Harnessing Popularity in Social Media for Extractive Summarization of Online Conversations

Ryuji Kano[†] Yasuhide Miura[†] Motoki Taniguchi[†]
Yan-Ying Chen[‡] Francine Chen[‡] Tomoko Ohkuma[†]
[†]Fuji Xerox Co., Ltd.
{kano.ryuji, yasuhide.miura, motoki.taniguchi, ohkuma.tomoko}
@fujixerox.co.jp
[‡]FX Palo Alto Laboratory
{yanying, chen}@fxpal.com

Abstract

We leverage a popularity measure in social media as a distant label for extractive summarization of online conversations. In social media, users can vote, share, or bookmark a post they prefer. The number of these actions is regarded as a measure of popularity. However, popularity is not determined solely by content of a post, e.g., a text or an image it contains, but is highly based on its contexts, e.g., timing, and authority. We propose *Disjunctive model* that computes the contribution of content and context separately. For evaluation, we build a dataset where the informativeness of comments is annotated. We evaluate the results with ranking metrics, and show that our model outperforms the baseline models which directly use popularity as a measure of informativeness.

1 Introduction

Online conversations are increasingly significant for communication, e.g., Slack[1] for work communication and Reddit[2] for general discussion. To organize overwhelming information from these conversations, researchers have been working on summarizing online conversations (Bhatia et al., 2014; Carenini et al., 2007; Mehdad et al., 2013, 2014; Oya et al., 2014). State-of-the-art models in both abstractive (Rush et al., 2015) and extractive (Cheng and Lapata, 2016) summarization tasks are based on neural networks, but these models require large amounts of training data. In previous research, these data were created automatically by retrieving headlines and highlights of news articles edited by news editors. However, these methodologies cannot be applied to the summarization of online conversations because of a lack of summary annotations.

Distant labels have been used to train models, thereby reducing the need for manual labeling; some of these labels were also applied to the summarization task. Categories of news articles (Isonuma et al., 2017) and ratings of online reviews (Xiong and Litman, 2014) were used as distant labels in extractive summarization. However, these have been used as supplementary labels to enhance conventional summarization models, whereas we present labels which a model can solely be trained with.

We leverage a measure of popularity as a distant label. In social media, users can vote, share, or bookmark a post they prefer, and the number of these actions are regarded as indications of popularity. We assume that measures of popularity reflects the *informativeness*, the index required for a summary (Erkan and Radev, 2004), and validate whether popularity can be used as a distant label for extractive summarization. However, popularity is not solely determined by content, e.g., a text or an image, but is highly affected by contexts, e.g., timing, and authority (Cheng et al., 2017; Burghardt et al., 2017; Suh et al., 2010; Hessel et al., 2017; Jaech et al., 2015). Therefore, to utilize popularity as an indicator of informativeness, we need to exclude the effect of context.

To exclude the effect, we propose *Disjunctive model.* This model computes two scalar values; one from a content feature and the other from a context feature. These two values are then multiplied to predict the popularity. The scalar values can be interpreted as the contribution of content and context to the prediction. We assume that the contribution of content to indicate informativeness.

For evaluation, we build a test dataset where comments are annotated for informativeness. We measure informativeness as an index indicative of the best sentences to extract as a summary. We

[1] https://slack.com
[2] https://www.reddit.com

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 1139–1145
Brussels, Belgium, October 31 - November 4, 2018.

select Reddit as a data source, where the *karma score*, a measure of popularity in Reddit, is known to be affected by contexts. Our test task is to extract informative posts. Because informativeness of each post is annotated via crowdsourcing, the extracts can be ranked, but the appropriate number is unknown. Therefore, we employ ranking metrics in the evaluation. Our experiment only use karma scores for training to verify that they reflect informativeness. The results show that our model outperforms baseline models that directly adopt karma scores as an indicator of informativeness. Furthermore, our model focus on a local feature of a single post, whereas conventional centrality-based models (Erkan and Radev, 2004; Mihalcea and Tarau, 2004) use a global context of posts, and the complementary hybrid of the both models outperform both the centrality-based models and our models.

