Title

Automatically Inferring Implicit Properties in Similes

Permalink

https://escholarship.org/uc/item/1k59345h

ISBN

9781941643914

Authors

Qadir, Ashequl
Riloff, Ellen
Walker, Marilyn A

Publication Date

2016

DOI

10.18653/v1/n16-1146

Peer reviewed

Automatically Inferring Implicit Properties in Similes

Ashequl Qadir and **Ellen Riloff**
School of Computing
University of Utah
Salt Lake City, UT 84112, USA
{asheq,riloff}@cs.utah.edu

Marilyn A. Walker
Natural Language and Dialogue Systems Lab
University of California Santa Cruz
Santa Cruz, CA 95064, USA
mawalker@ucsc.edu

Abstract

A simile is a figure of speech comparing two fundamentally different things. Sometimes, a simile will explain the basis of a comparison by explicitly mentioning a shared property. For example, *"my room is as cold as Antarctica"* gives "cold" as the property shared by the room and Antarctica. But most similes do not give an explicit property (e.g., *"my room feels like Antarctica"*) leaving the reader to infer that the room is cold. We tackle the problem of automatically inferring implicit properties evoked by similes. Our approach involves three steps: (1) generating candidate properties from different sources, (2) evaluating properties based on the influence of multiple simile components, and (3) aggregated ranking of the properties. We also present an analysis showing that the difficulty of inferring an implicit property for a simile correlates with its *interpretive diversity*.

1 Introduction

A simile is a figure of speech comparing two essentially unlike things, typically using "like" or "as" (Paul, 1970). Comparing fundamentally different types of entities is what makes a simile figurative (Israel et al., 2004). Similes may be *closed* or *open* (Beardsley, 1981). A *closed* simile explains the basis for a comparison by explicitly mentioning a shared property. For example, the simile *"my room is as cold as Antarctica"* gives "cold" as the property shared by both the room and Antarctica. But most similes do not explicitly mention the basis for comparison, leaving people to infer what the entities have in common. An *open* simile expressing the same comparison is *"my room feels like Antarctica"*, where the shared property of being cold is left implicit. In our study of similes in tweets, we found that 92% of similes are open similes so the property must be inferred. Our research tackles this problem of inferring the implicit property evoked by an open simile.

Inferring the basis of comparison in a simile is central to natural language understanding and metaphor interpretation. For example, *"John was like a lion in battle"* is probably a statement about John's bravery or courage, not a description of John's physical appearance. Methods to understand figurative similes could also be valuable to understand metaphor in other linguistic constructions, such as predicate nominals (e.g., *"he is a lion"*). Furthermore, identifying the implicit property of a simile could be useful for sentiment analysis, because similes are often used to express positive and negative feelings (Li et al., 2012). For example, *"John was like a lion in battle"* contains only neutral words, but inferring "bravery" as the implicit property suggests that the simile has positive polarity.

We designed a three step process to infer the implicit properties of open similes. First, we generate candidate properties for a simile by harvesting words that are associated with its verb ("event") or object of comparison ("vehicle") using a variety of methods, including syntactic patterns, dictionary definitions, and word embeddings. Each candidate property is generated from just one component of the simile. The second step of the process then evaluates each property's compatibility with the complementary component of the simile (event or vehi-

Proceedings of NAACL-HLT 2016, pages 1223–1232,
San Diego, California, June 12-17, 2016. ©2016 Association for Computational Linguistics

cle). Finally, the third step of the process aggregates all of the candidates generated by different methods and ranks them based on collective evidence from the different sources. We evaluate the performance of our approach using gold standard properties provided by seven human annotators. We also present an analysis of the similes in our data set with respect to their *interpretive diversity* (intuitively, a measure of how many plausible interpretations a simile has). We show that our method performs best on similes with low diversity, as one would expect since their implicit properties are most clear to humans.

2 Problem Description and Data

A simile typically consists of four key components: the **topic** or **tenor** (subject of the comparison), the **vehicle** (object of the comparison), the **event** (act or state), and a **comparator** (usually “as”, “like”, or “than”) (Niculae and Danescu-Niculescu-Mizil, 2014). For the simile *“the room feels like Antarctica”*, “room” is the tenor, “feels” is the event, and “Antarctica” is the vehicle. A **property** (shared attribute) can optionally be included to explicitly state how the tenor is being compared with the vehicle, (e.g., *“the room is as* ***cold*** *as Antarctica”*).

Table 1 shows examples of open similes from our Twitter data set, along with several properties inferred by our human annotators (our data set will be described in Section 2.1). We represent each simile using just the head noun of the tenor and vehicle, and the lemma of the event. Veale and Hao (2007) observed that when a property is explicitly given, it is usually a salient property of the vehicle. Table 1 illustrates some examples of inferred properties that are strongly associated with the vehicle (e.g., “melodic” and “dulcet” are musical attributes).

