Phrasal Substitution of Idiomatic Expressions

Changsheng Liu and **Rebecca Hwa**
Computer Science Department
University of Pittsburgh
Pittsburgh, PA 15260, USA
{changsheng,hwa}@cs.pitt.edu

Abstract

Idioms pose a great challenge to natural language understanding. A system that can automatically paraphrase idioms in context has applications in many NLP tasks. This paper proposes a phrasal substitution method to replace idioms with their figurative meanings in literal English. Our approach identifies relevant replacement phrases from an idiom's dictionary definition and performs appropriate grammatical and referential transformations to ensure that the idiom substitution fits seamlessly into the original context. The proposed method has been evaluated both by automatic metrics and human judgments. Results suggest that high quality paraphrases of idiomatic expressions can be achieved.

1 Introduction

An idiom is a combination of words that has a figurative meaning which differs from its literal meaning. Idioms pose a great challenge to many NLP tasks, such as machine translation, word sense disambiguation, and sentiment analysis (Volk, 1998; Korkontzelos et al., 2013; Zanzotto et al., 2010; Williams et al., 2015). Previous work (Salton et al., 2014) has shown that a typical statistical machine translation system might achieve only half of the BLEU score (Papineni et al., 2002) on sentences that contain idiomatic expressions than on those that do not. Idioms are also problematic for second language learners. In a pilot study we have surveyed seven non-native speakers on 100 Tweets containing idioms; we have found that, on average, the participants had trouble understanding 70% of them due to the inclusion of idioms.

This work explores the possibility of automatically replacing idiomatic expressions in sentences. The full pipeline of a successful system has to solve many problems. First, it has to determine that an expression is, in fact, being used as an idiom in a sentence (Fazly et al., 2009; Korkontzelos et al., 2013; Sporleder and Li, 2009). Moreover, the system has to *sense disambiguate* the idiom – it has to pick the correct interpretation when more than one is possible. Second, it has to generate an appropriate phrasal replacement for the idiom using literal English. Third, it has to ensure that the replacement phrase will fit seamlessly back into the original sentence. This paper focuses on the second and third problem, which have not been studied as extensively in previous works.

We propose to extract the phrasal replacement for an idiom from its definition, assuming the existence of an up-to-date dictionary of broad coverage and high quality.[1] Because a typical definition is quite long, it cannot directly serve as a replacement for the idiom. A major challenge of our work is in identifying the right nugget to extract from the definition. Another major challenge is the smooth integration of the substitution phrase into the sentence. We consider both grammatical fluency as well as references resolution in our automatic Post-Editing technique. These phrasal challenges set our goals apart from related work on lexical simplification and substitution (Specia et al., 2012; Jauhar and Specia, 2012; McCarthy, 2002; McCarthy and Navigli, 2007) and

[1] This work uses TheFreeDictionary.com.

Proceedings of NAACL-HLT 2016, pages 363–373,
San Diego, California, June 12-17, 2016.

from general sentence simplification (Wubben et al., 2012; Siddharthan, 2014; Zhu et al., 2010; Coster and Kauchak, 2011) methods.

We validate the plausibility of the proposed methods with empirical experiments on a manually annotated corpus.[2] Results from both automatic evaluations and user studies show that the proposed approach can generate high-quality paraphrases of sentences containing idiomatic expressions. A successful idiom paraphrase generator may not only benefit non-native speakers, but may also facilitate other NLP applications.

2 Background

The main idea of this work is to produce a fluent and meaningful paraphrase of the original sentence similar to how a human non-native reader might approach the problem. Suppose the reader encounters the following sentence:

> **Sentence:** This kind of language really barfs me out and *gets my blood up*.[3]

If they do not understand the expression *gets my blood up*, they may look it up in a dictionary:

> **Definition:** Fig. to get someone or oneself angry. (Fixed order.)[4]

Then they might try to reconcile the definition with the context of the sentence and arrive at:

> **Paraphrased :** This kind of language really barfs me out and *gets me angry*.