The contributions of this paper are three fold. 1) Propose a model that harnesses a popularity measure as a distant label for extractive summarization. 2) Create a dataset of online conversations in which the informativeness of contents are annotated to verify that popularity does not correlate with informativeness because of the effect of context. 3) Demonstrate that our model, when combined with a centrality-based model, outperforms baseline models in predicting the informativeness of posts.

2 Related Work

Previous research of summarizing online conversations can be categorized into graph-based methods (Mehdad et al., 2013, 2014; Shang et al., 2018), template-based methods (Oya et al., 2014), and methods which use dialogue acts as a feature (Bhatia et al., 2014; Carenini et al., 2007). In previous research, few or no training data was adopted because of a lack of labeled data. Our model harnesses a vast amount of data from social media.

Many researchers used user-contributed labels from social media as distant labels. Xiong (2014) used review scores on a movie-rating site for a summarization task. For a sentiment analysis on Twitter[3], Davidov(2010) used hashtags, and Guibon (2017) used emoji. In our study, we leverage a popularity measure for a summarization task.

Factor analysis quantifies the contribution of each feature to the result, using a linear model. For example, Suh (2010) analyzed factors contributing to popularity in Twitter. Our model assumes a linear relationship between context and content, and thus enables to utilize the contribution of content as an indicator of informativeness.

3 Data

In this study, we work with Reddit threads. A thread is a set of comments, and the first posted comment is called a submission. Comments can be made in response to submissions as well as comments under the submissions, resulting in a thread being tree-structured. Submissions and comments can be upvoted or downvoted by readers, and karma scores are computed as upvotes minus downvotes. Karma scores follow Zipf's law (Cheng et al., 2017). Therefore, we smooth the karma scores as follows:

$$f(k) = \begin{cases} \log(k+1) & (k \geq 0) \\ 0 & (k < 0) \end{cases}$$

where k represents the karma score.

Reddit is organized into subreddits by topic. Posts from the subreddits AskMen, AskWomen, and AskReddit with 420,598, 247,012, and 644,034 comments, respectively, are collected and split into training and validation sets with a 4:1 ratio. The validation sets are used for early-stopping. All comments were posted from June, 1, 2016 through June, 1, 2017.

Manual Annotation We crowdsource the annotation of comments in terms of informativeness to utilize them as test data. Annotators are asked to choose 3 informative comments from 10. We define 10 comments as a *subset*; each comment is a reply to a submission. For submissions with more than 10 replies, posts with the top 10 karma scores are selected. For each subreddit, 130 subsets were annotated, for a total of 1,300 comments. Because 10 annotators vote for 3 different comments each, the number of votes for a comment ranged from 0 to 10. These numbers we refer to as the *annotated score*. The comments in each subset are shuffled to invalidate the effect of the order in which annotators read the comments.

Liu and Liu (2008) observed that the best summary differs among annotators, especially when summarizing conversations, consequently resulting in low Kappa scores. In their study, the Kappa statistics for six different annotators varied from

[3] https://twitter.com

Table 1: Examples of posts with low karma scores and high annotated scores, and vice versa.

karma score	annotated score	post
0	8	Martin Shkreli was streaming League of legends and my brother messaged him to see if he could get an invite to the group. They have played a few times since.
1	10	Same thing goes for being bitten by a dog, instead of instinctively pulling away...force your arm/hand down their throat. Super effective.
1	10	1 Make sure you have solid internet. 2 Find work from home, they actually exist book keeping, software testing, data entry, etc. 3 Work from home while earning a modest wage you wont get rich on those jobs, but it will certainly pay the bills.
360	0	Ill try this the next time my toddler bites me.
253	0	Im an English major who wants to go into marketing. Wat do
228	0	Sadly people buy the first thing that the see and this is what is in season.