We observed that implicit properties can be strongly evoked from the event as well. For example, most inferred properties for *“person buzz like fridge”* emanate from the word “buzz”, such as “humming”, “vibrating”, “distracting”, and “annoying”. Similarly, the tenor can also evoke properties, as we see with the inferred property “squinty” for the simile *“eye feel like clam”* although our observation is that this is less common. The event and the tenor need to be semantically rich to evoke implicit properties. The event in many similes is a form of “to be” or a perception verb (e.g., “feels”), which are semantically weak and contribute little. A tenor provides limited information when it is a pronoun or unknown entity (e.g., *“John drives like a snail”* is understandable without knowing who John is).

Simile	**Properties Inferred by Humans**
laugh be like music	melodic, pleasing, dulcet, tinkly
person sound like prophet	wise, insightful, prescient, enlightened
eye feel like clam	slimy, squinty, weary, gummy, heavy
person look like carrot	orange, thin, scrawny, slim, tall
person buzz like fridge	humming, vibrating, distracting, annoying, motorized
person fight like animal	ferociously, scratches, tenaciously
person be like shark	sneaky, primordial, dangerous, cold
time be like river	flowing, fast, winding, unending, moving
praise be like sunlight	warm, rejuvenating, energizing, cheerful

Table 1: Similes with sample properties inferred by humans.

Ultimately, an implicit property must be compatible with the vehicle, event, and the tenor in order for a simile to make sense. For example, Antarctica is strongly associated with the color “white”, but it would not make sense to infer the property “white” for the simile *“my room feels like Antarctica”* because of the verb “feel”. Although in this example the tenor “room” is still compatible with “white” and will not help to eliminate “white” as a property, in other similes it may (e.g., rivers can be “wide”, but time can not be, so “wide” can be eliminated as an implicit property in the simile “time be like river”).

A novel aspect of our work is that our architecture is designed to consider a property’s compatibility with multiple components. In this research, for generating candidate properties and utilizing their influence for compatibility, we particularly focus on the vehicle and event terms. Initially, we generate candidate properties from the vehicle and the event separately. But the second step then evaluates each candidate property’s compatibility with the *complementary* simile component. If a property was initially generated from the vehicle, then we evaluate its compatibility with the event; if a property was initially generated from the event, then we evaluate its compatibility with the vehicle. This approach emphasizes the need to consider multiple components of a simile when inferring implicit properties.

2.1 Collecting Similes with Implicit Properties

For our research, we created a new data set of open similes, where the property is implicit. Similes are common on Twitter, so we extracted similes from roughly 140 million English tweets collected during the time period 2/13/2013 – 4/15/2014. To identify similes, we applied a part-of-speech tagger designed for Twitter (Owoputi et al., 2013) to tweets containing the word "like" and applied rules to recognize simple noun phrases and verb phrases. We then selected tweets matching the syntactic pattern: $NP_1\ VERB\ like\ NP_2$, where NP_2 can contain only a noun and an optional indefinite article. We required similes to have a vehicle term with no premodifiers to avoid problems associated with coreference (e.g., "the man" or "that man") and to focus on vehicles that represent general concepts. We leave for future work the challenge of tackling multi-word vehicle phrases (e.g., *"my room is like stepping into a hurricane"* or *"my room is like a boots store"*).

This selection process extracted many similes, but it also extracted literal comparisons with no apparent property (e.g., *"this flower smells like a rose"*) and statements that are not comparisons (e.g., *"I called like five times"*). To focus on figurative similes with an implicit property, we further filtered the collection to only retain similes with vehicle terms that had occurred in comparisons with an explicit property. Using the same Twitter data, we extracted nouns that appeared in the following syntactic patterns, which represent comparison constructions with an adjectival property: $ADJ\ like\ [a, an]\ NOUN$ (e.g., *"red like a tomato"*) and $ADJ\ as\ [a, an]\ NOUN$ (e.g., *"red as a tomato"*). We only kept similes whose vehicle occurred in these patterns. Finally, we filtered similes that contain a pronoun (except personal pronouns in the tenor, which we generalized to a *"person"* token), common person first names[1], profanity,[2] or words not in a dictionary[3] to avoid issues with Twitter language such as misspellings, elongated words, etc.

2.2 Gold Standard Implicit Properties

We developed a gold standard set of implicit properties for each simile using Mechanical Turk. We prequalified 7 workers, who each annotated 700 similes with frequency ≥ 3 randomly selected from our collection. Each annotator was asked to provide up to 2 properties that best captured the most likely basis for comparison between the tenor and vehicle. We also provided the annotators with the option to label a simile as Invalid if it was not a simile at all (most commonly due to parse errors, such as *"he looks like ran"*) or label a simile as having No Property (often due to literal or underspecified comparisons, such as *"she looks like my aunt"*). The annotators were asked to give adjectives, adverbs, or verbs but occasionally they provided a noun. Table 1 presents sample annotated simile properties.

