Implicit Feature Detection for Sentiment Analysis

Kim Schouten
schouten@ese.eur.nl

Flavius Frasincar
frasincar@ese.eur.nl

Erasmus University Rotterdam
PO Box 1738, NL-3000 DR
Rotterdam, the Netherlands

ABSTRACT

Implicit feature detection is a promising research direction that has not seen much research yet. Based on previous work, where co-occurrences between notional words and explicit features are used to find implicit features, this research critically reviews its underlying assumptions and proposes a revised algorithm, that directly uses the co-occurrences between implicit features and notional words. The revision is shown to perform better than the original method, but both methods are shown to fail in a more realistic scenario.

Categories and Subject Descriptors

I.2.7 [**Artificial Intelligence**]: Natural Language Processing—*Text analysis*

1. INTRODUCTION

Nowadays, consumer reviews and other opinionated texts that are published on the Web, are frequently mined for sentiment information. Instead of determining sentiment as a general characteristic for a whole document, a fine-grained form of sentiment analysis is now regularly performed, where sentiment scores are computed for each of the aspects or features of the topic or entity being discussed. In this scenario, the detection of features being written about is a crucial step. Most research focuses on the detection of explicit features (i.e., aspects that are explicitly mentioned in the text), as this is both the largest category of features, and, conveniently, also the easiest one to find. This research, however, focuses on the detection of implicit features, which are aspects of the entity being discussed that are not literally mentioned, but are implied by the sentence. An example of such an implicit feature is "size" in the example below.

> "... while light, it will not easily go in small handbags or pockets."

A popular way of finding implicit features is by using Association Rule Mining [2, 4], where each rule is a mapping from an item set A to an item set B. In this case, item set A would be a set of opinion words, and item set B would be a set of feature words. Feature and opinion words are clustered using a distance function that is based on the co-occurrence frequency of these clusters. Instead of using a clustering algorithm to find the implicit features, [5] utilizes the co-occurrence data of notional words and explicit features to find the implicit ones. By knowing which notional words often appear together with a certain explicit feature, the feature, when not explicitly mentioned in the sentence, can be correctly implied based on the fact that the same set of notional words appear in a given sentence.

This research extends [5] by testing two of its assumptions. The first is that explicit features are a good proxy for implicit features. We compare the original method of using the co-occurrence frequencies of notional words and explicit features to find implicit features with a more direct method where co-occurrence frequencies of the implicit features themselves and notional words are utilized. Note that this will transform the unsupervised method into a supervised one. The second assumption is that the algorithm works on sentences with at least one implicit feature. In reality, only a small fraction of the sentences has an implicit feature, hence, the algorithm is also tested in a more real-life environment, where it is not known beforehand whether a sentence contains an implicit feature or not.

Table 1: General statistics of the data set.

# implicit features	none	1	2
all sentences	3797	140	8
sentences w/o explicit features	2726	119	5

2. ORIGINAL METHOD

As mentioned above, [5] finds implicit features by linking explicit features to a set of notional words which often appear together. Then, a feature can be implied when these notional words are found. This is done by assigning a score $T(f_i)$ to each feature f_i:

$$T(f_i) = \frac{1}{v}\sum_{j=1}^{v}\frac{P(f_i|w_j)}{v}, \quad (1)$$

where v is the number of notional words in a sentence, f_i is the ith feature for which the $T(f_i)$ score is computed, and w_j is the jth notional word in the sentence. Thus, the score $T(f_i)$ is the average fraction of co-occurrences of feature f_i

WWW'14 Companion, April 7–11, 2014, Seoul, Korea.
ACM 978-1-4503-2745-9/14/04.
http://dx.doi.org/10.1145/2567948.2577378.

Table 2: Evaluation results using F_1-measure.

Method	Subset (sent. w/ impl. feat.)	All data (no threshold)	All data (with threshold)
Original	15.1%	1.2%	1.3%
Revised	28.5%	2.7%	8.0%

with all notional words in the sentence, and the higher this value, the more likely it is that f_i is implied by the notional words in the sentence. The feature with the highest T-score is determined to be the implicit feature in this sentence.

3. REVISED METHOD

Since a T-score will only be computed for explicit features that have been found in the text, it is assumed that any implicit feature in a review, will also occur as an explicit feature. Furthermore, because of co-occurrences between an explicit feature and the notional words (i.e., its context) are used to find the implicit version of that feature, it also assumed that implicit features have the same context in which they appear as explicit features. Using the co-occurrences between notional words and implicit features directly, will mitigate these concerns.