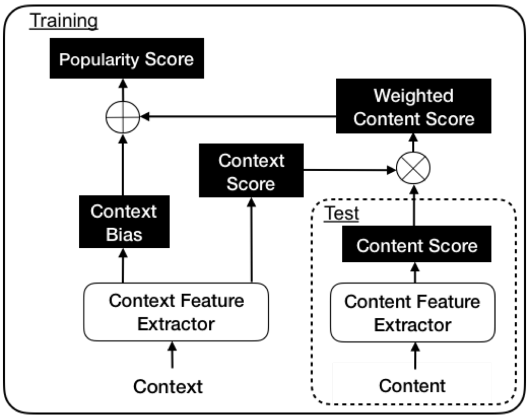

Figure 1: Description of the Disjunctive model. $\bigoplus$ and $\bigotimes$ represent addition and multiplication, respectively. The components of each model we use for training and testing are shown. Black and white squares represent scalars and functions, respectively.

0.11 to 0.35. In our study, the Fleiss Kappa coefficients (Fleiss and Cohen, 1973) of the annotated data are 0.252 for AskMen, 0.191 for AskWomen, and 0.213 for AskReddit.

The correlation coefficients of the karma scores versus the annotated scores are low: 0.063 for AskMen, 0.081 for AskReddit, and 0.107 for AskWomen. Table 1 shows some examples of posts with low karma scores and high annotated scores, and vice versa. It shows that there are informative posts with low karma scores, and non-informative posts with high karma scores. This implies that it is necessary to exclude the effect of context to leverage karma scores as distant labels for summarization.

4 Proposed Model

To exclude the effect of context from the popularity, we propose *Disjunctive model* (Figure 1). This model computes two scalar values, a *content score* and a *context score* from a content feature and a context feature, respectively by multiplying parameter vectors. The model is trained to predict popularity by multiplying the two scores and adding a *context bias*, which is also computed from a context feature. After training, the two scores represent the contribution of the content and context to popularity. We assume that the content score indicates informativeness. While training, the *popularity score* is used to predict the popularity, which is represented by the karma scores in our study. During evaluation, the content score is used for prediction of informativeness. The context score is constrained to be positive; otherwise, it can be either positive or negative, making it difficult to assume that a content score represents informativeness.

4.1 Context Feature Extractor

We use a multi-layer perceptron (MLP) to extract the features of the context of comments. Our study discusses six attributes of context: the karma score of a submission, the karma score of the previous comment, the depth in a thread, the relative time since the previous comment, the rank of the relative time among all replies to a previous comment, and the number of replies to a previous comment. The number of layers is set to 3, and the dimensions of each layer are 64.

4.2 Content Feature Extractor

We use two content extractors: long short-term memory (LSTM) as a basic language model, and a factored neural network (FNN) (Cheng et al., 2017) as a model that achieved state-of-the-art results in karma score prediction tasks. FNN, which is a language model, sequentially predicts the next words in a comment and its reply using an attention mechanism. As in the previous research, we pretrain this model using the same data used in the training, and fine-tune its parameters on the karma

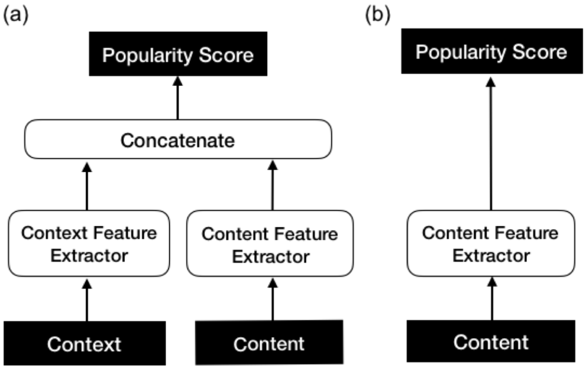
(a)
Popularity Score
Concatenate
Context Feature Extractor
Content Feature Extractor
Context
Content
(b)
Popularity Score
Content Feature Extractor
Content

Figure 2: Description of baseline models. (a) Concat model. (b) Text model.

score prediction task. A single-layer LSTM and FNN are used and the last hidden layers are used as the content feature. The dimensions of the hidden layers are set to 64, and the dimensions of the word embedding are set to 256.