Among the 700 similes, a majority of the annotators labeled 59 of them as either Invalid or No Property, so we did not use these. We set aside 183 similes (29%) as a development set and the remaining 458 similes (71%) as a test set.

3 Inferring Implicit Properties

Our research tackles the problem of inferring properties in open similes by decomposing the problem into three subtasks: (1) generating candidate properties, (2) evaluating the candidate properties with respect to multiple simile components, and (3) aggregated ranking of the properties. Figure 1 illustrates our approach.

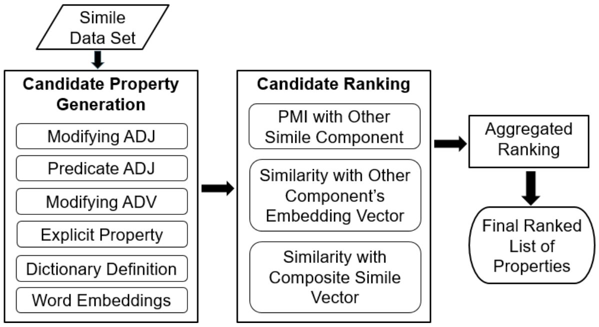

Simile Data Set
Candidate Property Generation
Modifying ADJ
Predicate ADJ
Modifying ADV
Explicit Property
Dictionary Definition
Word Embeddings
Candidate Ranking
PMI with Other Simile Component
Similarity with Other Component's Embedding Vector
Similarity with Composite Simile Vector
Aggregated Ranking
Final Ranked List of Properties

Figure 1: Framework for inferring implicit properties.

First, the vehicle and event components of a simile are used individually to generate candidate properties. We investigate a variety of candidate generation methods, including harvesting properties

[1] http://deron.meranda.us/data/census-derived-all-first.txt

[2] http://www.bannedwordlist.com/lists/swearWords.txt

[3] Using Wordnik: https://www.wordnik.com/

from syntactic structures and dictionary definitions, identifying relevant properties using statistical co-occurrence, and assessing similarity between word embedding vectors.

Second, the candidates generated by each method are evaluated based on their strength of association with the complementary component of the simile. For candidates generated from the vehicle term, we evaluate them based on their association with the event term, and vice versa. We explore three association measures: point-wise mutual information to measure statistical co-occurrence, and vector similarity using single and composite word embeddings.

Third, we produce an aggregate ranking over the entire set of properties hypothesized by *all* of the candidate generation methods. Intuitively, we view each candidate generation method as an independent source, and look at the aggregate evidence across the set of different candidate generation methods (similar to an ensemble). Each property is scored based on its average rank across the different methods, so that properties highly ranked by multiple methods are preferred.

3.1 Candidate Property Generation

We generate candidate properties from the vehicle and event words of a simile. However when the event is a form of "to be" or a perception verb (taste, smell, feel, sound, look), we do not generate candidate properties from the event because the verb is too general. Only 73 (16%) of the similes in our evaluation data have a verb other than "to be" or a perception verb. We restrict properties to be adjectives, adverbs, or verb forms that can function as nominal premodifiers (e.g., "crying baby", "wilted lettuce"). We explore a total of seven methods for generating candidate properties and generate candidates using our entire Twitter corpus.

Modifying ADJ: Given a vehicle term, we extract pre-modifying adjectives. For example, "ripe" is extracted for the vehicle "tomato" from the phrase "ripe tomato".

Predicate ADJ: Given a vehicle term, we extract adjectives in predicate adjective constructions with the vehicle. For example, "red" is extracted for the vehicle "tomato" from the phrase "tomato is red".

Modifying ADV: Given an event term (verb), we extract adverbs that precede or follow the verb. For example, "immaturely" is extracted for the event "act" due to the phrase "acts immaturely".

Explicit Property: We extract properties mentioned explicitly in comparison phrases. For vehicle terms, we extract properties from phrases of the form: "ADJ/ADV like NP" (e.g., *"cold like Antarctica"*) and "ADJ/ADV as NP" (e.g., *"cold as Antarctica"*). For event terms, we extract properties from phrases of the form: "VERB ADJ/ADV like" and "VERB as ADJ/ADV as" (e.g., *"feels as cold as"*).

