Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence

Fast and Accurate Prediction of Sentence Specificity

Junyi Jessy Li and **Ani Nenkova**
University of Pennsylvania
Philadelphia, PA 19104
{ljunyi,nenkova}@seas.upenn.edu

Abstract

Recent studies have demonstrated that specificity is an important characterization of texts potentially beneficial for a range of applications such as multi-document news summarization and analysis of science journalism. The feasibility of automatically predicting sentence specificity from a rich set of features has also been confirmed in prior work. In this paper we present a practical system for predicting sentence specificity which exploits only features that require minimum processing and is trained in a semi-supervised manner. Our system outperforms the state-of-the-art method for predicting sentence specificity and does not require part of speech tagging or syntactic parsing as the prior methods did. With the tool that we developed — SPECITELLER — we study the role of specificity in sentence simplification. We show that specificity is a useful indicator for finding sentences that need to be simplified and a useful objective for simplification, descriptive of the differences between original and simplified sentences.

1 Introduction

Sentences vary in the level of details they convey. Clearly written texts tailor the specificity of content to the intended reader and exhibit clear patterns in the flow of specificity (Scanlan 2000; Higgins et al. 2004). Consider the following two sentences, talking about test cheating:

[general] Evidence of widespread cheating has surfaced in several states in the last year or so.

[specific] California's education department suspects adult responsibility for erasures at 40 schools that changed wrong answers to right ones on a statewide test.

Both sentences convey the information that (exam) cheating is taking place. The first sentence is rather general: it contains a vague piece of information on the extent of cheating (*widespread*), location of cheating (*several states*) and time of the cheating (*in the last year or so*) and says nothing about exactly what cheating consisted of. The second is more specific, conveying that it was not students who did the cheating and that 40 schools in California were suspected and what activities constituted the cheating, making the extent, location and exact events much more precise.

It is clear that the specificity of text is expressed on multiple levels, and can be quantified at the level of words (*people* vs. *students* vs. *Mary Smith*) (Reiter and Frank 2010; Krahmer and van Deemter 2012), sentences, as in the example above (Mathew and Katz 2009; McKinlay and Markert 2011), or full texts or paragraphs, where the distinction boils down to determining if the intended audience of a text are lay people or experts (Elhadad et al. 2005).

For the work presented in this paper, we focus on the task of predicting sentence level specificity. Indictors of word specificity are used as features in the prediction task, and overall text specificity can be computed as a function of the specificity of the individual sentences in the text. This framework was introduced by Louis and Nenkova (2011a). In that work, specificity at the sentence level was already indirectly labeled in a corpus originally developed for a different type of analysis in discourse processing. Supervised methods for sentence specificity trained on this repurposed data, using a rich set of syntactic and lexical features, yielded good accuracy on a test set of sentences manually labeled for specificity (Louis and Nenkova 2012). These experiments confirmed that predicting sentence specificity is a viable task. The authors further used their automatic classifier to show differences in content specificity in summarization, where human summaries are significantly less specific than the original text being summarized but machine summaries are significantly more specific (Louis and Nenkova 2011b). They also showed that considerations of sentence specificity are at play in text quality assessment: science journalism articles that were considered to be written exceptionally well were overall more general and contained fewer stretches of contiguous specific sentences than other science journalism pieces (Louis and Nenkova 2013).

Despite the positive findings, several crucial questions remain. First, it remains unclear if specificity can be computed quickly so that it can become practical as a module in realistic applications. The original method incorporated syntactic, part of speech, named entity and language model features. In our work we explore light features that can be computed with string operations alone, possibly allowing look up in a static dictionary. We present experiments to show that such an impoverished feature set will still be able to achieve results better than chance but considerably inferior to the state of the art approach. However, using co-training in a semi-

supervised approach (Blum and Mitchell 1998), we can exploit unlabeled text corpora to increase the amount of available training data and significantly outperform the accuracy of prediction of the state of the art system.

The second question concerns the value of lexical features in predicting sentence specificity. The experiments presented by Louis and Nenkova show that word identity features are not robust. We separately study word identity features, dictionary look-up representation (polarity and concreteness) and word embedding and clustering representations. Our experiments show that the two latter representations of lexical content are powerful and robust when trained on a large dataset using a semi-supervised approach.

Finally, we turn to the question of validating the usefulness of sentence specificity prediction in a task-based manner. Sentence simplification, the task of identifying sentences that may be difficult for a target audience and rewording them to make them more accessible for the target audience, is a suitable task. We applied our automatic predictors of sentence specificity on sentence simplification data and showed that specificity scores at the sentence level are useful both for detecting sentences that need to be simplified and as a component of the objective during simplification.

In sum, in this paper we describe and provide a simple, accurate and practical predictor of sentence specificity: SPECITELLER[1]. Furthermore, we investigate the relationship between word properties and overall sentence specificity. We demonstrate that generalized representations of lexical identity are robust and useful representations for the task when coupled with semi-supervised techniques that allow us to considerably enlarge the training set. We also provide sentence-level task-based analysis of the utility of predicting sentence specificity in text simplification.

2 Data for sentence specificity

To train our semi-supervised model for sentence specificity, we follow prior work (Louis and Nenkova 2011a) and use a repurposed corpus of binary annotations of specific and general sentences drawn from Wall Street Journal articles originally annotated for discourse analysis (Prasad et al. 2008). We then make use of unlabeled data for co-training. The unlabeled data is extracted from the Associated Press and New York Times portions of the Gigaword corpus (Graff and Cieri 2003), as well as Wall Street Journal articles from the Penn Treebank corpus selected so that there is no overlap between them and the labeled training examples and the testing data.

For evaluation, we use the set of manual annotation of specificity by five annotators (Louis and Nenkova 2012). Annotations cover 885 sentences from nine complete news articles from three sources—Wall Street Journal, New York Times and Associated Press. In this dataset, 54.58% of the sentences are labeled specific. This is the same test data used for evaluation of earlier work on predicting sentence specificity, thus comparisons with prior work are straightforward.

The annotated training data comes from adjacent sentences in the same paragraph between which an implicit INSTANTIATION relation holds. In this relation the second sentence describes in further detail a set (of events, reasons, behaviors or attitudes) introduced in the first sentence (Miltsakaki et al. 2008), as illustrated in the example below:

> **[S1** He says he spent $300 million on his art business this year. **]** **[S2** A week ago, his gallery racked up a $23 million tab at a Sothebys auction in New York buying seven works, including a Picasso. **]**

In our work we do not preserve any information on the adjacency of sentences. We simply use the first argument as an example labeled *general* and the second as an example labeled *specific*. There are 2,796 training instances in total.

3 Light features for sentence specificity

Our goal is to develop a set of simple features which do not incur much computational or memory overhead. For the prediction we use only sentence splitting, descriptive statistics of the sentence string, dictionary features and non-sparse lexical representations.

3.1 Shallow features

Sentence surface features We use seven features capturing sentence surface characteristics. Among these, the number of words in the sentence is an important feature because on average specific sentences tend to be longer. To approximate the detection of named entities, we introduce features to track the number of numbers, capital letters and non-alphanumeric symbols in the sentence as three features, normalized by the number of words in the sentence. Symbols include punctuation so this feature captures a rudimentary aspect of syntactic complexity indicated by the presence of commas, colons and parenthesis. We also include a feature that is the average number of characters in the words that appear in the sentence, with the intuition that longer words are likely to be more specific. We also include as features the number of stop words in the sentence normalized by the total number of words, with the intuition that specific sentences will have more details, introduced in prepositional phrases containing prepositions and determiners. We use a predefined list of 570 stop words provided by the NLTK package. We also include as a feature the count of the 100 words that can serve as explicit discourse connectives (Prasad et al. 2008) because explicit discourse relations within the sentence, such as elaboration or contingency, may signal that extra information is present for some of the clauses in the sentence.

Dictionary features These are lexical features that capture the degree to which words in the sentence have a given property. Louis and Nenkova (2011a) observed that general sentences tend to be more subjective. Like them, we also include the number of polar and strongly subjective words (normalized by sentence length), according to the General Inquirer (Stone and Hunt 1963) and MPQA (Wilson, Wiebe, and Hoffmann 2009) lexicons to define two sentence features.

We also include two other dictionary features that have not been explored in prior work. We use the word norms

[1] http://www.cis.upenn.edu/∼nlp/software/speciteller.html

from the MRC Psycholinguistic Database (Wilson 1988). These are average ratings by multiple subjects of the familiarity, concreteness, imageability and meaningfulness of the word given by multiple people. We computed the cumulative ratings for words in specific and general sentences in the supervised portion of our training data. The familiarity (how familiar the word was to the subjects) and imageability (to what extent the word evoked an image according to the subjects) were significantly higher for general sentences compared to specific sentences in the "general" portion of the training data. The difference with respect to the other properties was small. So we record the average word familiarity and imageability ratings in the sentence as features.

Finally, we capture the informational value of words as approximated by their inverse document frequency (idf) weight calculated on the entire set of New York Times articles from 2006 (Sandhaus 2008). Very common words have low idf weight and fairly rare words have high idf. We compute the minimum, maximum and average inverse document frequency values of words in each sentence, accounting for three new sentence features in this representation.

3.2 Non-sparse word representations

It stands to reason that lexical features would be helpful in predicting sentence specificity, with general words characterizing general sentences. However, prior work (Louis and Nenkova 2011a) reported that word identity representations gave very unstable results for the sentence specificity prediction task. These findings can be explained by the fact that their method is fully supervised and the training set contains fewer than three thousand sentences. In that data, only 10,235 words occur more than three times. So in new test data many sentences would have few non-zero representations other than function words because few of the content words in the training data appear in them[2]. Overall there will be only weak evidence for the association between the feature and the specificity classes.

We explore two alternative representations that encode lexical information in a more general manner, tracking the occurrence of clusters of words or representing words in low dimensional dense vector space.

Word identity For comparison with prior work and as a reference for the non-sparse representations, we train a predictor for sentence specificity based on a word identity representation. Each word that occurs in the training data more than three times corresponds to a feature. The value of the feature is the number of times the word occurs in the sentence. When using the initial training data this representation is equivalent to the problematic one discussed in earlier work. During co-training, the training set is augmented and more words are included in the representation.

Brown clusters Brown clusters (Brown et al. 1992) are compact representations of word classes that tend to appear in adjacent positions in the training set. They were originally proposed as a way of dealing with lexical sparsity for bi-gram language models. In our work, we use the precomputed hierarchical clusters provided by Turian, Ratinov, and Bengio (2010). The clusters are derived from the RCV1 corpus which consists of about 34 million words. Each feature in this representation corresponds to a cluster and the value of the feature is the number of occurrence in the sentence of any of the words in the cluster. The number of clusters is a parameter of the representation which we tuned with 10-fold cross validation on the labeled training data. We use 100 clusters for the results reported here.

[2]About 40% of our test instances have fewer than 4 content words that can be found in the labeled training data.

Features	Accuracy	Precision	Recall
SF	71.53	66.52	75.12
WP	72.43	69.85	69.15
Shallow (SF+WP)	73.56	69.44	74.63
BC	70.85	66.59	71.89
WE	68.25	65.24	64.43
BC+WE	71.64	70.03	65.67
Word identity	63.39	58.48	66.92

Table 1: Supervised learning results: accuracy and precision and recall the general class, for sentence surface features (SF), word properties (WP), combined shallow features (SF+WP), brown clustering (BC), word embeddings (WE), words (Word identity).

Neural network word embeddings Real-valued word vectors are a natural product from neural network language models. In these models words are represented in low dimensional space that capture the distributional properties of words (Mikolov, Yih, and Zweig 2013). In our experiments we use the 100-dimensional word vector representations provided by Turian, Ratinov, and Bengio (2010). To represent a sentence in this space, we average the representations of the words in the sentence, i.e, component i of the sentence representation is equal to the average value of component i for the representations of all words in the sentence.

3.3 Supervised learning results

First we evaluate the feature classes introduced above in a standard supervised learning setting. We used the labeled training set to train a logistic regression classifier. We choose logistic regression in order to use the posterior class probability of an example being specific as a continuous measure of sentence specificity in later experiments. The models are tested on the human labeled test data.

In Table 1 we list the overall accuracy and the precision/recall for the general sentences achieved with each feature representation. For this test set, the majority baseline would give a 54.58% accuracy.

The class of shallow features performs reasonably well, achieving accuracy of 73.56%. This result is better than individually using surface features or word property dictionary features alone. As reported in prior work, word identity features work poorly and lead to results that are almost 10% worse than the shallow features. The non-sparse representations perform markedly better. The Brown cluster representation almost closes the gap between the lexical and

shallow features with accuracy of close to 71%. Combining this with the word embedding representation leads to further small improvements. These results show that it is feasible to predict specificity based on cheaply computable features alone and that non-sparse representations of lexical information are more suitable for the relatively small training set.

4 Semi-supervised learning via co-training

In co-training, two classifiers are trained on a labeled dataset. Then they are used iteratively to classify a large number of unlabeled examples, expanding the labeled data on which they are re-trained. An important characteristic that ensures improved performance is that the two classifiers are independent relying on different views of the data to make decisions about the class. In our work, the shallow features and the non-sparse lexical representation provide such different views on the data, as reflected by the different precision and recall values shown in Table 1.

4.1 Co-training

The co-training procedure for identifying general/specific sentences is detailed in Algorithm 1. It aligns with the traditional algorithm, except that we have one additional constraint as how new labeled data are added. The procedure can be viewed as a two-phase process: a supervised learning phase and a bootstrapping phase.

During the supervised learning phase, two classifiers are trained on the data from the implicit INSTANTIATION discourse relation: one with shallow features (C_0), the other with word representation features (C_1).

For the bootstrapping phase, the classifiers will take turns to label examples for each other. In each iteration, one classifier (C_i) will label each instance in the unlabeled examples. Then, at most p positive examples and n negative examples most confidently labeled are removed from the unlabeled set and added to the labeled examples. Here we set the values $p = 1000, n = 1500$. This 1:1.5 ratio is selected by tuning the accuracy of prediction on the initial discourse training data after 30,000 new examples are added.

We impose a further constraint that the posterior probability of a new example given by C_i must be greater than a threshold α_i. The value of α_i is determined via 10-fold cross validation on the labeled training data. We choose the lowest threshold for which the prediction accuracy of the classifier on sentences with posterior probability exceeding the threshold is greater than 85%. This thresholds turned out to be 0.8 for both classifiers. To prevent a highly imbalanced data distribution, we use a procedure *downsample*(K, γ) in each iteration when newly labeled data is added, in which we restrict the number of samples added in the larger class to be at most $\gamma = 2$ times the size of the smaller class.

The expanded labeled examples now contain the original labeled data from discourse annotations as well as initially unlabeled instances that were confidently labeled by C_i. Now, the other classifier C_{1-i} will be re-trained using the updated labeled examples, resulting in a new classifier C'_{1-i}. C'_{1-i} will then be used to label the remaining unlabeled examples, to expand the labeled training set for C_i.

Algorithm 1 Co-training algorithm for predicting sentence specificity

$L \leftarrow$ Labeled training examples
$U \leftarrow$ Unlabeled examples
$F_1 \leftarrow$ shallow features
$F_2 \leftarrow$ word representation features
for $i \leftarrow 0$ **to** 1 **do**
  Train classifier C_i over L using features F_i
end for
while $U \neq \emptyset$ **and** $|U|$ shrunk in the last iteration **do**
  for $j \leftarrow 0$ **to** 1 **do**
    $i \leftarrow 1 - j$
    C_i labels each example in U
    $P \leftarrow p$ examples in U most confidently labeled $+1$
    $N \leftarrow n$ examples in U most confidently labeled -1
    $K \leftarrow \{p \cup n \mid Pr_i(1|p_{\in P}) > \alpha_i, Pr_i(-1|n_{\in N}) > \alpha_i\}$
    $K' \leftarrow$ downsample(K, γ)
    $L \leftarrow L + K', U \leftarrow U - K'$
    Re-train C_j over L using features F_j
  end for
end while

The two classifiers will alternate in this fashion to label examples for each other from the unlabeled data, until no more unlabeled examples can be added.

The final prediction on the test data is decided based on the average posterior probability of labeling the sentence general from the two classifiers.

4.2 Experimental results

To illustrate the effect of the larger training set obtained in co-training, we plot the classifier performance as a function of the amount of unlabeled data used for the experiments. In Figure 1 we show the accuracies of our semi-supervised classifiers: *i)* the dotted line represents the classifier using word representation features (brown clustering and word embeddings); *ii)* the dashed line represents the classifier using shallow features; and *iii)* the solid line represents the final *combined* classifier. The number of unlabeled data added increases from 0 to 50,000 examples, with a 2,000 step size.

The leftmost dots in Figure 1 correspond to accuracies without adding any unlabeled data. Initially all three classifiers gain in performance as the size of the unlabeled data grows. The performance peaks when 34,000 unlabeled examples and flattens out after this point; increasing the size of the unlabeled data is not helpful beyond this point.

At first, in each iteration, the shallow classifier almost always outperforms the word representation classifier. However, as more unlabeled examples are added, the combined classifier gains better performance as the word representation classifier becomes better and more stable. This may be due to the fact that with more data, word representations capture more and more semantic information in the sentences. Eventually, the combined classifier is much better than either one of the individual classifiers.

We thus fix our final model as the combined classifier when the benefit of adding more unlabeled data in the co-training algorithm begins to diminishes (i.e., at 34,000 unlabeled examples). In Table 3, we show the accuracy, pre-

Newly labeled general sentences	Newly labeled specific sentences
1. Edberg was troubled by inconsistent serves. 2. Demands for Moeller's freedom have been a feature of leftist demonstrations for years. 3. But in a bizarre bit of social engineering, U.S. occupation forces instructed Japanese filmmakers to begin showing on-screen kisses. 4. Although many of the world's top track and field stars are Americans, the sport has suffered from a lack of exposure and popularity in the United States.	1. Shipments fell 0.7 percent in September. 2. Indian skipper Mohammed Azharuddin won the toss and decided to bat first on a slow wicket. 3. He started this week as the second-leading rusher in the AFC with 1,096 yards, just 5 yards behind San Diego's Natrone Means. 4. The other two, Lt. Gen. Cedras and Brig. Gen. Philippe Biamby, resigned and fled into self-imposed exile in Panama two days before Aristide's U.S.-backed homecoming on Oct. 15.

Table 2: Examples of general and specific sentences newly labeled during the co-training procedure.

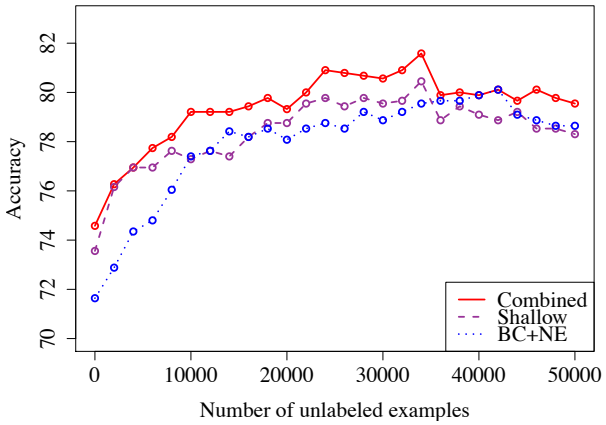

Figure 1: Accuracies with increasing size of unlabeled data.

cision, recall and F measure of the model on the human labeled test set. Also listed is the performance of the model proposed by Louis and Nenkova (2011a). A sign test was conducted and showed that both the combined model and the shallow model obtained via co-training is significantly better than Louis and Nenkova (2011a) at 95% confidence level. Furthermore, we observe a nearly 4% increase in F measure for the combined model and higher F measure for both the shallow and word representation model after co-training. At the end of the co-training stage, with only surface features, both shallow and word representation classifiers outperform that in Louis and Nenkova (2011a).

Again, to demonstrate the effect of using word representations, we run the co-training procedure where we substitute the word representation classifier with one that is trained from word identity representations as described in Section 3.3. Even with more data added, lexical identity representation do not perform that well. The increased size of the training data however helps to boost the performance of the the word identity representations immensely from the condition when only the original labeled data is used for training.

In Table 2, we show several examples of sentences from the unlabeled data that were labeled during co-training.

Classifier	Accuracy	Precision	Recall	F
Combined	81.58*	80.56	78.36	79.45
Shallow	80.45*	79.74	76.37	78.02
BC+NE	79.55	77.42	77.61	77.52
Word identity	69.83	65.10	72.39	68.55
L&N	77.40	74.40	76.62	75.49

Table 3: Accuracies for the best final stage of co-training. An asterisk (*) denotes significantly better than the model proposed by L&N (Louis and Nenkova 2011a) at 95% confidence level according to sign test.

5 Sentence specificity and text simplification

In this section we present a form of task-based evaluation for the accurate classifier for sentence specificity trained entirely on fast-to-compute surface features. We discuss the role of sentence specificity in text simplification applications. Specifically we wish to quantify the extent to which specificity changes during sentence simplification and to determine if sentence specificity is a useful factor for determining if a sentence needs to be simplified in the first place.

To give context to our findings, we also analyze the relationship between simplification and sentence length, automated readability index (ARI)[3] and language model perplexity[4]. We carry out analysis on two aligned corpora: Simple Wikipedia/Wikipedia and Britannica Elementary/Encyclopedia Britannica.

The Wikipedia corpus (Kauchak 2013) features automatic aligned sentence pairs from the Simple Wikipedia and the original English Wikipedia. The dataset consists of 167,689 aligned pairs, among which about 50K are the same sentences across Simple and original Wikipedia.

The Britannica corpus is created by (Barzilay and Elhadad 2003). People were asked to align sentences that share semantic content from several articles in the Britannica Elementary and the original Encyclopedia Britannica. There is

[3]We also considered Kincaid, Coleman-Liau, Flesh Reading Ease, Gunning Fog Index, LIX, SMOG and RIX. ARI was the readability measure that showed biggest difference in readability between original and simplified sentences.

[4]Our language model is trained on the New York Times articles from 2006. It is a trigram model using Good-Turing discounting, generated by SRILM (Stolcke and others 2002).

		%pairs	μ-simplified	μ-original
Wikipedia	ARI	73.60	9.76	12.94
	specificity	70.86	0.57	0.70
	perplexity	62.99	1272.61	1539.48
	length	55.19	23.74	27.57
Britannica	ARI	82.14	8.82	14.13
	specificity	77.12	0.45	0.70
	perplexity	74.29	635.50	1038.36
	length	73.42	19.75	30.10

Table 4: Mean values for each attribute and the percentage of pairs with lower attribute values for simplified sentences.

attribute	Wikipedia	Britannica
ARI	0.6158	0.7019
specificity	0.6144	0.6923
length	0.5454	0.6154
perplexity	0.3966	0.3308

Table 5: Precision for identifying sentences to simplify.

only one pair where the two sentences are the same.

5.1 Specificity as simplification objective

First, we studied the extent to which simple and original texts in the two corpora vary in terms of their automatically predicted specificity. We contrast these with the differences in average sentence length, average sentence readability and perplexity. For both corpora, we excluded pairs where the simplified version and the original are identical.

For both corpora, there can be more than one sentence on each side of an aligned pair. So to measure specificity, we first classify each sentence in each pair of the corpora using the final combined classifier obtained from co-training. Following the definition in Louis and Nenkova (2011a), the specificity of side $i \in \{\text{simplified,original}\}$ of a pair p is calculated as:

$$spec(p_i) = \frac{1}{\sum_{s \in p_i} |s|} \sum_{s \in p_i} |s| \times Pr(\text{specific}|s) \quad (1)$$

Here s denotes a sentence in p_i, $|s|$ denotes the length of the sentence and $Pr(\text{specific}|s)$ denotes the posterior probability of the classifier assigning sentence s as specific.

In Table 4, we show the average value of the attributes for simplified and original sides. For all attributes, we observe a significant ($p < 0.01$) drop in their values for the simplified sentences. More importantly shown in Table 4 are the percentage of pairs for each attribute where the simplified side has a lower value than the original side. The higher the percentage, the more one would expect that the attribute needs to be explicitly manipulated in a procedure for sentence simplification. Not surprisingly, the highest value here is for ARI, as improved readability is the goal of simplifying sentences for junior readers. Specificity score closely follow ARI, with about 71% and 77% of the simplified sentences showing lower specificity in the Wikipedia and Britannica corpora respectively. The numbers are much higher than those for sentence length and perplexity.

attribute A	attribute B	Wikipedia	Britannica
length	ARI	0.7897	0.7822
specificity	length	0.6996	0.7669
specificity	ARI	0.5975	0.6788
specificity	perplexity	0.3695	0.5306
perplexity	ARI	0.2454	0.3597
length	perplexity	0.1073	0.2293

Table 6: Spearman correlation for the attributes.

5.2 Identifying simplification targets

Now we analyze if specificity is an indicator that an *individual* sentence should be simplified in the first place. We train a predictor to detect a sentence that needs to be simplified with each of the sentence attributes in turn. Our positive training examples are those original sentences that have been simplified. All other sentences, including all of the examples where the simple and the original sentences are the same, serve as negative examples. We report the precision of each single-attribute classifier in identifying sentences in the original data that need to be simplified.

In Table 5 we show for each attribute, the precision for identifying sentences that need to be simplified, obtained by logistic regression via 10-fold cross-validation[5]. We also record in Table 6 the Spearman correlation between the attributes. For both corpora, sentence specificity is the second best attribute, closely following ARI with less than 1% difference in precision. Sentence length itself is not that good to identify which sentences require simplification. Perplexity from language model is the least helpful for this task. The correlation between specificity and ARI are not very high, indicating that the two attributes complement each other, each being useful as an indicator.

6 Conclusion

We presented a new model for identifying sentence specificity via co-training based on surface features that are easy and fast to compute. We make use of complementary surface features derived from sentence and word properties as well as non-sparse word representations. The result is a lightweight model free of heavy text-preprocessing requirements that significantly outperformed the model proposed in prior work, which we make available in our tool SPECITELLER. Using the model, we also analyze the impact of sentence specificity on sentence simplification. We showed that sentence specificity is not only a useful objective for simplification, but also indicative in identifying sentences that need simplification.

References

Barzilay, R., and Elhadad, N. 2003. Sentence alignment for monolingual comparable corpora. In *Proceedings of*

[5]We downsampled the negative class for the Wikipedia corpus such that the positive and negative classes are of the same size.

the 2003 Conference on Empirical Methods in Natural Language Processing (EMNLP), 25–32.

Blum, A., and Mitchell, T. 1998. Combining labeled and unlabeled data with co-training. In *Proceedings of the Annual Conference on Computational Learning Theory (COLT)*, 92–100.

Brown, P. F.; deSouza, P. V.; Mercer, R. L.; Pietra, V. J. D.; and Lai, J. C. 1992. Class-based n-gram models of natural language. *Computational Linguistics* 18(4):467–479.

Elhadad, N.; Kan, M.-Y.; Klavans, J.; and McKeown, K. 2005. Customization in a unified framework for summarizing medical literature. *Artificial Intelligence in Medicine* 33(2):179 – 198. Information Extraction and Summarization from Medical Documents.

Graff, D., and Cieri, C. 2003. English gigaword ldc2003t05. In *Philadelphia: Linguistic Data Consortium (LDC)*.

Higgins, D.; Burstein, J.; Marcu, D.; and Gentile, C. 2004. Evaluating multiple aspects of coherence in student essays. In *HLT-NAACL 2004: Main Proceedings*, 185–192.

Kauchak, D. 2013. Improving text simplification language modeling using unsimplified text data. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1537–1546.

Krahmer, E., and van Deemter, K. 2012. Computational generation of referring expressions: A survey. *Computational Linguistics* 38(1):173–218.

Louis, A., and Nenkova, A. 2011a. Automatic identification of general and specific sentences by leveraging discourse annotations. In *Proceedings of International Joint Conference on Natural Language Processing (IJCNLP)*, 605–613.

Louis, A., and Nenkova, A. 2011b. Text specificity and impact on quality of news summaries. In *Proceedings of the Workshop on Monolingual Text-To-Text Generation (MTTG)*, 34–42.

Louis, A., and Nenkova, A. 2012. A corpus of general and specific sentences from news. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*.

Louis, A., and Nenkova, A. 2013. A corpus of science journalism for analyzing writing quality. *Dialogue & Discourse* 4(2):87–117.

Mathew, T. A., and Katz, E. G. 2009. Supervised categorization for habitual versus episodic sentences. In *Sixth Midwest Computational Linguistics Colloquium*, 2–3.

McKinlay, A., and Markert, K. 2011. Modelling entity instantiations. In *Proceedings of the International Conference Recent Advances in Natural Language Processing (RANLP)*, 268–274.

Mikolov, T.; Yih, W.-t.; and Zweig, G. 2013. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 746–751.

Miltsakaki, E.; Robaldo, L.; Lee, A.; and Joshi, A. 2008. Sense annotation in the penn discourse treebank. In *Computational Linguistics and Intelligent Text Processing*, volume 4919 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg. 275–286.

Prasad, R.; Dinesh, N.; Lee, A.; Miltsakaki, E.; Robaldo, L.; Joshi, A.; and Webber, B. 2008. The Penn Discourse TreeBank 2.0. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*.

Reiter, N., and Frank, A. 2010. Identifying generic noun phrases. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 40–49.

Sandhaus, E. 2008. he new york times annotated corpus ldc2008t19. In *Philadelphia: Linguistic Data Consortium (LDC)*.

Scanlan, C. 2000. *Reporting and writing: Basics for the 21st century*. Harcourt College Publishers.

Stolcke, A., et al. 2002. SRILM - An extensible language modeling toolkit. In *Proceedings of the International Conference on Spoken Language Processing (INTERSPEECH)*, volume 2, 901–904.

Stone, P. J., and Hunt, E. B. 1963. A computer approach to content analysis: Studies using the general inquirer system. In *Proceedings of the May 21-23, 1963, Spring Joint Computer Conference (AFIPS)*, 241–256.

Turian, J.; Ratinov, L.-A.; and Bengio, Y. 2010. Word representations: A simple and general method for semi-supervised learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, 384–394.

Wilson, T.; Wiebe, J.; and Hoffmann, P. 2009. Recognizing contextual polarity: An exploration of features for phrase-level sentiment analysis. *Computational Linguistics* 35(3):399–433.

Wilson, M. 1988. MRC psycholinguistic database: Machine-usable dictionary, version 2.00. *Behavior Research Methods, Instruments, & Computers* 20(1):6–10.