Emotive or Non-emotive: That is The Question

Michal Ptaszynski **Fumito Masui** **Rafal Rzepka** **Kenji Araki**
Department of Computer Science, Kitami Institute of Technology
{ptaszynski,f-masui}@cs.kitami-it.ac.jp
Graduate School of Information Science and Technology, Hokkaido University
{rzepka,araki}@ist.hokudai.ac.jp

Abstract

In this research we focus on discriminating between emotive (emotionally loaded) and non-emotive sentences. We define the problem from a linguistic point of view assuming that emotive sentences stand out both lexically and grammatically. We verify this assumption experimentally by comparing two sets of such sentences in Japanese. The comparison is based on words, longer n-grams as well as more sophisticated patterns. In the classification we use a novel unsupervised learning algorithm based on the idea of language combinatorics. The method reached results comparable to the state of the art, while the fact that it is fully automatic makes it more efficient and language independent.

1 Introduction

Recently the field of sentiment analysis has attracted great interest. It has become popular to try different methods to distinguish between sentences loaded with positive and negative sentiments. However, a few research focused on a task more generic, namely, discriminating whether a sentence is even loaded with emotional content or not. The difficulty of the task is indicated by three facts. Firstly, the task has not been widely undertaken. Secondly, in research which addresses the challenge, the definition of the task is usually based on subjective ad hoc assumptions. Thirdly, in research which do tackle the problem in a systematic way, the results are usually unsatisfactory, and satisfactory results can be obtained only with large workload.

We decided to tackle the problem in a standardized and systematic way. We defined emotionally loaded sentences as those which in linguistics are described as fulfilling the emotive function of language. We assumed that there are repetitive patterns which appear uniquely in emotive sentences. We performed experiments using a novel unsupervised clustering algorithm based on the idea of language combinatorics. By using this method we were also able to minimize human effort and achieve F-score comparable to the state of the art with much higher Recall rate.

The outline of the paper is as follows. We present the background for this research in Section 2. Section 3 describes the language combinatorics approach which we used to compare emotive and non-emotive sentences. In section 4 we describe our dataset and experiment settings. The results of the experiment are presented in Section 5. Finally the paper is concluded in Section 6.

2 Background

There are different linguistic means used to inform interlocutors of emotional states in an everyday communication. The emotive meaning is conveyed verbally and lexically through exclamations (Beijer, 2002; Ono, 2002), hypocoristics (endearments) (Kamei et al., 1996), vulgarities (Crystal, 1989) or, for example in Japanese, through mimetic expressions (*gitaigo*) (Baba, 2003). The function of language realized by such elements of language conveying emotive meaning is called the **emotive function of language**. It was first distinguished by Bühler (1934-1990) in his *Sprachtheorie* as one of three basic functions of language[1]. Bühler's theory was picked up later by Jakobson (1960), who by distinguishing three other functions laid the grounds for structural linguistics and communication studies.

2.1 Previous Research

Detecting whether sentences are loaded with emotional content has been undertaken by a number

[1]The other two being *descriptive* and *impressive*.

Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, pages 59–65,
Baltimore, Maryland, USA. June 27, 2014.

of researchers, most often as an additional task in either sentiment analysis (SA) or affect analysis (AA). SA, in great simplification, focuses on determining whether a language entity (sentence, document) was written with positive or negative attitude toward its topic. AA on the other hand focuses on specifying which exactly emotion type (joy, anger, etc.) has been conveyed. The fact, that the task was usually undertaken as a subtask, influences the way it was formulated. Below we present some of the most influential works on the topic, but formulating it in slightly different terms.

Emotional vs. Neutral: Discriminating whether a sentence is emotional or neutral is to answer the question of whether it can be interpreted as produced in an emotional state. This way the task was studied by Minato et al. (2006), Aman and Szpakowicz (2007) or Neviarouskaya et al. (2011).

Subjective vs. Objective: Discriminating between subjective and objective sentences is to say whether the speaker presented the sentence contents from a first-person-centric perspective or from no specific perspective. The research formulating the problem this way include e.g, Wiebe et al. (1999), who classified subjectivity of sentences using naive Bayes classifier, or later Wilson and Wiebe (2005). In other research Yu and Hatzivassiloglou (2003) used supervised learning to detect subjectivity and Hatzivassiloglou and Wiebe (2012) studied the effect of gradable adjectives on sentence subjectivity.

Emotive vs. Non-emotive: Saying that a sentence is emotive means to specify the linguistic features of language which where used to produce a sentence uttered with emphasis. Research that formulated and tackled the problem this way was done by, e.g., Ptaszynski et al. (2009).

Each of the above nomenclature implies similar, though slightly different assumptions. For example, a sentence produced without any emotive characteristics (non-emotive) could still imply emotional state in some situations. Also Bing and Zhang (2012) notice that "not all subjective sentences express opinions and those that do are a subgroup of opinionated sentences." A comparison of the scopes and overlaps of different nomenclature is represented in Figure 1. In this research we formulate the problem similarly to Ptaszynski et al. (2009), therefore we used their system to compare with our method.

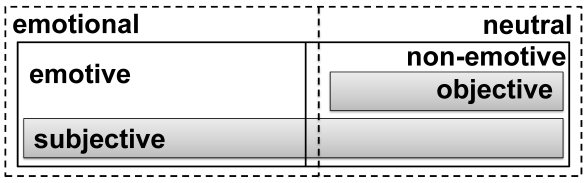


Figure 1: Comparison of between different nomenclature used in sentiment analysis research.

3 Language Combinatorics

The idea of language combinatorics (LC) assumes that patterns with disjoint elements provide better results than the usual bag-of-words or n-gram approach (Ptaszynski et al., 2011). Such patterns are defined as ordered non-repeated combinations of sentence elements. They are automatically extracted by generating all ordered combinations of sentence elements and verifying their occurrences within a corpus.

In particular, in every n-element sentence there is k-number of combination clusters, such as that $1 \leq k \leq n$, where k represents all k-element combinations being a subset of n. The number of combinations generated for one k-element cluster of combinations is equal to binomial coefficient, like in eq. 1. Thus the number of all possible combinations generated for all values of k from the range of $\{1, ..., n\}$ is equal to the sum of all combinations from all k-element clusters, like in eq. 2.

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} \tag{1}$$

$$\sum_{k=1}^{n} \binom{n}{k} = \frac{n!}{1!(n-1)!} + \frac{n!}{2!(n-2)!} + ... + \frac{n!}{n!(n-n)!} = 2^n - 1 \tag{2}$$

One problem with combinatorial approach is the phenomenon of exponential and rapid growth of function values during combinatorial manipulations, called combinatorial explosion (Krippendorff, 1986). Since this phenomenon causes long processing time, combinatorial approaches have been often disregarded. We assumed however, that it could be dealt with when the algorithm is optimized to the requirements of the task. In preliminary experiments Ptaszynski et al. (2011) used a generic sentence pattern extraction architecture SPEC to compare the amounts of generated sophisticated patterns with n-grams, and noticed that it is not necessary to generate patterns of all lengths, since the most useful ones usually appear in the group of 2 to 5 element patterns. Following their experience we limit the pattern length in our research to 6 elements. All non-subsequent el-

Table 1: Some examples from the dataset representing emotive and non-emotive sentences close in content, but differing in emotional load expressed in the sentence (Romanized Japanese / Translation).

emotive	**non-emotive**
Takasugiru kara ne / 'Cause its just too expensive	*Kōgaku na tame desu.* / Due to high cost.
Un, umai, kangeki da. / Oh, so delicious, I'm impressed.	*Kono karē wa karai.* / This curry is hot.
Nanto ano hito, kekkon suru rashii yo! / Have you heard? She's getting married!	*Ano hito ga kekkon suru rashii desu.* / They say she is gatting married.
Chō ha ga itee / Oh, how my tooth aches!	*Ha ga itai* / A tooth aches
Sugoku kirei na umi da naaa / Oh, what a beautiful sea!	*Kirei na umi desu* / This is a beautiful sea

ements are also separated with an asterisk ("*") to mark disjoint elements.

The weight w_j of each pattern generated this way is calculated, according to equation 3, as a ratio of all occurrences of a pattern in one corpus O_{pos} to the sum of occurrences in two compared corpora $O_{pos}+O_{neg}$. The weights are also normalized to fit in range from +1 (representing purely emotive patterns) to -1 (representing purely non-emotive patterns). The normalization is achieved by subtracting 0.5 from the initial score and multiplying this intermediate product by 2. The score of one sentence is calculated as a sum of weights of patterns found in the sentence, like in eq. 4.

$$w_j = \left(\frac{O_{pos}}{O_{pos} + O_{neg}} - 0.5\right) * 2 \quad (3)$$

$$score = \sum w_j, (1 \geq w_j \geq -1) \quad (4)$$

The weight can be further modified by either

- awarding length k, or
- awarding length k and occurrence O.

The list of generated frequent patterns can also be further modified. When two collections of sentences of opposite features (such as "emotive vs. non-emotive") are compared, a generated list will contain patterns appearing uniquely on only one of the sides (e.g. uniquely emotive patterns and uniquely non-emotive patterns) or in both (ambiguous patterns). Therefore the pattern list can be modified by deleting

- all ambiguous patterns, or
- only ambiguous patterns appearing in the same number on both sides (later called "zero patterns", since their weight is equal 0).

Moreover, since a list of patterns will contain both the sophisticated patterns as well usual n-grams, the experiments were performed separately for all patterns and n-grams only. Also, if the initial collection was biased toward one of the sides (sentences of one kind were longer or more numerous), there will be more patterns of a certain sort. To mitigate this bias, instead of applying a rule of thumb, the threshold was optimized automatically.

4 Experiments

4.1 Dataset Preparation

In the experiments we used a dataset developed by Ptaszynski et al. (2009) for the needs of evaluating their affect analysis system ML-Ask for Japanese language. The dataset contains 50 emotive and 41 non-emotive sentences. It was created as follows.

Thirty people of different age and social groups participated in an anonymous survey. Each participant was to imagine or remember a conversation with any person they know and write three sentences from that conversation: one free, one emotive, and one non-emotive. Additionally, the participants were asked to make the emotive and non-emotive sentences as close in content as possible, so the only difference was whether a sentence was loaded with emotion or not. The participants also annotated on their own free utterances whether or not they were emotive. Some examples from the dataset are represented in Table 1.

In our research the above dataset was further preprocessed to make the sentences separable into elements. We did this in three ways to check how the preprocessing influences the results. We used MeCab[2], a morphological analyzer for Japanese to preprocess the sentences from the dataset in the three following ways:

- **Tokenization:** All words, punctuation marks, etc. are separated by spaces.
- **Parts of speech (POS):** Words are replaced with their representative parts of speech.
- **Tokens with POS:** Both words and POS information is included in one element.

The examples of preprocessing are represented in Table 2. In theory, the more generalized a sentence is, the less unique patterns it will produce, but the produced patterns will be more frequent. This can be explained by comparing tokenized sentence with its POS representation. For example, in the sentence from Table 2 we can see that a simple phrase *kimochi ii* ("feeling good") can be

[2]https://code.google.com/p/mecab/

Table 2: Three kinds of preprocessing of a sentence in Japanese; N = noun, TOP = topic marker, ADV = adverbial particle, ADJ = adjective, COP = copula, EXCL = exclamation mark.

Sentence: 今日はなんて気持ちいい日なんだ！ **Transliteration:** *Kyōwanantekimochiiihinanda!* **Glossing:** Today TOP what pleasant day COP EXCL **Translation:** What a pleasant day it is today!
Preprocessing examples
1. Words: *Kyō wa nante kimochi ii hi nanda !* **2. POS:** `N TOP ADV N ADJ N COP EXCL` **3.Words+POS:** *Kyō*`[N]` *wa*`[TOP]` *nante*`[ADV]` *kimochi*`[N]` *ii*`[ADJ]` *hi*`[N]` *nanda*`[COP]` *!*`[EXCL]`

represented by a POS pattern `N ADJ`. We can easily assume that there will be more `N ADJ` patterns than *kimochi ii*, because many word combinations can be represented as `N ADJ`. Therefore POS patterns will come in less variety but with higher occurrence frequency. By comparing the result of classification using different preprocessing methods we can find out whether it is better to represent sentences as more generalized or as more specific.

4.2 Experiment Setup

The experiment was performed three times, once for each kind of preprocessing. Each time 10-fold cross validation was performed and the results were calculated using Precision (P), Recall (R) and balanced F-score (F) for each threshold. We verified which version of the algorithm achieves the top score within the threshold span. However, an algorithm could achieve the best score for one certain threshold, while for others it could perform poorly. Therefore we also looked at which version achieves high scores for the longest threshold span. This shows which algorithm is more balanced. Finally, we checked the statistical significance of the results. We used paired t-test because the classification results could represent only one of two classes (emotive or non-emotive). We also compared the performance to the state of the art, namely the affect analysis system ML-Ask developed by Ptaszynski et al. (2009).

5 Results and Discussion

The overall F-score results were generally the best for the datasets containing in order: both tokens and POS, tokens only and POS only. The F-scores for POS-preprocessed sentences revealed the least constancy. For many cases n-grams scored higher than all patterns, but almost none of the results reached statistical significance. The F-score results for the tokenized dataset were also not unequivocal. For higher thresholds patterns scored higher, while for lower thresholds the results were similar. The scores were rarely significant, utmost at 5% level ($p<0.05$), however, in all situations where n-grams visibly scored higher, the differences were not statistically significant. Finally, for the preprocessing including both tokens and POS information, pattern-based approach achieved significantly better results (p-value <0.01 or <0.001). The algorithm reached its plateau at F-score around 0.73–0.74 for tokens and POS separately, and 0.75–0.76 for tokens with POS together. In the POS dataset the elements were more abstracted, while in token-POS dataset the elements were more specific, producing a larger number, but less frequent patterns. Lower scores for POS dataset could suggest that the algorithm works better with less abstracted preprocessing. Examples of F-score comparison between n-grams and patterns for tokenized and token-POS datasets are represented in Figures 2 and 3, respectively.

Table 3: Best results for each version of the method compared with the ML-Ask system.

	ML-Ask	SPEC					
		tokenized		POS		token-POS	
		n-grams	patterns	n-grams	patterns	n-grams	patterns
Precision	0.80	0.61	0.6	0.68	0.59	0.65	0.64
Recall	0.78	1.00	0.96	0.88	1.00	0.95	0.95
F-score	0.79	0.75	0.74	0.77	0.74	0.77	0.76

Results for Precision showed similar tendencies. They were the most ambiguous for POS preprocessing. For the tokenized dataset, although there always was one or two thresholds for which n-grams scored higher, scores for patterns were more balanced, starting with a high score and decreasing slowly. As for the token-POS preprocessing patterns achieved higher Precision for most of the threshold span. The highest Precision of all was achieved in this dataset by patterns with P = 0.87 for R = 0.50.

As for Recall, the scores were consistent for all kinds of preprocessing, with higher scores for patterns within most of the threshold span and equaling while the threshold decreases. The highest scores achieved for each preprocessing for n-grams and patterns are represented in Table 3.

The affect analysis system ML-Ask (Ptaszynski et al., 2009) on the same dataset reached F = 0.79, P = 0.8 and R = 0.78. The results were generally

comparable, however slightly higher for ML-Ask when it comes to P and F-score. R was always better for the proposed method. However, ML-Ask is a system requiring handcrafted lexicons, while our method is fully automatic, learning the patterns from data, not needing any particular preparations, which makes it more efficient.

5.1 Detailed Analysis of Learned Patterns

Within some of the most frequently appearing emotive patterns there were for example: *!* (exclamation mark), *n*yo*, *cha* (emotive verb modification), *yo* (exclamative sentence ending particle), *ga*yo*, *n*!* or *naa* (interjection). Some examples of sentences containing those patterns are below (patterns underlined). Interestingly, most elements of those patterns appear in ML-Ask handcrafted databases, which suggests it could be possible to improve ML-Ask performance by extracting additional patterns with SPEC.

Ex. 1. *Megane, soko ni atta nda yo.* (The glasses were over there!)
Ex. 2. *Uuun, butai ga mienai yo.* (Ohh, I cannot see the stage!)
Ex. 3. *Aaa, onaka ga suita yo.* (Ohh, I'm so hungry)

Another advantage of our method is the fact that it can mark both emotive and non-emotive elements in sentence, while ML-Ask is designed to annotate only emotive elements. Some examples of extracted non-emotive patterns were for example: *desu, wa*desu, mashi ta*, or *te*masu*. All of them were patterns described in linguistic literature as typically non-emotive, consisting in copulas (*desu*), verb endings (*masu, mashi ta*). Some sentence examples with those patterns include:

Ex. 4. *Kōgaku na tame desu.* (Due to high cost.)
Ex. 5. *Kirei na umi desu* (This is a beautiful sea)
Ex. 6. *Kyo wa yuki ga futte imasu.* (It is snowing today.)

6 Conclusions and Future Work

We presented a method for automatic extraction of patterns from emotive sentences. We assumed emotive sentences are distinguishable both lexically and grammatically and performed experiments to verify this assumption. In the experiments we used a set of emotive and non-emotive sentences preprocessed in different ways (tokens, POS, token-POS) The patterns extracted from sentences were applied to recognize emotionally loaded sentences.

The algorithm reached its plateau for F-score around 0.75–0.76 for patterns containing both tokens and POS information. Precision for patterns was balanced, while for n-grams, although occasionally achieving high scores, it was quickly decreasing. Recall scores were almost always better for patterns. The generally lower results for POS-represented sentences suggest that the algorithm works better with less abstracted elements.

The results of the proposed method and the affect analysis system ML-Ask were comparable. ML-Ask achieved better Precision, but lower Recall. However, our method is more efficient as it does not require handcrafted lexicons. Moreover, automatically extracted patterns overlap with handcrafted databases of ML-Ask, which suggests it could be possible to improve ML-Ask performance with our method. In the near future we plan to perform experiments on larger datasets, also in other languages, such as English or Chinese.

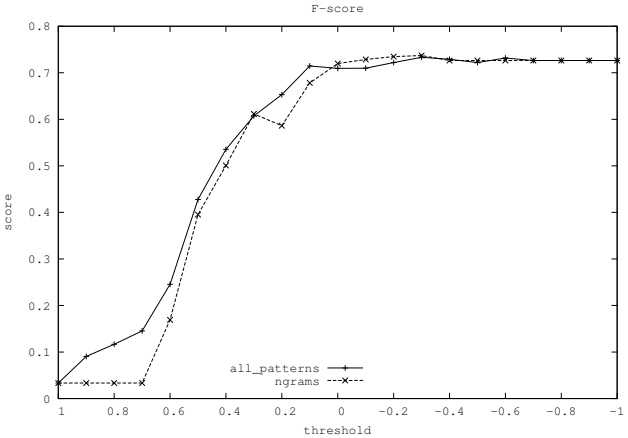


Figure 2: F-score comparison between n-grams and patterns for tokenized detaset (p = 0.0209).

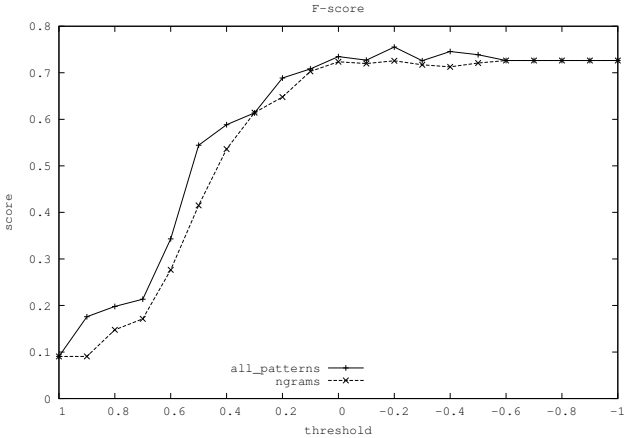


Figure 3: F-score comparison for n-grams and patterns for dataset with tokens and POS (p = 0.001).

References

Saima Aman and Stan Szpakowicz. 2007. Identifying expressions of emotion in text. In *Proceedings of the 10th International Conference on Text, Speech, and Dialogue (TSD-2007)*, Lecture Notes in Computer Science (LNCS), Springer-Verlag.

Junko Baba. 2003. Pragmatic function of Japanese mimetics in the spoken discourse of varying emotive intensity levels. *Journal of Pragmatics*, Vol. 35, No. 12, pp. 1861-1889, Elsevier.

Fabian Beijer. 2002. The syntax and pragmatics of exclamations and other expressive/emotional utterances. *Working Papers in Linguistics 2*, The Dept. of English in Lund.

Bing Liu, Lei Zhang. 2012. A survey of opinion mining and sentiment analysis. In *Mining Text Data*, pp. 415-463. Springer.

Karl Bühler. 1990. *Theory of Language. Representational Function of Language*. John Benjamins Publ. (reprint from Karl Bühler. *Sprachtheorie. Die Darstellungsfunktion der Sprache*, Ullstein, Frankfurt a. M., Berlin, Wien, 1934.)

David Crystal. 1989. *The Cambridge Encyclopedia of Language*. Cambridge University Press.

Vasileios Hatzivassiloglou and Janice Wiebe. Effects of adjective orientation and gradability on sentence subjectivity. In *Proceedings of International Conference on Computational Linguistics (COLING-2000)*, pp. 299-305, 2000.

Roman Jakobson. 1960. Closing Statement: Linguistics and Poetics. *Style in Language*, pp.350-377, The MIT Press.

Takashi Kamei, Rokuro Kouno and Eiichi Chino (eds.). 1996. *The Sanseido Encyclopedia of Linguistics*, Vol. VI, Sanseido.

Klaus Krippendorff. 1986. Combinatorial Explosion, In: Web Dictionary of Cybernetics and Systems. Princia Cybernetica Web.

Junko Minato, David B. Bracewell, Fuji Ren and Shingo Kuroiwa. 2006. Statistical Analysis of a Japanese Emotion Corpus for Natural Language Processing. *LNCS* 4114, pp. 924-929.

Alena Neviarouskaya, Helmut Prendinger and Mitsuru Ishizuka. 2011. Affect analysis model: novel rule-based approach to affect sensing from text. *Natural Language Engineering*, Vol. 17, No. 1 (2011), pp. 95-135.

Hajime Ono. 2002. *An emphatic particle DA and exclamatory sentences in Japanese*. University of California, Irvine.

Christopher Potts and Florian Schwarz. 2008. Exclamatives and heightened emotion: Extracting pragmatic generalizations from large corpora. Ms., UMass Amherst.

Michal Ptaszynski, Pawel Dybala, Rafal Rzepka and Kenji Araki. 2009. Affecting Corpora: Experiments with Automatic Affect Annotation System - A Case Study of the *2channel* Forum -, In *Proceedings of The Conference of the Pacific Association for Computational Linguistics (PACLING-09)*, pp. 223-228.

Michal Ptaszynski, Rafal Rzepka, Kenji Araki and Yoshio Momouchi. 2011. Language combinatorics: A sentence pattern extraction architecture based on combinatorial explosion. *International Journal of Computational Linguistics (IJCL)*, Vol. 2, Issue 1, pp. 24-36.

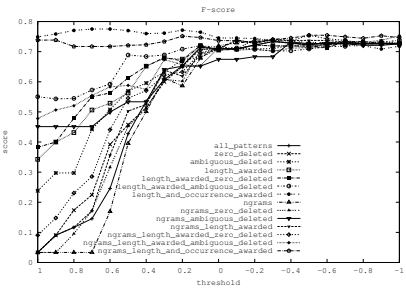
Kaori Sasai. 2006. The Structure of Modern Japanese Exclamatory Sentences: On the Structure of the *Nanto*-Type Sentence. *Studies in the Japanese Language*, Vol, 2, No. 1, pp. 16-31.

Janyce M. Wiebe, Rebecca F. Bruce and Thomas P. O'Hara. 1999. Development and use of a gold-standard data set for subjectivity classifications. In Proceedings of the Association for Computational Linguistics (ACL-1999), pp. 246-253, 1999.

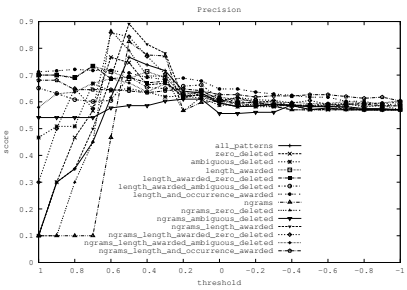
Theresa Wilson and Janyce Wiebe. 2005. Annotating Attributions and Private States. *Proceedings of the ACL Workshop on Frontiers in Corpus Annotation II*, pp. 53-60.

Hong Yu and Vasileios Hatzivassiloglou. 2003. Towards answering opinion questions: separating facts from opinions and identifying the polarity of opinion sentences. In *Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP-2003)*, pp. 129-136, 2003.

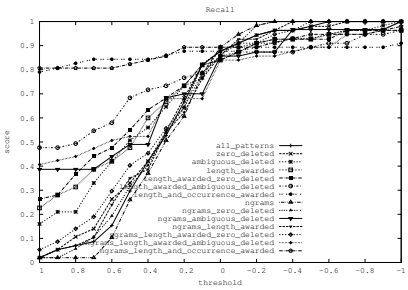
Appendix: Comparison of experiment results in all experiment settings for all three ways of dataset preprocessing.



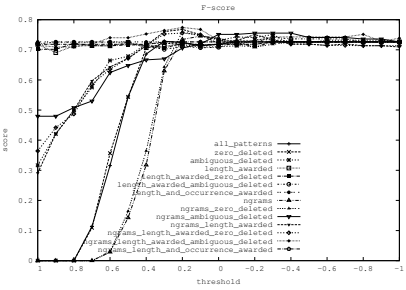
(a) F-score comparison for tokenized dataset.



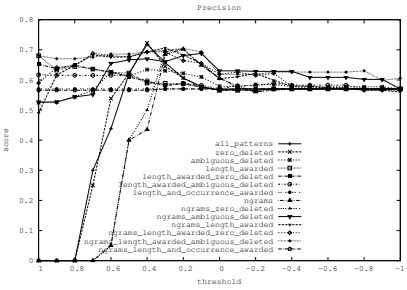
(b) Precision comparison for tokenized dataset.



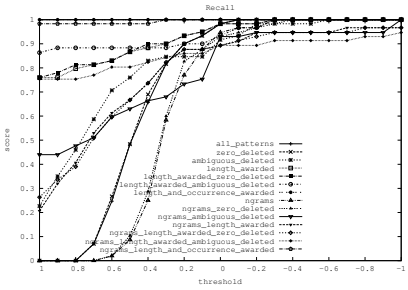
(c) Recall comparison for tokenized dataset.



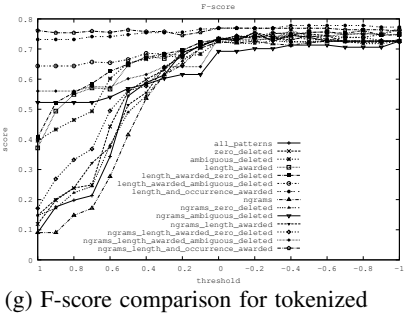
(d) F-score comparison for POS-tagged dataset.



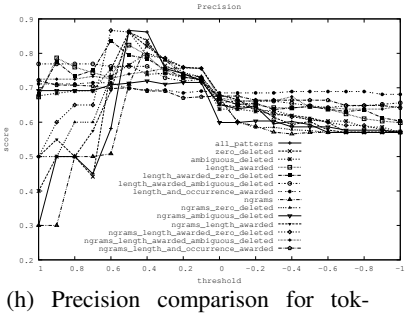
(e) Precision comparison for POS-tagged dataset.



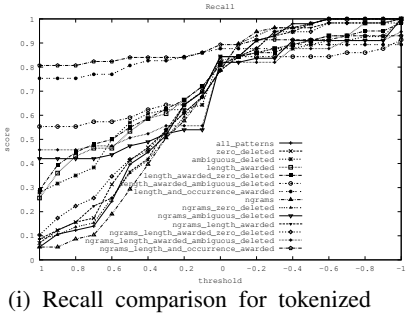
(f) Recall comparison for POS-tagged dataset.



(g) F-score comparison for tokenized dataset with POS tags.



(h) Precision comparison for tokenized dataset with POS tags.



(i) Recall comparison for tokenized dataset with POS tags.