Dictionary Definition: Dictionary definitions often mention salient properties associated with a word. We harvest adjectives, adverbs and verbs (functioning as premodifiers) as candidate properties from the dictionary definitions of the vehicle and event terms. For the definitions, we use Wordnik[4], which contains 5 source dictionaries: Heritage Dictionary of the English Language, Wiktionary, the Collaborative International Dictionary of English, The Century Dictionary and Cyclopedia, and WordNet 3.0 (Miller, 1995).

PMI: Given a vehicle or event term, we compute point-wise mutual information (PMI) between that term and candidate properties (appearing in $\geq$ 100 tweets) in our Twitter corpus.

Word Embedding: We train a word embedding model using our tweet collection, limiting the vocabulary to nouns, verbs, adjectives and adverbs that occurred in $\geq$ 100 tweets. For training, we use word2vecf[5] (Levy and Goldberg, 2014) which allows training for arbitrary context using the skip-gram model. We use 300 dimensions for the output word and context vectors. Candidate properties are generated by selecting the words whose context vector[6] is most similar to the vehicle or event's word vector using cosine similarity. To control for noisy candidates, we require that the property occurred with the vehicle (or event) as a bigram with frequency $\geq$ 10 in the Twitter corpus.

For each generation method, we rank the candidates and select the top 20 properties. For the four methods that use syntactic patterns, we calculate

[4] https://www.wordnik.com/

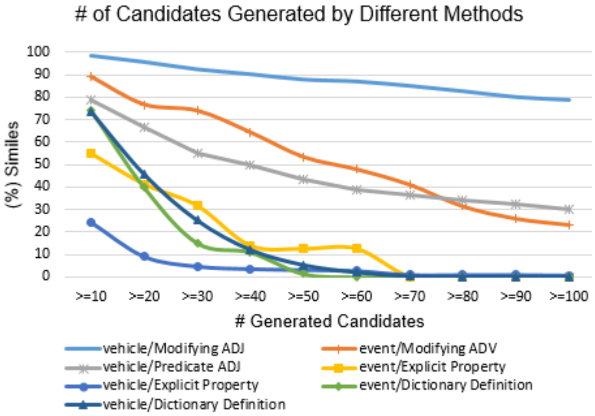
[5] https://bitbucket.org/yoavgo/word2vecf

[6] properties are expected in the context of a component word.

P(property | vehicle) based on the number of times the property and the vehicle appear together in that syntactic construction among all times the vehicle appear in that syntactic construction. We use this probability to rank the candidates. For the dictionary definition method, we sort the properties based on how many of the 5 dictionaries mention the property in the word's definition. We break ties based on the frequency of the property in the definitions. For the word embedding-based method, we use cosine similarity scores.

3.2 Productivity of the Candidate Generation Methods

First we investigate how many candidates each method is able to generate. If a method generates too few candidates, it will not be very useful. Conversely, if a method generates a large number of candidates, then our ranking framework needs to be robust to rank the plausible properties higher than the properties that do not fit.



	Average	Min	Max
# of Candidates Generated from Vehicle			
Modifying ADJ	423.62	1	3177
Predicate ADJ	104.21	0	1070
Explicit Property	8.28	0	116
Dictionary Def.	20.5	0	71
# of Candidates Generated from Event			
Modifying ADV	68.67	2	223
Explicit Property	19.85	0	61
Dictionary Def.*	18.59	3	55

Figure 2: Statistics about candidates generated by different methods. Similes with a "to be" or perception verb were excluded for the methods that use the event as the source.

Figure 2 presents statistics about the candidate properties generated by different methods. The PMI and Word Embedding-based methods were excluded here as these methods evaluate all words in the corpus. The methods that used the explicit property extraction patterns and dictionary definitions generate fewer candidates than the methods that used general syntactic structures. The trend lines in Figure 2 show that these methods do not generate more than 20 candidate properties for most similes.

3.3 Coverage of the Generated Candidates

Next, we investigate the effectiveness of our candidate generation methods. The last column of Table 2 shows candidate ranking results based on Mean Reciprocal Rank (MRR) for the top 20 properties produced by each candidate generation method. MRR is calculated by:

$$MRR = \frac{1}{|S|}\sum_{s\in S}\frac{1}{(rank\ of\ 1^{st}\ acceptable\ property)}$$

where S is the set of similes. We observe that the *PMI* method (for both vehicles and events) and the *Dictionary Definition* method (for events) produced low MRR scores < 0.10. Therefore we decided not to use these candidate generation methods.[7]

One of our primary concerns is assessing the ability of our candidate generation methods to generate at least some acceptable properties. We expect them to over-generate, but they need to produce at least one acceptable property or the downstream components will be helpless. To assess this, we evaluated the coverage of each candidate generation method based on the Top 10, Top 20, and Top 30 properties that it produced. Coverage is the percentage of similes for which the method generates at least one gold standard property (from the human annotators). Table 2 shows that the *Dictionary Definitions* for vehicles was the best performing method for the Top 10 candidates, generating at least one acceptable property for 40% of the similes. The *Modifying ADJ* method performed best for the Top 30 candidates, generating an acceptable property for 63% of similes. Note that the *Explicit Property* method performs reasonably well (40% coverage for Top 30 properties generated from vehicles and 6% coverage for properties generated from events), but clearly is

[7] We made this decision based on similar results observed on our development data.

not sufficient on its own, showing the limitation of harvesting explicitly stated properties.