The other issue that will be addressed is the fact that while the algorithm was trained (e.g., computing the co-occurrence matrix between notional words and features) on sentences with explicit features, only sentences with implicit features but without any explicit features are present in the test set. This obviously makes the task much easier, and it stands to reason to also gauge the performance of the method when it is not known beforehand whether there is any implicit feature in a sentence at all. Since the original method always chooses the best alternative for each sentence, a threshold parameter is now necessary to give the algorithm the option of choosing no feature at all. Still, just like the original method, it can not choose two implicit features for one sentence, so sentences with two annotated implicit features are not used for testing.

4. EVALUATION

For the evaluation of the proposed extensions, the well-known data set of [3] is used. It contains consumer reviews from Amazon.com about five different electronic products. Both explicit and implicit features are annotated. Some general statistics are given in Table 1. All evaluations are performed using 10-fold cross-validation.

The evaluation results are shown in Table 2. The first column represents the same evaluation conditions as in [5]: only sentences that are known to have exactly one implicit feature are used for testing. The second column gives the performance of both methods when this assumption is dropped. Since the method will always return exactly one implicit feature for each sentence, even when no feature is actually present, performance drops significantly. Adding the discussed threshold to both algorithms, while not particularly helpful for the original method, yields an F_1-score that is about thrice the score of not using a threshold.

Clearly, the direct use of co-occurrence data between notional words and implicit features gives much better performance than when using explicit features as an intermediate proxy. However, when the assumption of always having at least one implicit feature per sentence is dropped, performance declines to very low levels. Adding a threshold, while particularly beneficial for the revised method, does not yield the desired performance. Hence, the algorithm in its current form is not suitable for this later task.

5. CONCLUDING REMARKS

This research has two main contributions. The first contribution is that using co-occurrence data between notional words and implicit features to find implicit features gives better results than using co-occurrence data between notional words and explicit features to find implicit features. This comes, however, at the cost of transforming the originally unsupervised algorithm into a supervised one, since implicit feature annotations are now required to correctly build the co-occurrence matrix. The second contribution is that the algorithm is shown to be unsuitable to deal with sentences that contain no implicit feature. Adding a threshold, while improving the results, is not enough to render this algorithm useful in practice.

As future work, it would be interesting to test the impact of the various word categories (i.e., Parts-of-Speech). For example, the co-occurrence between an implicit feature and nouns might be more useful than the co-occurrence with an adverb. Also interesting could be to perform a word sense disambiguation step and count co-occurrences between implicit features and WordNet [1] synsets instead of words. Another avenue for future research could be to add a classifier that pre-processes all sentences first to predict the existence of an implicit feature in a sentence.

Acknowledgment

The authors are partially supported by the Dutch national program COMMIT. We would also like to thank the following students for their work: Sven van den Berg, Gino Mangnoesing, Andrew Hagens, Marnix Moerland, Lotte Snoek, Marijn Waltman, Onne van der Weijde, and Arno de Wolf.

6. REFERENCES

[1] C. Fellbaum. *WordNet: An Electronic Lexical Database.* MIT Press, 1998.

[2] Z. Hai, K. Chang, and J. Kim. Implicit Feature Identification via Co-occurrence Association Rule Mining. In *Proceedings of the 12th International Conference on Computational Linguistics and Intelligent Text processing (CICLing 2011)*, volume 6608, pages 393–404. Springer, 2011.

[3] M. Hu and B. Liu. Mining Opinion Features in Customer Reviews. In *Proceedings of the 19th National Conference on Artificial Intelligence (AAAI 2004)*, pages 755–760. AAAI, 2004.

[4] Q. Su, X. Xu, H. Guo, Z. Guo, X. Wu, X. Zhang, B. Swen, and Z. Su. Hidden Sentiment Association in Chinese Web Opinion Mining. In *Proceedings of the 17th International Conference on World Wide Web (WWW 2008)*, pages 959–968. ACM, 2008.

[5] Y. Zhang and W. Zhu. Extracting Implicit Features in Online Customer Reviews for Opinion Mining. In *Proceedings of the 22nd International Conference on World Wide Web (WWW 2013)*, pages 103–104. ACM, 2013.