phrase to a single core concept. In many cases, this means simply using the root verb from the dependency parse. So, we might represent the phrase "clean up the living room" using only the word embedding for "clean". However, we acknowledge that semantically light verbs such as "go", "have", and "do" would not adequately represent an entire activity, and so in the case of light verbs we instead select either the direct object or a nominal modifier that is connected to the root verb. If the noun selected as the core concept has another noun attached by a compound relationship, we also include that noun. This means, for example, that we would represent the phrase "go to an amusement park" as just "amusement park" when we are considering just the core concept.

4.1 Composed Word-level Embeddings

The methods in this section are based on word-level embeddings trained on some outside data. Since they operate at a word level, we apply a composition function to the words in a given phrase in order to achieve an embedding for the phrase. We tested both the arithmetic mean and element-wise multiplication for composition functions, but the former gave better performance and thus we do not report results found when using the element-wise product. Given an aggregate embedding for a phrase, we generate a score for each pair of activity phrases by computing the cosine similarity between the embeddings for the two phrases. We consider the following word-level methods:

Wiki-BOW: Skip Gram with Negative Sampling Word Embeddings trained on Wikipedia data using a context window of size 2 (Wiki-BOW2) and size 5 (Wiki-BOW5). These vectors are the same ones used in (Levy and Goldberg, 2014a).

Wiki-DEP: Skip Gram with Negative Sampling Word Embeddings trained on Wikipedia data with dependency-based contexts (Wiki-DEP) from (Levy and Goldberg, 2014a).

GoogleNews: Skip Gram with Negative Sampling Word Embeddings trained on the Google News

[3] We use the dependency parser from Stanford CoreNLP (http://stanfordnlp.github.io/CoreNLP/).

corpus from (Mikolov et al., 2013).

Paragram: Embeddings trained on the Paraphrase Database (Ganitkevitch et al., 2013) by fitting the embeddings so that the difference between the cosine similarity of actual paraphrases and that of negative examples is maximized(Wieting et al., 2015). We use the Paragram-Phrase XXL embeddings combined with the Paragram-SL999 embeddings, the latter of which has been tuned on SimLex999 (Hill et al., 2016). We also use a variation of Paragram Embeddings that employs counter fitting (Paragram-CF). This method further tunes the Paragram embeddings to capture a more strict sense of similarity rather than general association between words. This is accomplished via optimization with the goal of increasing the vectorspace differences between known antonyms and altering synonym embeddings to make them more similar to one another (Mrkšić et al., 2016).

Nondistributional vectors: Highly sparse vectors that encode a huge number of binary variables that capture interesting features about the words such as part of speech, sentiment, and supersenses (Faruqui and Dyer, 2015).

4.1.1 Graph-Based Embeddings

We also experiment with approaches that seek to incorporate higher order relationships between activity phrases by building semantic graphs that can be exploited to discover relations that hold between the phrases. Each graph G is of the form $G = (V, E)$ where V is a set of human activity phrases and E is some measure of semantic similarity, which is computed differently depending on the graph type. We run Node2vec (Grover and Leskovec, 2016) using the default settings to generate an embedding for each node in the graph and then measure the cosine similarity between nodes (phrases) to get the final system output. The types of graphs that we use are:

Similarity Graph: We first generate a fully connected graph of all activities in our dataset using a high performing semantic similarity method (Paragram in this case) as a way to generate edge weights. Next, we prune all edges with a weight less than some threshold. The results reported here use a threshold of .5 (on a 0-1 continuous scale). We also tried threshold values of .3, .4, and .6., but found them to produce inferior results for all dimensions.

People Graph: For each activity, we know at least four other activities that were done by the same person because each person submitted five activities. We add an unweighted edge to the graph for each pair of activities that were done by the same person. On its own, this graph does not have enough information to be competitive, so we only report results for the combined graph.

Combined Graph: Here, we combine information from both the Similarity Graph and the People Graph. Since the People Graph is unweighted, we follow the approach used in (Tripodi and Pelillo, 2016) and compute the average weight of all edges in the Similarity Graph and assign this weight to all edges in the People Graph. We then add the edge weights of the two graphs, treating nonexistant edges as edges with weight 0.

4.1.2 Phrase-level Embeddings

The methods in this section are designed to create an embedding directly from phrases of arbitrary length. Since these approaches are tailored toward phrases in their entirety, we do not evaluate them on the pre-processed variations of the phrases in our dataset. The phrase-level approaches we consider are:

Skip-thoughts vectors: This encoder-decoder model induces sentence level vectors by learning to predict surrounding sentences of each sentence in a large corpus of books (Kiros et al., 2015). The encoder is a recurrent neural network (RNN) which creates a vector from the words in the input sentence, and the RNN decoder generates the neighboring sentences. The model also learns a linear mapping from word-level embeddings into the encoder space to handle rare words that may not appear in the training corpus.

Charagram embeddings: Embeddings that represent character sequences (i.e., words or phrases) based on an elementwise nonlinear transformation of embeddings of the character n-grams that comprise the sequence (Wieting et al., 2016). Here we use the pre-trained charagram-phrase model.

5 Results

Because human annotations should fall on an ordinal scale rather than a ratio scale, it would not be fair to directly compare the average values human judges gave to the systems' output. Rather, the systems should be evaluated based on their ability to rank the set of phrases in the same order as the ranking given by the average human annotations scores for each dimension. Therefore, we calculate the Spearman Rank correlation between scores given by the automated systems and the human judges our final score for each system. In a previous study of evaluation metrics for intrinsic semantic textual similarity tasks, this metric was

	Method	SIM	REL	MA	PAC
Full phrase	Wiki-BOW-2	.434	.395	.383	.230
	Wiki-BOW-5	.480	.446	.431	.268
	Wiki-DEP	.388	,346	,339	.191
	GoogleNews	.550	.528	.514	.343
	Paragram	.578	.554	.530	.363
	Paragram-CF	.487	.455	.434	.276
	Sim Graph	.508	.489	.460	.330
	+ People Graph	.520	.502	.467	.340
	Skip-thoughts	.435	.408	.411	.276
	Charagram	.566	.550	.520	.381*
Simplified	Wiki-BOW-2	.532	.501	.475	.316
	Wiki-BOW-5	.563	.537	.507	.342
	Wiki-DEP	.499	.463	.443	.284
	GoogleNews	.606*	.582*	.552*	.383*
	Paragram	.616*	.594*	.560*	.397*
	Paragram-CF	.617*	.592*	.556*	.394*
	Sim Graph	.533	.520	.478	.340
	+ People Graph	.543	.533	.492	.350
- Light Verbs	Wiki-BOW-2	.523	.500	.481	.315
	Wiki-BOW-5	.565	.545	.522	.350
	Wiki-DEP	.484	.457	.443	.280
	GoogleNews	.618*	.599*	.577*	.394*
	Paragram	**.639***	**.623***	**.595***	.418*
	Paragram-CF	.637*	.618*	.587*	.416*
	Sim Graph	.577	.572	.534	.360
	+ People Graph	.584	.576	.535	.375
- All Verbs	Wiki-BOW-2	.434	.436	.419	.334
	Wiki-BOW-5	.482	.492	.469	.381*
	Wiki-DEP	.395	.392	.379	.290
	GoogleNews	.529	.542	.515	.425*
	Paragram	.547	.566	.541	**.445***
	Paragram-CF	.522	.538	.510	.435*
	Sim Graph	.417	.452	.417	.363
	+ People Graph	.433	.468	.432	.379
Core Only	Wiki-BOW-2	.360	.321	.316	.153
	Wiki-BOW-5	.402	.364	.363	.184
	Wiki-DEP	.319	.276	.274	.108
	GoogleNews	.436	.394	.393	.209
	Paragram	.444	.401	.402	.223
	Paragram-CF	.438	.397	.397	.225
	Sim Graph	.330	.281	.291	.146
	+ People Graph	.334	.283	.293	.134
	Human Agree.	*.768*	*.768*	*.745*	*.620*

Table 5: Spearman correlation between phrase similarity methods and human annotations across four annotated relations: Similarity (SIM), Relatedness (REL), Motivational Alignment (MA) and Perceived Actor Congruence (PAC). Top performing methods for each dimension are in bold font. * indicates correlation coefficient is not statistically significantly lower than the best method for that relational dimension ($\alpha = .05$).

recommended for tasks in which the ranking of all items is important (Reimers et al., 2016). Results for all methods using all phrase variations are shown in Table (Table 5).

For our dataset, Paragram in the Simplified - Light Verbs setting gives the best results for similarity, relatedness, and motivational alignment. It is somewhat expected that the same method has the best performance for these three dimensions as they are strongly correlated with one another. Paragram in the Simplified - All Verbs setting gives the best result on perceived actor congruence. We can see that removing light verbs is a helpful step for most methods when trying to predict similarity, relatedness, and motivational alignment indicating that light verbs mostly add noise to the overall meaning of the phrases. Interestingly, the best results for pereceived actor congruence come when ignoring all root verbs in longer phrases. This was a filtering step that led to decreased performance when ranking across the other three dimensions. This suggests that for determining perceived actor congruence, the context of the action found within a phrase is more important than the action itself. Based on statistical significance testing (Z-test using Fisher r-z transformation, single-tailed), however, we cannot be confident that all of these results will hold for larger sets of human activity phrase pairs, as several other methods had scores that were not found to be significantly lower than the best methods.

6 Conclusion

In this paper, we addressed the task of measuring semantic relations between human activity phrases. We introduced a new dataset consisting of human activity pairs that have been annotated based on their similarity, relatedness, motivational alignment, and perceived actor congruence. Using this dataset, we evaluated a number of semantic textual similarity methods to automatically determine scores for each of the four dimensions, and found that similarity between averaged paragram embeddings of the simplified phrases with light verbs removed was most highly correlated with human judgements of similarity, relatedness, and motivational alignment. The method that yielded the best result for the perceived actor congruence dimension also used the paragram embeddings, but when averaged across the simplified phrases with all verbs removed.

We believe there is still plenty of room for improvement on this task, and we hope that the release of our data will encourage greater participation on this task. Future work should explore methods to handle more subtle semantic differences between activities that we noticed are often missed by the automated methods including the effects of function words and polysemy. It should also be helpful to learn better weight-based composition methods (e.g., those proposed in (Yu and Dredze, 2015)) rather than filtering out words in a

rule-based fashion.

We make our dataset, including all activity pairs and averaged human ratings, publicly available at http://lit.eecs.umich.edu/downloads.html.

Acknowledgments

This material is based in part upon work supported by the Michigan Institute for Data Science, by the National Science Foundation (grant #1344257), and by the John Templeton Foundation (grant #48503). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the Michigan Institute for Data Science, the National Science Foundation, or the John Templeton Foundation. We would also like to thank members of the University of Michigan LIT lab for help with data annotation.

References

Eneko Agirre, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, and Weiwei Guo. 2013. sem 2013 shared task: Semantic textual similarity, including a pilot on typed-similarity. In *SEM 2013: The Second Joint Conference on Lexical and Computational Semantics. Association for Computational Linguistics*. Citeseer.

Eneko Agirre, Mona Diab, Daniel Cer, and Aitor Gonzalez-Agirre. 2012. Semeval-2012 task 6: A pilot on semantic textual similarity. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, pages 385–393. Association for Computational Linguistics.

Icek Ajzen. 1987. Attitudes, traits, and actions: Dispositional prediction of behavior in personality and social psychology. *Advances in experimental social psychology*, 20:1–63.

Mohit Bansal, Kevin Gimpel, and Karen Livescu. 2014. Tailoring continuous word representations for dependency parsing. Association for Computational Linguistics.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155.

Kenneth Ward Church and Patrick Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational linguistics*, 16(1):22–29.

Manaal Faruqui and Chris Dyer. 2015. Non-distributional word vector representations. *CoRR*, abs/1506.05230.

Manaal Faruqui, Yulia Tsvetkov, Pushpendre Rastogi, and Chris Dyer. 2016. Problems with evaluation of word embeddings using word similarity tasks. *arXiv preprint arXiv:1605.02276*.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2001. Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on World Wide Web*, pages 406–414. ACM.

Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2013. Ppdb: The paraphrase database. In *HLT-NAACL*, pages 758–764.

Daniela Gerz, Ivan Vulić, Felix Hill, Roi Reichart, and Anna Korhonen. 2016. SimVerb-3500: A Large-Scale Evaluation Set of Verb Similarity. In *EMNLP*.

Jeremy Goecks and Jude Shavlik. 2000. Learning users' interests by unobtrusively observing their normal behavior. In *Proceedings of the 5th international conference on Intelligent user interfaces*, pages 129–132. ACM.

Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 855–864. ACM.

Felix Hill, Roi Reichart, and Anna Korhonen. 2016. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*.

Eric H Huang, Richard Socher, Christopher D Manning, and Andrew Y Ng. 2012. Improving word representations via global context and multiple word prototypes. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 873–882. Association for Computational Linguistics.

Kate Kearns. 1988. Light verbs in english.

Ryan Kiros, Yukun Zhu, Ruslan R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Skip-thought vectors. In *Advances in neural information processing systems*, pages 3294–3302.

Quoc V Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *ICML*, volume 14, pages 1188–1196.

Omer Levy and Yoav Goldberg. 2014a. Dependency-based word embeddings. In *ACL (2)*, pages 302–308. Citeseer.

Omer Levy and Yoav Goldberg. 2014b. Neural word embedding as implicit matrix factorization. In *Advances in neural information processing systems*, pages 2177–2185.

Marco Marelli, Stefano Menini, Marco Baroni, Luisa Bentivogli, Raffaella Bernardi, and Roberto Zamparelli. 2014. A sick cure for the evaluation of compositional distributional semantic models. In *LREC*, pages 216–223.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Jeff Mitchell and Mirella Lapata. 2010. Composition in distributional models of semantics. *Cognitive science*, 34(8):1388–1429.

Nikola Mrkšić, Diarmuid O Séaghdha, Blaise Thomson, Milica Gašić, Lina Rojas-Barahona, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. 2016. Counter-fitting word vectors to linguistic constraints. *arXiv preprint arXiv:1603.00892*.

Bennet B Murdock Jr. 1962. The serial position effect of free recall. *Journal of experimental psychology*, 64(5):482.

Judith A Ouellette and Wendy Wood. 1998. Habit and intention in everyday life: The multiple processes by which past behavior predicts future behavior. *Psychological bulletin*, 124(1):54.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–1543.

Nils Reimers, Philip Beyer, and Iryna Gurevych. 2016. Task-oriented intrinsic evaluation of semantic textual similarity. In *COLING*, pages 87–96.

Milton Rokeach. 1973. *The nature of human values.* Free press.

Rocco Tripodi and Marcello Pelillo. 2016. A game-theoretic approach to word sense disambiguation. *Computational Linguistics*.

John Wieting, Mohit Bansal, Kevin Gimpel, and Karen Livescu. 2015. Towards universal paraphrastic sentence embeddings. *arXiv preprint arXiv:1511.08198*.

John Wieting, Mohit Bansal, Kevin Gimpel, and Karen Livescu. 2016. Charagram: Embedding words and sentences via character n-grams. *arXiv preprint arXiv:1607.02789*.

Mo Yu and Mark Dredze. 2015. Learning composition models for phrase embeddings. *Transactions of the Association for Computational Linguistics*, 3:227–242.