	Top10	Top20	Top30	MRR
Coverage of Candidates Generated from Vehicle				
PMI*	18%	31%	37%	.06
Modifying ADJ	39%	55%	63%	.16
Predicate ADJ	28%	39%	43%	.11
Explicit Property	37%	39%	40%	.23
Dictionary Def.	40%	47%	49%	.22
Word Embedding	35%	48%	58%	.15
ALL	76%	84%	86%	n/a
Coverage of Candidates Generated from Event				
PMI*	2%	3%	4%	.09
Modifying ADV	4%	5%	5%	.13
Explicit Property	4%	5%	6%	.16
Dictionary Def.*	3%	4%	4%	.09
Word Embedding	5%	6%	6%	.16
ALL	9%	10%	10%	n/a
All Candidates				
TOTAL	78%	86%	88%	n/a

Table 2: Coverage and MRR for the candidate generation methods. Top10, Top20, Top30 = percent of similes with a plausible property within top 10, 20, 30 ranked properties. Methods excluded in "ALL" and "TOTAL" rows are marked with (*). In the MRR calculation when the event component is source, similes with a "to be" or a perception verb were excluded.

The ALL rows show the coverage obtained by combining the property lists from all generation methods listed above in the table. The combined set of properties (Top 30) generated from vehicles yields 86% coverage, while the combined set of properties generated from events yields only 10% coverage (partly because these methods apply to only 16% of the similes), showing that vehicles are more effective for candidate generation. However, the TOTAL row shows that combining properties generated from both vehicles and events yields 88% coverage using the Top 30 candidates. The Top 20 candidates provide coverage that is nearly as good (86%) with substantially fewer properties to process downstream, so we use the Top 20 candidates for all of our experiments.[8]

3.4 Ranking the Candidate Properties Using Influence from the Second Component

Next, we investigate whether the initial ranking results in the previous step can be improved by considering the second component of the simile. Intuitively, suppose that "green", "slow", and "endangered" are generated as candidate properties from the vehicle "turtle" (e.g., for *"dad drives like a turtle"*). Taking the event verb "drive" into account can help to rank "slow" more highly than the other candidates. We explore three criteria to rank candidates generated from one simile component based on its association with the second component (unless the event is "to be" in which case we retain the original candidate ranking because the verb is too general).

PMI with second component (PMI): We calculate Pointwise Mutual Information between a candidate property and the second component of a simile.

Embedding word vector similarity with the second component (EMB$_1$): We use our trained word embeddings model to calculate cosine similarity between a candidate property and the second component of the simile. As before, for properties we use the context vectors.

Embedding word vector similarity with composite simile vector (EMB$_2$): For a given event and vehicle, we create a composite simile vector by performing element-wise addition of the vectors for the event and the vehicle, and calculate cosine similarity with the candidate properties. For example, for *"person talks like robot"*, the vectors for "talk" and "robot" are used to create a composite vector, and the similarity of the resulting vector with a candidate property's context vector is used as the ranking criteria. The intuition here is to capture what is common in the context distribution (Mikolov et al., 2013) of "robot" and "talk", and the context vector of a suitable property should have strong similarity with the resulting vector.

3.5 Results for Candidate Re-ranking

Table 3 presents MRR results after the initially generated candidates are re-ranked using the influence of the second simile component. For comparison, the MRR results from Table 2 are also presented in the first column (**Orig**).

Influence from the second simile component assessed with PMI and EMB$_1$ improved the MRR scores for some candidate generation methods (e.g., Predicate ADJ), but did not for others (e.g., Modifying ADV). However using the composite word embedding vector (EMB$_2$) to capture the common

[8] The decision to use the Top 20 candidates was based on similar results on our development data.

Ranking Method	Orig	PMI	EMB$_1$	EMB$_2$
Candidates Generated from Vehicle				
Modifying ADJ	.16	.22	.19	**.24**
Predicate ADJ	.11	.16	.14	**.22**
Explicit Property	.23	.25	.23	**.28**
Dictionary Def.	.22	.21	.20	**.25**
Word Embedding	.15	.19	.20	**.21**
Candidates Generated from Event				
Modifying ADV	.13	.10	.13	**.19**
Explicit Property	.16	**.18**	**.18**	**.18**
Word Embedding	.16	.11	.14	**.18**

Table 3: MRR scores for candidate ranking methods.

aspects in the context distributions of the event and vehicle consistently improved MRR for all candidate generation methods. Consequently, we use the composite word embedding vector as the ranking method for each set of candidate properties.