5 Experiments

We train the summarization model using karma scores as distant labels and evaluate the prediction of informativeness with the annotated dataset. As explained in Sec 3, the informativeness of each post is annotated via crowdsourcing and it is difficult to determine the appropriate number of posts needed to create a summary. Therefore, we employ ranking metrics for evaluation. In each subset, where subsets were defined in Sec 3, we rank each comment from 1 to 10 in terms of predicted scores and annotated scores. Ranks of tied scores are set randomly. To avoid randomness from affecting the result, we evaluate 100 times and compute an average as a result. We use three metrics: Spearman's Rho ($S\rho$), precision@3 (prec3), and Mean Reciprocal Rank (MRR) (Mcfee and Lanckriet, 2010).

5.1 Experimental Setting

Experiments are conducted by using mean-squared error as the loss function and Adam as the optimizer (Kingma and Ba, 2014). We replace words that appear fewer than five times with <unk>. There are 63,093 unique terms for AskMen, 53,589 for AskWomen, and 80,426 for AskReddit. The maximum length of each comment is clipped to 50. The mini-batch size is 64.

5.2 Baseline Model

We experiment with four baseline models. Two are supervised models as shown in Figure 2: the Concat model concatenates content and context features, and the Text model uses only content features. The other two are centrality-based unsupervised models: LexRank (Erkan and Radev, 2004) and TextRank (Mihalcea and Tarau, 2004). The unsupervised models only use content features. Disjunctive model use different scores for prediction in the training and in the test, as shown in Figure 1; however, Concat and Text models use a popularity score, the predicted value of a karma score, both in the training and in the test. This is because there is no substitute for the content score in these models.

5.3 Hybrid Model

The supervised models in our study, including the Disjunctive models, compute features from a single post only. To also harness the global information encoded in all posts in a subset, we build a Hybrid model which multiplies the scores from the Disjunctive model and the TextRank.

6 Results

The results of the experiments described in Sec 5 are shown in Table 2. The suffixes Disjunctive, Concat, and Text denote the supervised models described in Sec 4 and Sec 5. The prefixes LSTM and FNN indicate the models we use for content feature extractors. Among the supervised models, our Disjunctive models outperform both LSTM and FNN-based baseline models. In contrast, the results of the Concat models are poor. Unsupervised models which use the global feature of posts in a subset perform well. The FNNDisjunctive model combined with TextRank outperforms both the supervised models and the unsupervised models. To confirm that multiplication performs better in our task, we also experimented with Additive models, which simply add the context score and the content score instead of multiplying. Although better performing than the Text model, the performance was not as good as the Disjunctive model (Not shown in Table).

7 Discussion

Here we discuss the comparison of the results shown in Table 2, and how our Disjunctive model separate the effect of content and context.

Text Model vs Concat Model vs Disjunctive Model The Text model performs worse than our model. A possible reason is that karma scores can

Table 2: Result of ranking annotated scores. The best results among the supervised models are underlined, and the best results among all the models are bolded. The names of our models are also bolded.

Content Type	Model	AskMen			AskReddit			AskWomen		
		Sρ	MRR	prec3	Sρ	MRR	prec3	Sρ	MRR	prec3
Super. Local	LSTMConcat	0.021	0.279	0.297	0.028	0.294	0.323	0.146	0.315	0.367
	FNNConcat	0.032	0.276	0.298	0.068	0.325	0.323	0.129	0.309	0.346
	LSTMText	0.057	0.313	0.337	0.08	0.335	0.342	0.230	0.373	0.413
	FNNText	0.045	0.308	0.318	0.052	0.326	0.320	0.169	0.360	0.393
	LSTMDisjunctive	0.046	0.325	0.327	0.137	0.362	0.384	0.302	0.409	0.452
	FNNDisjunctive	0.164	0.375	0.378	0.196	0.411	0.414	0.259	0.402	0.435
Unsup. Global	TextRank	0.300	0.405	0.417	0.301	0.425	0.435	0.291	0.416	0.422
	LexRank	0.043	0.321	0.352	0.137	0.305	0.384	0.122	0.347	0.383
Hybrid	**TextRank+LSTMDisjuntive**	0.095	0.373	0.358	0.199	0.396	0.422	0.340	**0.438**	**0.468**
	TextRank+FNNDisjuntive	**0.319**	**0.436**	**0.452**	**0.336**	**0.452**	**0.448**	**0.345**	0.437	0.460