In the example above, only a portion of the full definition is needed. One possible way to identify this relevant nugget is to apply sentence compression techniques (McDonald, 2006; Siddharthan, 2011; Štajner et al., 2013; Filippova et al., 2015; Filippova and Strube, 2008; Narayan and Gardent, 2014; Cohn and Lapata, 2009). However, all these methods have been developed for standard texts with complete sentences, and it is not clear whether they are suited to dictionary definitions. Consider Table 1, in which a corpus of 1000 randomly selected idiom definitions is compared with samples from two normal text corpora (CLwritten and CLspoken) used by Clarke and Lapata (2008). The CLwritten corpus comes from written sources in the British National Corpus and the American News Text corpus; the CLspoken corpus comes from transcribed broadcast news stories. We see that on average, definitions are shorter than complete sentences; arguably, each word in a definition carries more information. The density of punctuation per sentence shows that definitions are more fragmented. These factors are problematic for sentence compression techniques that rely heavily on the syntactic parse trees of complete, well-formed sentences (Cohn and Lapata, 2009; Narayan and Gardent, 2014). One recent compression method that does not rely as heavily on syntax is the work of Filippova et al. (2015). However, their approach requires a training set of considerable size, which is not practical for the domain of idiom definitions. The most likely to succeed text compression method for our domain is the work of McDonald (2006) as they only use syntactic information as soft evidence to compress target sentences. We choose this method as a comparative baseline in our experimental evaluation.

Corpus	Average length	Punctuation density
CLspoken	17	2.16
CLwritten	18	2.07
Definition	12	2.86

Table 1: Some statistics over normal text corpora and an idiom definition corpus.

After obtaining an appropriately shortened definition, *get someone angry*, more operations are needed to properly replace the idiom with it in the original sentence. First, we need to convert *get* to *gets* to make the tense consistent. Second, we need to resolve the reference *someone* to the appropriate person in the context of the original sentence: *me*. These operations are important to fit the shortened definition seamlessly into the original context, which will be covered in the Post Editing section.

3 Our Method

As outlined in the previous section, our proposed method consists of two components: *substitution*

[2] https://github.com/liucs1986/idiom_corpus

[3] https://twitter.com/ezzwanaezwnd/status/231992426548559872

[4] http://idioms.thefreedictionary.com/get+blood+up

generation and *post editing*.

3.1 Substitution Generation

This component aims to extract relevant replacement phrases from an idiom's dictionary definition. Rather than using generic sentence compression techniques, we argue that the taxonomy of a definition follows certain conventions that can be exploited. In most definitions, the *core meaning* is presented first; it is then optionally followed by additional information that supports, explains, and/or exemplifies the main point. The relationship between the core meaning and different types of addition information is akin to relationships between nucleus and surrounding sentences as described by the rhetorical structure theory (Mann and Thompson, 1988). Using a development set of idiom definition, we have identified four types of additional information:

Type	Example
Coordination	to discover **or apprehend** someone with something
Reason	to be feeling happy **because you are satisfied with your life**
Supplement	time is very important.**(Used especially when time is limited)**
Example	to apply thick soapsuds to something, **such as part of the body**

Table 2: Different types of additional information.

Below, we present two methods for extracting the core meaning from a full definition. We first consider a rule-based approach, under the assumption that definitions can be fully described by a small set of regular patterns. We also present a supervised machine-learning approach, showing that these regular patterns do not have to be predefined, thus opening up for possibilities of adapting the method to different dictionaries and languages.

3.1.1 A Rule-based Method

Analyzing the development set, we observe that additional information are often signaled by a small sets of lexical cues[5]; we call them *boundary words*. Using these boundary words and some shallow syntactic features[6], we have hand-crafted a small set of rules to pare down the definition. Below are five main types of rules:

1. **Delete coordinated phrase after the word "or" or "and"**. We consider that phrase to be an equivalent alternative and discard it.
2. **Delete subordinate clause after "because" or "since"**. The subordinate clause is used to elaborate the reason for a fact or event.
3. **Delete the clause after words such as "so that", "when", "if" and "especially"**. These are often extraneous supplemental information.
4. **Delete sentence after words such as "for instance", "e.g.", etc.**. The clause following these words usually gives further examples.
5. **Delete sentences in bracket**. This is often just supplemental information.

If multiple rules are applicable, we start from the rule that covers the widest range first, then to rules covering smaller ranges. After all these steps, if the output has more than one sentence, we always keep the first sentence for simplicity.

There is also a case of keyword ambiguity with respect to the word "as:" it could signal an explanation (like "because") or an example (like "such as"). Because TheFree Dictionary rarely use "as" by itself to signal an explanation, we have only encoded the "such as" sense in our rule-set to avoid the ambiguity.

3.1.2 A ML-based Method

Not every definition follows the schema expected by the rule-based system. To generalize the patterns, we cast substitution generation as a binary classification problem. The most straightforward way is to decide whether each word in a definition should be deleted or kept, but this will degrade the sequential fluency of the shortened definition.

A better alternative is to segment the definition into syntactic chunks such as non-embedded NP, VP, ADJP, ADVP and PP phrases using off-the-shelf

[5]We have identified 23 words and punctuation marks: and, or, because, since, 'cause, especially, if, for example, for instance, such as, e.g., i.e., etc., in particular, like, particularly, namely, viz. , specifically, so that, when, (,).

[6]These are obtained by using the shallow parser in Natural Language Tool Kit (NLTK) (Bird, 2006) and the parts-of-speech tagger in Stanford Parser (De Marneffe et al., 2006).

shallow parsers (e.g., NLTK). Chunks have been shown to minimize the generation of discontinuous sentences in previous works in machine translation (Zhang et al., 2007; Watanabe et al., 2003). We apply a trained binary Support Vector Machine (SVM) classifier to each chunk to predict whether it should be kept or discarded. The shortened definition consists of only chunks that are kept.

Lexical and syntactical features are extracted from definition chunks as well as the sentence containing the original idiom. We have also incorporated features that related previous works have found to be beneficial (Štajner et al., 2013; Narayan and Gardent, 2014). The following is a brief description of our feature set.

Features from the Sentence: These features encode the syntactic context of the idiom. One feature is the constituent label of the entire idiom from the sentence's full parse. It aims to show the big picture of the grammatical function of the idiom in the original sentence. Another feature is the part-of-speech (POS) tag of the word preceding the idiom. These features help to select definition chunks that fit better into the sentence context in which the idiom is used. Due to data sparsity and overfitting concerns, we do not extract lexical features from the sentences.

Features from the Definition: These features encode the syntactic information extracted from all the chunks that made up the definition. In addition to chunking, we also apply a full parser on the definition to obtain its dependency and constituency tree. Although the parse trees may not be reliable enough to serve as hard constraints, they offer useful syntactic information as soft evidences. For example, the dependency tree helps us to identify the head word of every chunk (denoted here as w_h). The constituency tree helps us to determine whether w_h is a node in a subordinate clause (subtree with its root labeled as 'SBAR'). This feature is useful because two adjacent chunks in a relative clause tend to be kept or discarded together. We also include features indicating the relation of the typed dependency of the chunks. Thus, if a verb chunk is kept, its arguments are also likely to be kept. Other features includes whether w_h is the root, whether w_h is the leaf node in the dependency tree. Since certain adjacent words tend to be discarded or kept together, we reinforce this property by adding a bigram POS feature of w_h to encode its context. Additionally, we extract various surface features from the chunks such as their lengths, their positions in the definition, POS of w_h, etc. Some definitions are very long and have several sub-sentences, while a good shortened definition is usually extracted from one sub-sentence. Thus, we have also included a feature indicating whether the definition has more than one sub-sentence, and if the definition has more than one sub-sentence, whether the chunk is in the first sub-sentence.

Features adapted from the Rule-Based method: These include: whether the chunk contains a boundary word, whether the preceding word of the chunk is a boundary word, whether the following word of the chunk is a boundary word, whether the chunk is in a bracket.

3.2 Post Editing

To ensure that the shortened definition is a fluent replacement for the idiom in the context of the original sentence, we must make grammatical adjustments, resolve references, and smooth over the replacement boundaries.

3.2.1 Grammatical Adjustments

We perform several agreement checks. For example, when replacing a noun phrase idiom, we need to make sure that the grammatical number of the replacement phrase agrees with how it is used in the sentence. Similarly, when replacing a verb phrase idiom, we need to perform verb tense, person and number agreement checks, such as converting *get someone angry* to *gets someone angry* in the example mentioned in Section 2.

3.2.2 Reference Resolution

Reference expression is common in definition of idiom. For example, the idiom *see eye to eye* has a shortened definition of *they agree with each other*. The referent *they* has to be resolved when we substitute the idiom with it. The general reference resolution problem is a long-standing challenge in NLP (Mitkov, 1998; Hobbs, 1978; Hobbs, 1979); even in the limited context of our idiom substitution problem, it is not trivial. While regular expression matching may work for idioms that contain simple slot replacements (e.g., the idiom *lather something up* with the definition *to apply thick soapsuds to*

something), further analyses on the idiom's sentential context are necessary for many idioms (e.g., *see eye to eye* has no obvious slot).

Typical reference expressions in a definition include *something*, *someone*, *somebody*, *you*, *they*, which often refer to noun phrases (NPs) in and around the idiom in the sentence. When the sentence context contains multiple NPs, we need to choose the right one to resolve the reference. To do so, we rely on two commonly used factors: recency and syntactic agreement (Lappin and Leass, 1994). Similar to the work of Siddharthan (2006), we extract all NPs in the original sentence with their agreement types and grammatical functions; for each NP, we assign it a score with equal weights of recency and syntactic factors. We choose the NP that satisfy the agreements and grammatical functions with the highest score, breaking ties by selecting the closest NP. When no contextual NP is suitable, we replace the reference expression with generics such as "it," "people," or "person" instead.

There is one subtle difference between reference resolution in our work and typical cases. In addition to deriving the correct interpretation of a reference expression, our system has to actually insert the referent to the shortened definition and make the paraphrased sentence grammatical. This means that we need to make the appropriate PRP (personal pronoun) and PRP$ (possessive pronouns) conversions. Consider the example from Section 2 again. the *someone* in the shortened definition is initially resolved to *my* in the original sentence, but to make the substitution grammatical, it has to be transformed to *me*. In addition, special processing is also needed when the substitution is in the form of *subordinate clause*. For example:

> **Tweet:** Maybe if the NFL stopped treating him as such, he wouldn't act like *a prima Donna.*[7]
>
> **Substitution:** *someone* who demands to be treated in a special way[8]

Although *someone* refers to *he* in the original sentence, no pronoun substitution is plausible. Therefore, *someone* is replaced by a generic expression, "a person."

3.2.3 Boundary Smoothing

Boundary smoothing is the last step of the Post-Editing process to improve the fluency of the resulting sentence. We rely on a standard n-gram language model to evaluate the "smoothness" of the transitions between the original sentence and the substitution phrase. For the left boundary, we begin by checking the bigram probability of the word immediately before the substitution and the first word of the substitution. If it is 0, we would drop the first word and recheck until we find a bigram with non-zero probability or until we have reached the fourth word, whichever occurs first. If a non-zero bigram cannot be found within the first three words, we substitute the original shortened definition as is, without any word deletion. The range of three word is chosen based on our analysis of the development set. A mirror image process is applied to the right boundary. The language model is trained via NLTK using the Brown corpus[9].

4 Evaluation

To determine the performance of the definition shortening methods and post editing operations, we have carried out two experiments. The first (Section 4.2) evaluates the quality of the substitution generation methods; we also argument the evaluation with statistical analysis of post-editing as a reference for future work. The second (Section 4.3) evaluates whether the resulting paraphrased sentence is grammatical and preserves the original meaning.

4.1 Corpus

To evaluate our method on real data, we chose to select Tweets that contain idioms. The reasons are twofold. First, the inspiration for our problem formulation was to help non-native speakers understand social media contents. The limited context of a Tweet makes it harder for someone who does not know an embedded idiom to induce its meaning from the rest of the text. Second, Tweet are self-contained, making the paraphrase task as well as its evaluation (by human judges) more stand-alone. The short context limits the set of mentioned en-

[7] https://twitter.com/KingKylino/status/678931385608966144

[8] http://idioms.thefreedictionary.com/a+prima+donna

[9] http://www.hit.uib.no/icame/brown/bcm.html

Dataset	Agree	MED_{avg}	
		Disagree	Total
Training	32.9%	**3.25**	2.18
Testing	36.9%	3.42	**2.15**
All data	34.8%	3.33	2.17

Table 3: Agreement between the two annotators. MED_{avg} represents the average minimum edit distance (by word).

tities, which helps with pronoun resolution; otherwise, we foresee no significant hurdles in applying our system to regular sentences.

To build the dataset, we randomly selected 200 idioms (100 for train and 100 for test) and automatically collected tweets in which they appeared using the query API[10]. There were six idioms for which no exact match was found; so we included the usage examples from TheFreeDictionary.com instead. We presented these sentences along with each idiom's definition and asked a volunteer native speaker (Annotator #1) to manually shorten the definition. After filtering out sentences that do not exemplify the idioms[11], we had a total of 88 instances for training and 84 for testing. Next, a near-native speaker (Annotator #2) also performed the same task so that we may compute the inter-annotator agreement. The shortened definitions from Annotator #1 are used as the gold standard.

Table 3 shows the agreement between two annotators. The overall average edit distance is 2.17 words; since the average length of the definitions is about twelve words long (cf. Table 1), the annotators have significant overlaps with each other (the Cohen's kappa is 0.64, suggesting that the inter-annotator agreement is within an acceptable range (Viera et al., 2005)). However, although the annotators extracted the exact same phrase 34.8% of the times, in general they do not completely agree. Some people may select more words to convey a more precise meaning while others sacrifice some precision in meaning for a greater fluency. Thus, in addition to measuring against the gold standard (Annotator #1) using automatic metrics, we also need to perform a human evaluation to directly judge the qualities of the paraphrases.

[10] `https://dev.twitter.com/rest/public/search`

[11] For example, "the bitter end" was used in reference to the name of a club.

4.2 Automatic evaluation

In this experiment, we compare different approaches for substitution generation using automatic metrics. We wish to determine: 1) How well does each method replicate human annotators' phrasal extractions? 2) Do we need specialized methods for extracting core meanings from idiom definitions? 3) Is the ML-based method more general and flexible?

The training data contains 88 definitions for a total of 645 chunks that have been labeled as "keep" or "discard" according to the gold standards. The test data consists of 84 unique idioms used in tweets. The evaluation metric is the minimum edit distance of each proposed substitution from the gold standard. We also calculate the *compression rate*, the ratio between numbers of tokens kept with total numbers of tokens in original sentence.

We compare our proposed methods with McDonald (2006). Specifically we use an adapted version described in Filippova et al. (2015). We also implemented two simpler baselines:

Equal-POS: Extract those words from the definition that have the same POS tags as the idiom. For example, if the idiom consists of a VB and an NN, then the first two words tagged as VB and NN in the definition are returned as the substitution. When POS matching fails, the whole definition is returned.

First-Six: Always return the first six words. We choose six because the average length of the gold standard extractions from the training set is six words long.

From the results presented in Table 4, we see that the problem of extracting the core explanation from a long definition is not trivial. The average minimum edit distances from the gold standard are high for the two simple baselines (6.29 for First-Six, 4.92 for Equal-POS). The text compression baseline, McDonald, is only a little better, at 4.86. Because the proposed methods are developed especially for idiom definitions, they are closer to the gold standard. Considering the inter-annotator agreement as an upper-bound (with an average minimum edit distance of 2.15 for the test set), the ML-based approach comes the closest to the upper-bound (with an average distance of 2.75).

Method	Agree	MED_{avg} Disagree	MED_{avg} Total	Compression Rate
First-Six	0%	6.29	6.29	37%
Equal-POS	6.0%	5.10	4.92	49%
McDonald	6.0%	5.14	4.86	**22%**
Rule-Based	23.8%	4.04	3.27	59%
ML-based	25.0%	**3.5**	**2.75**	51%

Table 4: A comparison of different substitution generation methods with gold standard. MED_{avg} denotes the average minimum edit distance of the method's extraction from the gold standard.

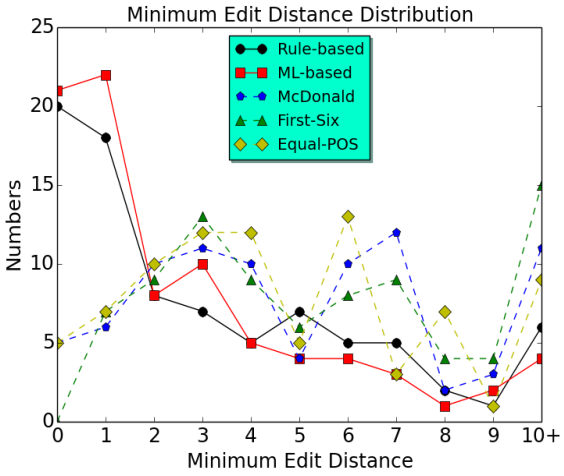

Figure 1: A distribution of edit distances for individual instances.

	Method	Grammaticality	Meaning
Def	McDonald	3.74	3.32
	Rule-Based	**4.92**	**4.71**
	ML-based	4.79	4.68
Sen	McDonald	3.44	3.27
	Rule-Based	4.25	4.31
	ML-based	**4.61**	**4.64**

Table 5: Human evaluation of the different methods in terms of the grammaticality and meaning preservation. In *Def* only the shortened definition are evaluated; in *Sen* the final paraphrased sentences are evaluated.

Figure 1 plots a distribution of each method's minimum edit distances for the instances in the test set. While the rule-based approach has a similar distribution as the trained classifier, it is almost always slightly worse. We see that half of the extractions based on the keep/discard classification are within a one word difference from the gold standard, and there are fewer than five instances for which the edit distance is at least 10; in contrast, the rule-based approach has fewer cases of (nearly) perfect matches and more cases of large mismatches (with the exception of an edit distance of 9). These results suggest that specialized methods are necessary for processing idiom definitions and that an ML-based approach is more general and flexible.

We have also measured the compression rate (CR) of each method; however, this may not be an appropriate metric for our domain in the sense that lower is not necessarily better. The CR of gold standard is 45%, while the ML-based method is 51%. Although the McDonald method has the lowest CR, at 22%, it is lower than that of the gold standard; this suggests that its approach is too aggressive.

To evaluate the contribution of the post-editing component, we have performed data analyses on each step individually: grammatical adjustment, reference resolution and boundary smoothing (using the outputs of the ML-based method). In terms of grammatical adjustments, there are two cases of noun number adjustment and five verb related adjustment.

In terms of reference resolution, we need to address not only the typical reference expressions, but also special cases relating to PRP and PRP$ conversions and subordinate clauses that was discussed in section 3.2.2. We have found fifteen cases of typical reference resolution and nine special cases, out of which, seven were related to subordinate clause (cf. Example 2 in Table 6). Finally, there are three cases for which reference expression cannot be resolved due to the lack of an appropriate noun phrase (cf. Example 5 in Table 6).

Sentence	French and British police are working *in harness* to solve the problem.
Definition	*if two or more people work in harness, they work together to achieve something*
ML-based	French and British police are *working together* to solve the problem.
Rule-Based	French and British police are *working together to achieve it* to solve the problem.
McDonald	French and British police are *working to achieve* to solve the problem.
Sentence	Don't buy *a pig in a poke*.
Definition	*something that you buy without knowing if it is good or not*
ML-based	Don't buy *something without knowing if it is good.*
Rule-Based	Don't buy *something that you buy without knowing if it is good.*
McDonald	Don't buy *without is good.*
Sentence	*Band-Aid solutions* for a homeless Senate worker.
Definition	*a temporary solution to a problem, or something that seems to be a solution but has no real effect*
ML-based	*Temporary solutions* for a homeless Senate worker.
Rule-Based	*Temporary solutions to a problem* for a homeless Senate worker.
McDonald	Or to be a for a homeless Senate worker.
Sentence	I've said all I had to say, *the ball is in your court* .
Definition	*if the ball is in someone's court, they have to do something before any progress can be made in a situation.*
ML-based	I've said all I had to say, *you have to do something.*
Rule-Based	I've said all I had to say, *you have to do something before any progress can be made in a situation.*
McDonald	I've said all I had to say, *do before can made.*
Sentence	I had to *spill my guts* about the broken window.
Definition	*to tell someone all about yourself, especially your problems*
ML-based	I had to *tell me all about myself* about the broken window .
Rule-Based	I had to *tell me all about myself* about the broken window .
McDonald	I had to *tell* about the broken window .

Table 6: Example of paraphrased sentences. The underlined pronouns in Examples 4&5 have to be resolved. Example 5 shows a failure of reference resolution.

With respect to boundary smoothing, there are many more cases of left boundary smoothing than right boundary smoothing (39 vs. 2 cases). Although many of the left boundary cases simply involve deleting the word "to" from the shortened definitions, some boundary smoothing cases do address the more severe redundancy disfluencies (cf. Example 1 in Table 6).

4.3 Human evaluation

Minimum edit distance to the gold standard cannot fully indicate the grammaticality and meaning preservation of the extracted phrase. In this experiment, we follow standard human evaluation procedures (McDonald, 2006) to verify our findings from the first experiment. The results will answer two questions: 1) Are the shortened definitions grammatical and are they representative of the core meanings? 2) Are the final paraphrased sentences grammatical and do they retain their original meanings?

We used the same 84 idioms which were the test set in the automatic evaluation. Four native speakers were recruited to evaluate the grammaticality and meaning of the shortened definitions and paraphrased sentences on a five-point scale. Each person took approximately 90 minutes to finish the study. We did not evaluate the simple baselines (First-Six and Equal-POS) because their qualities were obvi-

ously low; including them may bias the human subjects to give inflated scores to the better methods. The results are presented in Table 5.

In terms of shortening the definition, the rule-based method obtains the highest scores in both grammaticality and meaning; this is because it tends to be relatively conservative. The compression rate is 59%, while the ML-based method is 51%. Keeping more words in the definition reduces the chance of introducing grammar error and meaning loss; however, a longer definition makes poorer substitution in the full sentence because it introduces redundancy and thwarts post-editing efforts. This is validated in our experimental results – in terms of the paraphrased sentences, the rule-based method is outperformed by the ML-based method, which achieves the best result, with 4.61 in grammaticality and 4.64 in meaning.

Table 6 shows some typical examples of the paraphrases produced using substitution generation from the ML-based method, the rule-based method, and the McDonald method followed by processing with the proposed post-editing techniques. The first example features the effect of boundary smoothing. The shortened definition from the ML-based method is *work together to*. Direct replacement into the original sentence creates a disfluent bigram "working work", which has a probability of 0; thus the first word in the shortened definition (*work*) is deleted automatically. Similarly, the word *to* is deleted for the right boundary. In the third example, an automatic grammar adjustment is applied during substitution: *a temporary solution* is converted to *Temporary solutions* to keep the number consistent. In the forth example, the reference *they* is successfully resolved to *you* in the definition. The fifth example features a challenging rare case that results in a failed reference resolution.

Shortening the definition is a trade off between length and meaning. In these examples, the rule-based method keeps as many words as possible from the definition and leads to redundancy in the final output. It has a negative impact on the readability of the paraphrase. The McDonald method is too aggressive for short text such as definition, so the outputs are often discontinuous. The ML-based method offers a reasonable balance between length and meaning, and produces paraphrases that people seem to prefer.

5 Conclusion

We have proposed a phrasal substitution method for paraphrasing idiomatic expressions. Our system extracts the core meaning of an idiom from its dictionary definition and replaces the idiom with it. Empirical evaluations shows that the proposed method produces grammatical paraphrases that preserves the idioms' meanings, and it outperforms other methods such as sentence compression. In the future, we will explore the uses of the idiom paraphrases in NLP applications such as machine translation and intelligent tutor for second-language learners.

Acknowledgments

We would like to thank Ric Crabbe, Xiaobing Shi and Huichao Xue for the helpful discussions and suggestions. We also would like to thank the anonymous reviewers for their feedback.

References

Steven Bird. 2006. NLTK: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, pages 69–72. Association for Computational Linguistics.

James Clarke and Mirella Lapata. 2008. Global inference for sentence compression: An integer linear programming approach. *Journal of Artificial Intelligence Research*, pages 399–429.

Trevor Anthony Cohn and Mirella Lapata. 2009. Sentence compression as tree transduction. *Journal of Artificial Intelligence Research*, pages 637–674.

William Coster and David Kauchak. 2011. Learning to simplify sentences using wikipedia. In *Proceedings of the workshop on monolingual text-to-text generation*, pages 1–9. Association for Computational Linguistics.

Marie-Catherine De Marneffe, Bill MacCartney, Christopher D Manning, et al. 2006. Generating typed dependency parses from phrase structure parses. In *Proceedings of LREC*, volume 6, pages 449–454.

Afsaneh Fazly, Paul Cook, and Suzanne Stevenson. 2009. Unsupervised type and token identification of idiomatic expressions. *Computational Linguistics*, 35(1):61–103.

Katja Filippova and Michael Strube. 2008. Dependency tree based sentence compression. *INLG '08: Proceedings of the Fifth International Natural Language Generation Conference*, pages 25–32.

Katja Filippova, Enrique Alfonseca, Carlos A Colmenares, Lukasz Kaiser, and Oriol Vinyals. 2015. Sentence Compression by Deletion with LSTMs. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 360–368.

Jerry R Hobbs. 1978. Resolving pronoun references. *Lingua*, 44(4):311–338.

Jerry R Hobbs. 1979. Coherence and coreference. *Cognitive science*, 3(1):67–90.

Sujay Kumar Jauhar and Lucia Specia. 2012. Uow-shef: Simplex–lexical simplicity ranking based on contextual and psycholinguistic features. 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 477–481. Association for Computational Linguistics.

Ioannis Korkontzelos, Torsten Zesch, Fabio Massimo Zanzotto, and Chris Biemann. 2013. Semeval-2013 task 5: Evaluating phrasal semantics. In *Second Joint Conference on Lexical and Computational Semantics (* SEM)*, volume 2, pages 39–47.

Shalom Lappin and Herbert J Leass. 1994. An algorithm for pronominal anaphora resolution. *Computational linguistics*, 20(4):535–561.

William C Mann and Sandra A Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text-Interdisciplinary Journal for the Study of Discourse*, 8(3):243–281.

Diana McCarthy and Roberto Navigli. 2007. Semeval-2007 task 10: English lexical substitution task. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 48–53. Association for Computational Linguistics.

Diana McCarthy. 2002. Lexical substitution as a task for wsd evaluation. In *Proceedings of the ACL-02 workshop on Word sense disambiguation: recent successes and future directions-Volume 8*, pages 109–115. Association for Computational Linguistics.

Ryan T McDonald. 2006. Discriminative sentence compression with soft syntactic evidence. In *Proc. of EACL-06*, pages 297–304.

Ruslan Mitkov. 1998. Robust pronoun resolution with limited knowledge. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 2*, pages 869–875. Association for Computational Linguistics.

Shashi Narayan and Claire Gardent. 2014. Hybrid Simplification using Deep Semantics and Machine Translation. *Acl*, pages 435–445.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Giancarlo D Salton, Robert J Ross, and John D Kelleher. 2014. An Empirical Study of the Impact of Idioms on Phrase Based Statistical Machine Translation of English to Brazilian-Portuguese. *The Third Workshop on Hybrid Approaches to Translation (HyTra 2014)*.

Advaith Siddharthan. 2006. Syntactic simplification and text cohesion. *Research on Language & Computation*, 4(1):77–109.

Advaith Siddharthan. 2011. Text simplification using typed dependencies: a comparison of the robustness of different generation strategies. *Proceedings of the 13th European Workshop on Natural Language Generation*, (September):2–11.

Advaith Siddharthan. 2014. A survey of research on text simplification. *International Journal of Applied Linguistics*, 165(2):259–298.

Lucia Specia, Sujay Kumar Jauhar, and Rada Mihalcea. 2012. Semeval-2012 task 1: English lexical simplification. 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 347–355. Association for Computational Linguistics.

Caroline Sporleder and Linlin Li. 2009. Unsupervised recognition of literal and non-literal use of idiomatic expressions. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 754–762. Association for Computational Linguistics.

Anthony J Viera, Joanne M Garrett, et al. 2005. Understanding interobserver agreement: the kappa statistic. *Fam Med*, 37(5):360–363.

Martin Volk. 1998. The automatic translation of idioms. machine translation vs. translation memory systems. *Machine Translation: Theory, Applications, and Evaluation, An Assessment of the State-of-the-art, St. Augustin, Gardez Verlag*.

Sanja Štajner, Biljana Drndarevíc, and Horacio Saggion. 2013. Corpus-based sentence deletion and split decisions for Spanish text simplification. *Computacion y Sistemas*, 17(2):251–262.

Taro Watanabe, Eiichiro Sumita, and Hiroshi G Okuno. 2003. Chunk-based statistical translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 303–310. Association for Computational Linguistics.

Lowri Williams, Christian Bannister, Michael Arribas-Ayllon, Alun Preece, and Irena Spasić. 2015. The role of idioms in sentiment analysis. *Expert Systems with Applications*.

Sander Wubben, Antal Van Den Bosch, and Emiel Krahmer. 2012. Sentence simplification by monolingual machine translation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 1015–1024. Association for Computational Linguistics.

Fabio Massimo Zanzotto, Ioannis Korkontzelos, Francesca Fallucchi, and Suresh Manandhar. 2010. Estimating linear models for compositional distributional semantics. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 1263–1271. Association for Computational Linguistics.

Yuqi Zhang, Richard Zens, and Hermann Ney. 2007. Chunk-level reordering of source language sentences with automatically learned rules for statistical machine translation. In *Proceedings of the NAACL-HLT 2007/AMTA Workshop on Syntax and Structure in Statistical Translation*, pages 1–8. Association for Computational Linguistics.

Zhemin Zhu, Delphine Bernhard, and Iryna Gurevych. 2010. A monolingual tree-based translation model for sentence simplification. In *Proceedings of the 23rd international conference on computational linguistics*, pages 1353–1361. Association for Computational Linguistics.