3.6 Aggregated Ranking

Finally, we need to consider all of the properties produced by the various candidate generation methods. As we saw in Table 2, they produce complementary sets of properties and coverage is highest when we use all of them together. To produce an aggregated ranking of all candidate properties, we calculate the harmonic mean of the rank for each individual candidate generation method. This approach rewards properties that have a consistently high ranking across different methods.

For comparison, we also show results for a voting method where a candidate property is ranked based on how many different methods generated it. To break ties, we used the frequency of the candidate in our Twitter corpus.

3.7 Results for Aggregated Ranking

Our final results use two gold standard property sets: (1) Gd (Gold): uses the set of properties from the human annotators, and (2) Gd+WN expands Gold with WordNet synsets (words in the same synset of a gold property are added) and WordNet's "similar to" relation (words that are connected to a gold property by the relation are added). The reason for using *Gd+WN* is to include synonyms of a gold property that would otherwise be considered wrong (e.g., if a human annotator said "beautiful" and our system said "pretty").

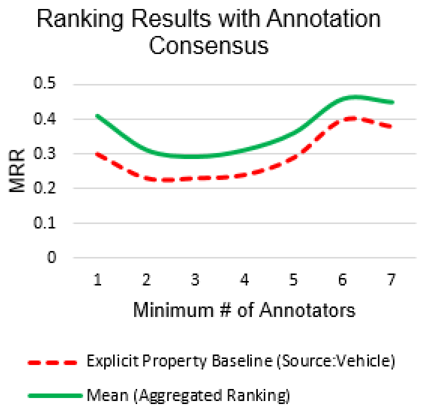
The first two columns in Table 4 present MRR results for our final ranking. The results show that with both *Gd* and *Gd+WN*, our aggregated ranking using harmonic mean yields much better MRR results than the individual methods and better than the Voted method, yielding our highest MRR: .33 and .41.

	MRR		**Top 1**		**Top 5**	
	Gd	Gd + WN	Gd	Gd + WN	Gd	Gd + WN
Voted	.25	.35	14%	21%	36%	52%
Mean	**.33**	**.41**	**21%**	**27%**	**46%**	**58%**

Table 4: Aggregate ranking results.

The last 4 columns of Table 4 present the percentage of similes for which an acceptable property was ranked #1 (Top 1) or within the Top 5. Our aggregate ranking scheme ranks an acceptable property in the Top 1 position for 27% of similes based on *Gd+WN*, and inferred an acceptable property within the Top 5 positions for 58% of all similes.

For the above evaluations, any property given by the annotators is deemed correct, and any consensus that the annotators may have had is not accounted for. To address this, we retained properties with different degrees of consensus, and subdivided the evaluation data set. Each subset of the data kept similes that have properties from a minimum number of annotators, and only those properties are used as the gold standard. WordNet synsets and "similar to" relations are also used in determining consensus.



Min # of Annotators	1	2	3	4	5	6	7
# of Similes in Data Set	641	588	418	252	136	67	27

Figure 3: Ranking results tracked by annotation consensus with Gd+WN gold standard, and corresponding data set sizes.

Figure 3 shows that for all degrees of consensus, the aggregated ranking is consistently better than the method that uses the explicit property extraction patterns, which was the best individual candidate generation method. When properties given by at least 2 annotators are considered as the gold standard, MRR is lower than when properties given by any annotator are used. With higher consensus, MRR gradually increases, which is probably because the properties with high consensus have stronger association with the simile components, so are easier to infer.

4 Analysis and Discussion

Our gold standard property collection confirmed our intuition that some similes have many plausible interpretations while others do not. We hypothesized that this should contribute to the difficulty of implicit property inference. Utsumi and Kuwabara (2005) introduced "interpretive diversity" with the hypothesis that similes with more diversity in the inferred property tend to be more metaphorical, and the values of salience of the properties are more uniform. They used Shannon's entropy to measure the interpretive diversity of a simile.