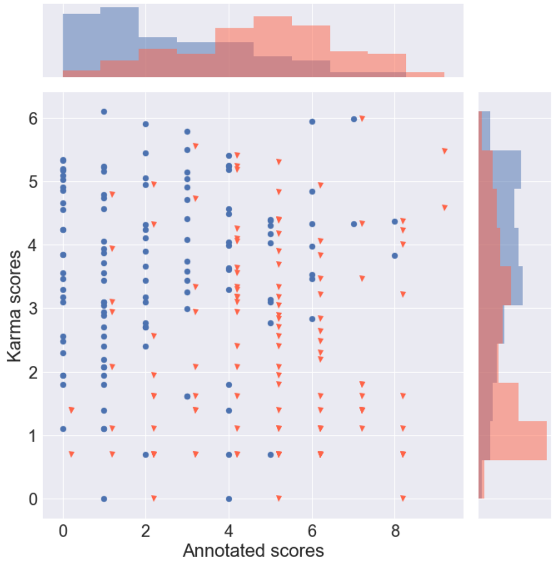

Figure 3: Karma scores and annotated scores of posts extracted by the LSTMConcat model (blue dots on the left) and the LSTMDisjunctive model (red dots shifted to the right for visibility) from the AskWomen dataset. Histograms of the karma scores and the annotated scores are also shown above and on the right.

be different even with similar text content because of different context, and this confuses the models that only use content features. Our model, by contrast, can avoid this problem because it considers the effect of context.

The Text model outperform the Concat model because context is a strong factor in predicting karma scores. If a model can use both content and context (as the Concat model does), it might overfit to context and ignore content. This does not happen in our model because it does not use context in test.

Hybrid Model vs Disjunctive Model The good performance results of the TextRank model indicate that global features of the posts are informative for ranking. While the supervised models just focus on one post at a time in each subset, the unsupervised models look at all the posts together in a subset. The hybrid model of the TextRank and the FNNDisjunctive models takes advantage of the complementary focus of the individual models, and outperforms both the supervised and the unsupervised models.

Separation of Content and Context The visualization in Figure 3 shows that our model can predict informativeness whereas the Concat model cannot. From each subset explained in Sec 3, we extract the post with the highest predicted score by the LSTMConcat and the LSTMDisjunctive. The karma scores and annotated scores of the extracted posts are plotted as blue and red dots, respectively. The karma scores are smoothed by the equation described in Sec 3. There are 130 subsets, for a total of 260 dots plotted. The Concat model extracts posts with low annotated scores but high karma scores, whereas the Disjunctive model extracts posts with high annotated scores regardless of the karma scores.

8 Conclusion

We proposed Disjunctive model that harnesses popularity as distant labels for use in extractive summarization. Our model was shown capable of separating the effects of content and context in a popularity measure and predicting the informativeness of content. To evaluate this, we built a Reddit dataset where informative comments were annotated. Our model, combined with a centrality-based model, outperformed the baseline models on the task of ranking posts to correspond to the rank of the annotated scores in three ranking metrics. Our models currently use only a single post as a feature. In the future, we will develop a model which uses a series of posts as a feature.

References

Sumit Bhatia, Prakhar Biyani, and Prasenjit Mitra. 2014. Summarizing online forum discussions – can dialog acts of individual messages help? In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2127–2131, Doha, Qatar. Association for Computational Linguistics.

Keith Burghardt, Emanuel F. Alsina, Michelle Girvan, William Rand, and Kristina Lerman. 2017. The myopia of crowds: Cognitive load and collective evaluation of answers on stack exchange. *PLOS ONE*, 12(3):e0173610+.

Giuseppe Carenini, Raymond T. Ng, and Xiaodong Zhou. 2007. Summarizing email conversations with clue words. In *Proceedings of the 16th International Conference on World Wide Web*, WWW '07, pages 91–100, New York, NY, USA. ACM.