To explore our hypothesis regarding difficulties associated with property inference, we first cluster our gold-standard annotated properties. When a property appears in the WordNet synset of another property, or if two properties are connected by the WordNet "similar to" relation, we group the properties to form property clusters. So each property cluster represents a set of words that are synonyms of each other. We aggregate frequency statistics of individual words in a cluster and measure interpretive diversity of a simile using Shannon's entropy (here, X is the random variable representing property clusters of a simile):

$$H(X) = -\sum_{x \in X} P(x) \log_2 P(x)$$

Figure 4 shows the entropy curve after the 641 similes are sorted by the entropy values of their property clusters. Based on changes in the slope of the curve, we then divided the data into 3 subsets, similes with high (1–100 similes), medium (101–500 similes), and low (501–641 similes) interpretive diversity. Table 5 presents examples of similes in each category. High interpretive diversity is clearly demonstrated by *"person act like mom"*, showing properties with many different characteristics attributed to mom. Note that the properties contain both positive (e.g., friendly, loving) and negative (scolding, annoying) attributes. On the other side of the spectrum are similes with low interpretive diversity, as exemplified by *"throat feel like sandpaper"* where the vocabulary of the property set is more limited.

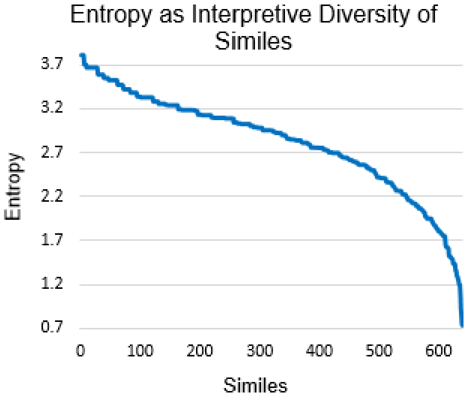


Figure 4: Entropy as interpretive diversity of similes.

High Interpretive Diversity
person act like mom : bossy (2), friendly, nuturing, overbearing, loving, scolding, caring, hovers, strict protective, cleans, nurturing, annoying
person act like baby : {childish,immature,young} (4), crying (2), whine, silly, cry, dependent, needy, pouting, whiny, weak
Medium Interpretive Diversity
person look like robot : stiff (5), jointed, stoic, blank, expressionless, mechanical, inhuman, dull, uneasy
girl be like butterfly : {beautiful,pretty} (4), free (2), delicate (2), graceful (2), fluttering, floating, happy, flowy
Low Interpretive Diversity
person act like clown : {goofy,ridiculous,silly} (5), {amusing,comical,funny} (5), stupid, degrading, disruptive, childish
throat feel like sandpaper : {rough,scratchy} (9), coarse (2), raspy, sore, dry

Table 5: Similes with different levels of interpretive diversity. Aggregated frequencies are presented within parenthesis.

Table 6 shows that it is much harder to infer the implicit property in similes with high interpretive diversity, demonstrated by a .19 difference in MRR score from high to low. This trend is also consistent when we see the percentage of similes for which the system ranks a plausible property at the topmost

Diversity	High	Medium	Low
MRR	.31	.40	.50
Top 1	15%	26%	37%
Top 5	47%	57%	66%

Table 6: Results for different subsets of similes divided by interpretive diversity, using Gold+WN properties.

position (Top 1) or within the Top 5. It is possible that with low interpretive diversity, when the property distribution is unimodal or bimodal, statistical associations between a property and simile components are stronger, and so more easily discovered by our candidate generation and ranking methods.

5 Related Work

Similes have been studied in linguistics and psycholinguistics to understand how humans process similes, comparisons, and metaphors, and the interplay among different components of these linguistic forms. Glucksberg et al. (1997) presented a property attribution model of metaphor comprehension where the candidate properties are selected from a vehicle and applied to a topic. Chiappe and Kennedy (2000) investigated if the number of properties varies between a metaphor and its simile form. The impacts of semantic dimensions of tenor and property salience have been compared by Gagné (2002). Fishelov (2007) experimented with affective connotation and degrees of difficulty associated with understanding a simile when a simile property is conventional or unconventional, or no property is given. Hanks (2005) manually categorized vehicle nouns of similes into semantic categories.

Automatic approaches that use computational models for similes are relatively rare. Veale and Hao (2007) extracted salient properties of vehicles from the web using "as ADJ as a/an NOUN" extraction pattern to acquire knowledge for concept categories. Veale (2012) built a knowledge-base of affective stereotypes by characterizing simile vehicles with salient properties. Li et al. (2012) used explicit property extraction patterns to determine the sentiment that properties convey toward simile vehicles. Niculae and Yaneva (2013) and Niculae (2013) used constituency and dependency parsing-based techniques to identify similes in text. Qadir et al. (2015) classified similes into positive and negative affective polarities using supervised classification, with features derived from simile components. Niculae and Danescu-Niculescu-Mizil (2014) designed a classifier with domain specific, domain agnostic, and metaphor inspired features to determine when comparisons are figurative.

Computational approaches to work on figurative language also include figurative language identification using word sense disambiguation (Rentoumi et al., 2009), harvesting metaphors by using noun and verb clustering-based techniques (Shutova et al., 2010), interpreting metaphors by generating literal paraphrases (Shutova, 2010), etc.

Although previous research has extensively used explicit property extraction patterns for various tasks, none has explored the impact of multiple simile components for inferring properties. To our knowledge, we are the first to introduce the task of automatically inferring the implicit properties in open similes, which is fundamental to automatic understanding of similes.

6 Conclusion

In this work, we addressed the problem of inferring implicit properties in open similes. We showed that acceptable properties for most similes can be identified by harvesting properties using syntactic structures, dictionary definitions, statistical co-occurrence, and word embedding vectors. We then demonstrated that capturing the combined influence of a simile's event and vehicle terms using a composite word embedding vector improved our ability to rank candidate properties. Finally, we showed that properties harvested by different methods can be aggregated and effectively ranked using the harmonic mean of rankings from the individual methods. Our method for inferring implicit properties performed best on similes with low interpretive diversity. In future work, we plan to use the inferred properties to improve affective polarity recognition in similes.

Acknowledgments

This material is based upon work supported in part by the National Science Foundation under grants IIS-1450527 and IIS-1302668.

References

Monroe C Beardsley. 1981. *Aesthetics, problems in the philosophy of criticism*. Hackett Publishing.

Dan L Chiappe and John M Kennedy. 2000. Are metaphors elliptical similes? *Journal of Psycholinguistic Research*, 29(4):371–398.

David Fishelov. 2007. Shall i compare thee? simile understanding and semantic categories. *Journal of literary semantics*, 36(1):71–87.

Christina L Gagné. 2002. Metaphoric interpretations of comparison-based combinations. *Metaphor and Symbol*, 17(3):161–178.

Sam Glucksberg, Matthew S McGlone, and Deanna Manfredi. 1997. Property attribution in metaphor comprehension. *Journal of memory and language*, 36(1):50–67.

Patrick Hanks. 2005. Similes and sets: The english preposition like. *Languages and Linguistics: Festschrift for Fr. Cermak. Charles University, Prague*.

Michael Israel, Jennifer Riddle Harding, and Vera Tobin. 2004. On simile. *Language, culture, and mind*, 100.

Omer Levy and Yoav Goldberg. 2014. Dependency-based word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, volume 2, pages 302–308.

Bin Li, Haibo Kuang, Yingjie Zhang, Jiajun Chen, and Xuri Tang. 2012. Using similes to extract basic sentiments across languages. In *Web Information Systems and Mining*, pages 536–542. Springer.

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.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Vlad Niculae and Cristian Danescu-Niculescu-Mizil. 2014. Brighter than gold: Figurative language in user generated comparisons. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2008–2018. Association for Computational Linguistics.

Vlad Niculae and Victoria Yaneva. 2013. Computational considerations of comparisons and similes. In *ACL (Student Research Workshop)*, pages 89–95.

Vlad Niculae. 2013. Comparison pattern matching and creative simile recognition. In *Proceedings of the Joint Symposium on Semantic Processing*.

Olutobi Owoputi, Brendan O'Connor, Chris Dyer, Kevin Gimpel, Nathan Schneider, and Noah A. Smith. 2013. Improved part-of-speech tagging for online conversational text with word clusters. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 380–390, Atlanta, Georgia, June. Association for Computational Linguistics.

Anthony M Paul. 1970. Figurative language. *Philosophy & Rhetoric*, pages 225–248.

Ashequl Qadir, Ellen Riloff, and Marilyn Walker. 2015. Learning to recognize affective polarity in similes. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 190–200, Lisbon, Portugal, September. Association for Computational Linguistics.

Vassiliki Rentoumi, George Giannakopoulos, Vangelis Karkaletsis, and A. George Vouros. 2009. Sentiment analysis of figurative language using a word sense disambiguation approach. In *Proceedings of the International Conference RANLP-2009*, pages 370–375. Association for Computational Linguistics.

Ekaterina Shutova, Lin Sun, and Anna Korhonen. 2010. Metaphor identification using verb and noun clustering. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 1002–1010. Association for Computational Linguistics.

Ekaterina Shutova. 2010. Automatic metaphor interpretation as a paraphrasing task. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 1029–1037. Association for Computational Linguistics.

Akira Utsumi and Yuu Kuwabara. 2005. Interpretive diversity as a source of metaphor-simile distinction. In *Proceedings of the 27th Annual Meeting of the Cognitive Science Society*, pages 2230–2235.

Tony Veale and Yanfen Hao. 2007. Learning to understand figurative language: from similes to metaphors to irony. In *Proceedings of CogSci*.

Tony Veale. 2012. A context-sensitive, multi-faceted model of lexico-conceptual affect. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2*, pages 75–79. Association for Computational Linguistics.