Hao Cheng, Hao Fang, and Mari Ostendorf. 2017. A factored neural network model for characterizing online discussions in vector space. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2296–2306, Copenhagen, Denmark. Association for Computational Linguistics.

Jianpeng Cheng and Mirella Lapata. 2016. Neural summarization by extracting sentences and words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 484–494. Association for Computational Linguistics.

Dmitry Davidov, Oren Tsur, and Ari Rappoport. 2010. Enhanced sentiment learning using twitter hashtags and smileys. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING '10, pages 241–249, Stroudsburg, PA, USA. Association for Computational Linguistics.

Günes Erkan and Dragomir R. Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *J. Artif. Int. Res.*, 22(1):457–479.

Joseph L. Fleiss and Jacob Cohen. 1973. The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability. *Educational and Psychological Measurement*, 33(3):613–619.

Gael Guibon, Magalie Ochs, and Parice Bellot. 2017. From emojis to sentiment analysis. WACAI 2016, hal-01529708.

Jack Hessel, Lillian Lee, and David Mimno. 2017. Cats and captions vs. creators and the clock: Comparing multimodal content to context in predicting relative popularity. In *Proceedings of the 26th International Conference on World Wide Web*, WWW '17, pages 927–936, Republic and Canton of Geneva, Switzerland. International World Wide Web Conferences Steering Committee.

Masaru Isonuma, Toru Fujino, Junichiro Mori, Yutaka Matsuo, and Ichiro Sakata. 2017. Extractive summarization using multi-task learning with document classification. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2101–2110. Association for Computational Linguistics.

Aaron Jaech, Victoria Zayats, Hao Fang, Mari Ostendorf, and Hannaneh Hajishirzi. 2015. Talking to the crowd: What do people react to in online discussions? In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Feifan Liu and Yang Liu. 2008. Correlation between rouge and human evaluation of extractive meeting summaries. In *Proceedings of ACL-08: HLT, Short Papers*, pages 201–204. Association for Computational Linguistics.

Brian Mcfee and Gert Lanckriet. 2010. Metric learning to rank. In *In Proceedings of the 27th annual International Conference on Machine Learning (ICML*.

Yashar Mehdad, Giuseppe Carenini, and Raymond T. Ng. 2014. Abstractive summarization of spoken and written conversations based on phrasal queries. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1220–1230, Baltimore, Maryland. Association for Computational Linguistics.

Yashar Mehdad, Giuseppe Carenini, Frank Tompa, and Raymond T. NG. 2013. Abstractive meeting summarization with entailment and fusion. In *Proceedings of the 14th European Workshop on Natural Language Generation*, pages 136–146. Association for Computational Linguistics.

R. Mihalcea and P. Tarau. 2004. TextRank: Bringing order into texts. In *Proceedings of EMNLP-04and the 2004 Conference on Empirical Methods in Natural Language Processing*.

Tatsuro Oya, Yashar Mehdad, Giuseppe Carenini, and Raymond Ng. 2014. A template-based abstractive meeting summarization: Leveraging summary and source text relationships. Proceedings of the 8th International Natural Language Generation Conference, pages 45-53.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389. Association for Computational Linguistics.

Guokan Shang, Wensi Ding, Zekun Zhang, Antoine Tixier, Polykarpos Meladianos, Michalis Vazirgiannis, and Jean-Pierre Lorré. 2018. Unsupervised abstractive meeting summarization with multi-sentence compression and budgeted submodular maximization. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 664–674. Association for Computational Linguistics.

Bongwon Suh, Lichan Hong, Peter Pirolli, and Ed H. Chi. 2010. Want to be retweeted? large scale analytics on factors impacting retweet in twitter network. In *Proceedings of the 2010 IEEE Second International Conference on Social Computing*, SOCIALCOM '10, pages 177–184, Washington, DC, USA. IEEE Computer Society.

Wenting Xiong and Diane Litman. 2014. Empirical analysis of exploiting review helpfulness for extractive summarization of online reviews. Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics.