Text-Mining on Incident Reports to Find Knowledge on Industrial Safety

Toru Nakata, PhD, National Institute of Advanced Industrial Science and Technology

Key Words: text mining, accident report, event recognition

SUMMARY & CONCLUSIONS

To prevent accidents, it is very important to learn why and how past accidents occurred and escalated. The information of accidents is mostly recorded in natural language texts, which is not convenient to analyze the flow of events in the accidents. This paper proposes a method to recognize typical flow of events in a large set of text reports. By focusing two adjacent sentences, our system succeeded to detect typical pairs of predecessor word and successor word. Then we can recognize the typical flows of accidents.

1 INTRODUCTION

Learning from past accidents is very important to prevent accident in the future. That is why many societies concerning industrial safety are collecting records of past accidents [1].

Like ordinary traffic accidents, industrial accidents are easily repeated. Although some accidents can occur as unprecedented and unexpected pattern, most cases of them have parts similar to typical pattern of past accidents. Even unheard-of accidents may have partial similarity to common accidents. Those who know well about past accidents can notice risks for future accidents.

To learn past accidents, we usually rely on text information, such as newswires or official reports edited by expert investigation committees. The best data shapes to understand accidents are raw and vivid data like videos or evidential objects of the accident scene. Such data are not convenient to preserve and to collect. So we usually write text reports with summarizing the accidents, and we abandon the raw data.

For certain purposes, text is a convenient shape of information. By reading accident reports, we can understand the story of the accident deeply. But text is costly shape for statistical processing. For instance, it requires careful consideration to find similarities and differences between different accidents by reading their text information. We have to spend long time for it.

There is huge quantity of text information reporting industrial accidents in the world, and the amount is increasing. Human beings cannot read all of them any longer, so the most of texts are left unread and unused.

Natural language processing (NLP) technology can be utilized for the task of reading and understanding of such huge text information. We can use NLP to detect similarities and differences among accident and to clarify causality of events in the accidents. Such analysis will help us to prevent future accidents.

This paper proposes a NLP method that can process huge amount of incident reports to understand typical patterns of progress of incidents.

2 POINTS TO UNDERSTAND INDUSTIRAL INCIDENTS

2.1 Causality

We use the term of ‘incident’ or ‘accident’ for harmful and unpleasant event. An incident is a series of events, which have causality among them and end with bad result.

Finding causality is crucial to prevent similar incidents, because we cannot hinder occurrence of the bad event without knowing its cause events.

Also, analysis of causality is important to find similarities or differences among incidents. Different incidents may have partially common flows of events. We should detect common patterns of incidents without being deluded by minor differences.

Safety engineering employs graph methods to analyze causality. Traditional methods represented by Event Tree Analysis (ETA), Fault Tree Analysis (FTA), and HAZOP are used to generate graphs of causality among events in an incident [2].

It usually requires deep consideration of experts to composing event causality graphs, so we cannot generate the graphs easily. Automation of the graph composition is strongly required.

Even though automatic detection of events and causal relationship described in texts is one of most active topics in natural language processing, its particular difficulties are becoming clear.

First of all, the definition of causal relationship is difficult [3]. While we usually regard causal relationship such as “A person will die if he eats potassium cyanide,” we do not accept awkward (but logically true) relationships like “Churchill was not born if Cleopatra was not born.”

Moreover, there is a problem of false correlation like “Mr. A's age will go up by two if age of Mr. B goes up by two.” Fake correlations cannot be eliminated by observing only their correlation coefficient.

Deep consideration on meaning of texts with common sense is required for detection of causal relationship.

2.2 Counting Number of Events

We have to consider about what is 'one' event. In general, an event is composed with a sequence of several micro-events, and any event can be regarded as a part of the larger event. There is no universal and objective way to count the number of event.

In natural language processing, an easy way of event counting, which regards one verb as a sign of one event, is often employed. If we employ this way, count of events depends on sense of the writer of the text.

Also other methods depend on subjectivity of the author of the text.

3 PROPOSED METHOD

3.1 Simplification for Expedient

In this paper, we propose an expedient method to handle events described in texts.

Our method considers causality less strictly: it regards a certain relationship among terms as causality, if the text data contain enough number of sequences that contains term A followed by term B with small interval. Our method considers an event related to term A as a candidate of a cause of another event related to term B.

This method may misunderstand fake causality as true. This fault can be eliminated to some extent when the amount of data is enough large. True causality can be observed steadily, while fortuitous sequences of events will be ignored by the law of large numbers.

Regarding way to count the number of events, we assume that one sentence describes one event that has proper size for causality analysis of the incident. The validity of this assumption will be verified in the following experiment.

3.2 Scope of Two Adjacent Sentences

Our method works with focusing 2 adjacent sentences in the source text to detect the causality among terms.

Assume a text report of an incident consist of N sentences. Our system puts a focusing scope on 2 sentences of i-th, and $i+1$th one. Here we call such scope bag-of-word (BoW).

For example, an incident report contains a sequence of sentences shown in Figure 1. Our system will observe the contents of the text within the BoW scopes indicated with red braces.

Our system counts co-occurrence of meaningful words in this BoW. Our system accepts noun, proper noun, verb and adjective as 'meaningful' words. Underlined words in Fugure 2 are regarded as meaningful terms.

Then the system counts all pairs of co-occurrence: pairs of terms appear in the BoW together. It finds pairs of co-occurrences as shown in Figure 3. Here the system used 2 BoW scopes: one is BoW of the first sentence and the second sentence, and the other is BoW of the second sentence and the third sentence.

- **<<StoryStart>>**
- **A gate agent Supervisor entered the aircraft 11 minutes before the scheduled departure time and wanted to close the aircraft door.**
- **I explained that overhead bins were still open as the last of the passengers had just entered.**
- **And a long line in the aisle was still present.**
- **I told her that it was 11 minutes until scheduled departure.**
- **I had also informed the gate agents twice during boarding that oversized carry on baggage was entering the aircraft.**
- **The agent Supervisor stated that the cargo bin had already closed.**
- **I told her that we may still have bags to check due to being oversized.**
- **At that time a bag came to the forward galley that was obviously well larger than the bag sizer would allow.**

Figure 1- An example of incident report. Red braces indicate Bow of two adjacent sentences.

I was cleared for the runway 08.
CTAF on GPS was 123.
I was broadcasting in the blind on CAVU.

Figure 2 – Sequence of adjacent sentences in an incident report.

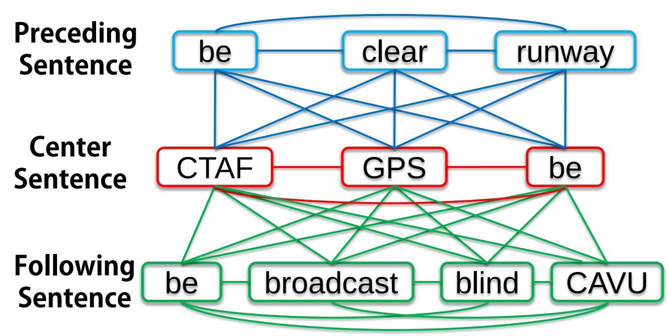


Figure 3 – Co-occurrence pairs of words appearing in the sentences of Figure 2.

3.3 Detection of Comfortable Position of Words in Stories

In addition, our system puts markers that indicate beginning and end of each report. Ahead of the first sentence of a report, we put a marker, which is 'StoryStrart'. Likewise, we put the marker of 'StoryEnd' just behind the actual end of the report.

Those markers will bring information about position in incident stories. Words that have large frequency of co-occurrence with 'StoryStart' marker tend to appear the first sentence in many reports. We regard them as words familiar to the very beginning. Moreover, words co-occur with words familar to the first sentence have a tendency of appearing in the second sentence. Similarly, we can detect words that are likely to appear in the tail of stories by observing co-occurrence with 'StoryEnd' marker.

3.4 Finding Firm Order of Words by Multi-Dimensional Scaling Plot

We apply the process explained above over a large dataset of reports. Observing statistical result of it, our system will distinguish firm tendency of word co-occurrence and word order from fortuitous co-occurrences. The system will discover typical patterns of sequence of words that commonly appear in many incident reports.

To visualize the result, we employ multi-dimensional scaling (MDS) method. MDS is a method to generate scattered plots of words with proper distances that reflect significance of co-occurrences among words.

4 EXPERIMENT

In general, firm tendencies of word order in stories are not exactly equal to of causal relationship, even though we hope the order may be used as an indicator of causality. We test the effectiveness of the proposed method through an experiment with real incident reports.

4.1 Dataset of Aviation Incident Report

We used 4,469 reports of NASA's Aviation Safety Reporting System (ASRS). The data consist of all reports of 2013. We chose the particular part named 'narrative', which is made up of recollected story of event written by people who concerned the incidents. Figure 1 is a part of a real ASRS narrative report.

Characteristics of the data are as the following:

- Language: English
- Author: Persons who concerned the incidents (i.e. pilots, ground crew, cabin crew, etc.)
- Amount of reports: 4,4469. (All data of 2013.)
- Amount of words: 1,365,260.
- Amount of kinds of words: 28,615.
- Amount of sentences: 110,963.
- In average, a report may contain 305 words in 25 sentences.

4.2 Procedure of Analysis and Result

We used KH Coder [4] software for morphological analysis, measurement of co-occurrences, and generation of MDS plot.

Let us see the result with conventional analysis first.

The traditional and ordinary analysis for such huge set of texts is document classification with document-wise BoW. This method treats each report as one BoW and measures co-occurrences to detect main topic of the report.

Figure 4 shows the MDS plot result. Here the system split all reports into 7 groups by using clustering method. (The number of 7 is mere typical number of clustering and does not have particular intention.) We can label the theme names on each cluster as the following:

1. Maintenance and passenger. *(Typical words: passenger, maintenance, takeoff)*
2. Movement on the ground. *(Taxi, ground position)*
3. Communication toward other airplanes. *(Traffic, radio, frequency, cross)*
4. Emergency. *(Emergency, problem, declare, decide, Quick Reference Handbook (QRH))*
5. Coping. *(Procedure, incident, receive, action)*
6. Control of own airplane. *(Course, climb, level, knot)*
7. General verbs. *(Take, use, get, call, make)*

This result means that all of the reports can be classified into those 7 groups depending on their contents. (If it cannot, those clusters will not appear.) In fact, we can find some plausible labels of topics as aviation incidents.

However, we cannot find stories nor causalities of event progress in each cluster.

Now, we apply the proposed method. The MDS plot result is shown in Figure 5.

The difference between the conventional method and the proposed method is only the difference of BoW setting.

We applied clustering on the MDS plot to understand meanings of words gathering each other closely. We found 7 clusters with meanings as the following:

1. Beginning, weather, movement at beginning. *(Typical words: 'StoryStart', takeoff, level, climb, weather, begin, start)*
2. Engine, landing gear, emergency. *(Engine, gear, emergency, declare, QRH, return)*
3. Things about airport. *(Airport, runway, Air traffic control (ATC))*
4. Course setting. *(Degree, clearance, cross, hold, change)*
5. Communication toward other traffics or controller. *(Traffic, controller, tower, frequency, call, hear, ask)*
6. Belief and thinking. *(Believe, think, feel, need)*
7. Ending, reporting incident, maintenance. *('StoryEnd', Incident, problem, issue, inform, maintenance, crew, passenger)*

Most of the stories start with a sentence containing the word of first cluster, and they end with a sentence comprising the seventh cluster. In other words, we can regard that the focus of story moves on the Figure 5, so terms of cluster 2 to 6 will appear as the focus point goes by. Such movement stands for the typical flows of story.

We can summarize the 3 typical flows of the story as Figure 6:

1. <u>Story of machine troubles:</u>
   Cluster 1:*"takeoff, request, indicate"*
   → Cluster 2: *"taxing, captain, First Officer (co-pilot), Quick Reference Handbook, declare*

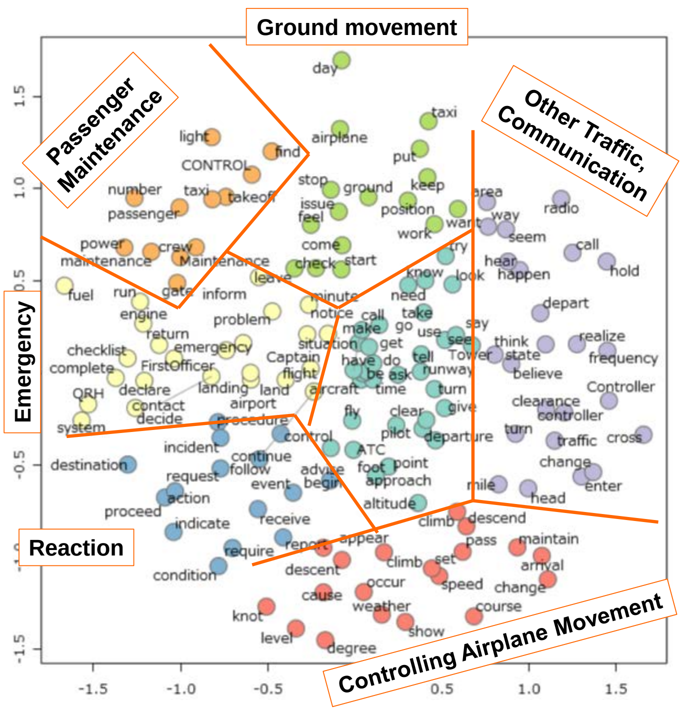
emergency, engine, flap, gear, return"
→ Cluster 7: *"gate, maintenance."*

2. <u>Story of near miss on runway at airport:</u>
Cluster 1: *"start, stop, descent"*
→ Cluster 3:*"airport, runway, ATC (air traffic control), approach, call"*
→ Cluster 7: *"problem, contact, issue."*

3. <u>Story of near miss during cruising:</u>
Cluster 1: *"set, level, climb, speed,"*
→ Cluster 4: *"descend, maintain, course, clearance"*
→ Cluster 5: *"traffic, miss, airspace, call, hear, frequency, call, controller"*
→ Cluster 6: *"believe, think, feel, know, way"*
→ Cluster 7: *"cause, find, issue, situation"*.

Those 3 stories patters are plausible as aviation incidents. So it can be accepted that the proposed method can detect stories contained in huge amount text reports.

As described above, our system found steady orders of words in the incidents to some extent, but the system could not extract exact terms of causes of each incident.



Fiure.4 - Conventional analysis result of multi-dimensional scaling of co-occurrence of words in each report. Colors and labels indicate their clustering.

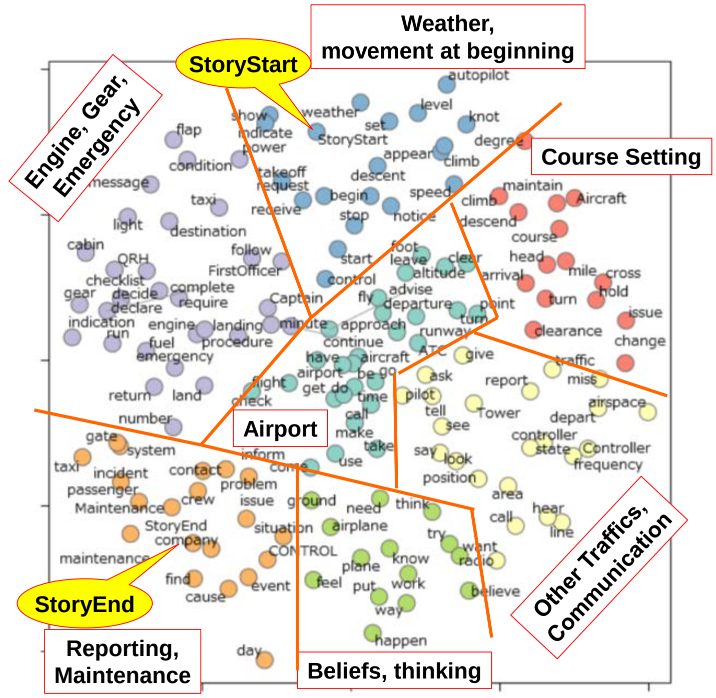


Figure 5 - Result of proposed method. "StoryStart" stands for the beginning of each report, and "StoryEnd" is for the end.

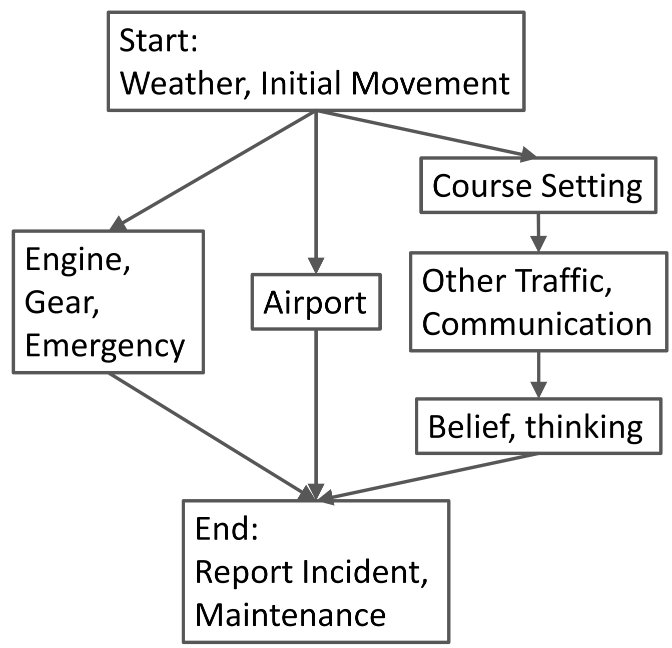


Figure 6 - Typical pattern of "flow of events" found in aviation incident reports.

4.3 Evaluation for Accuracy

In general, evaluation of accuracy of text mining result is difficult. We often do not know the "golden answer" of results of text mining, since the original data is so large to comprehend.

Instead of direct evaluation of accuracy, we can regard performance on reproducibility as one of measures related to accuracy. We consider whether the method can produce similar result from similar dataset or not.

We used all ASRS narrative data reported not in 2013 but in 2015. The number of reports was 5,962. Following the same process as Figure 5, we got a MDS result shown in Figure 7.

It is interesting to note that this result has similarities to Figure 5:

- "StoryStart" maker and "StoryEnd" maker were allocated with long distance. (Stories start from left-upper corner and end at right-lower corner.)
- The same flows can be found. Figure 7 is very similar to the horizontally turned image of Figure 5. For instance, the story pattern of near-miss with other traffic appeared on left side of Figure 7. In MDS plot, The direction in MDS plot does not represent particular meaning,

We can regard our method has enough reproducibility of result, and it can partly support proof of accuracy of the method.

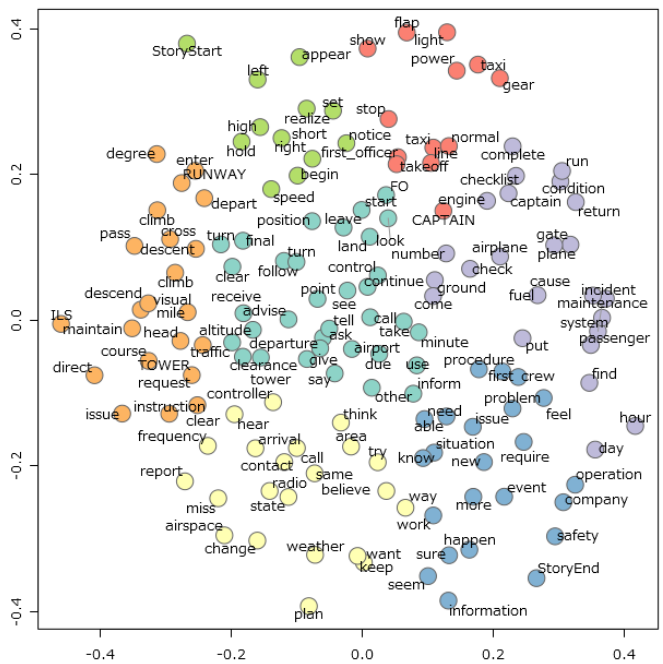


Figure 7 –A visualization result of story flow. The process is same to Figure 5, but it used data reported 2015.

5 CONCLUSION

This paper proposed a text-mining method to analyze texts of accident reports automatically. The method extracts the flows of events of accidents by using bag-of-word of neighboring two sentences. Multi-dimensional scaling plot revealed the flows of accident.

We applied the method on large dataset of 4,468 reports about real aviation incidents and found the typical flows.

In future work, we will utilize more detailed information of words order. In this paper, we ignored order within BoW, but the order of words contains more information about causality. Observing the details, we try to find more precise results on causality.

ACKNOWLEDGEMET

This paper is based on results obtained from a project commissioned by the New Energy and Industrial Technology Development Organization (NEDO).

REFERENCES

1. Center for Chemical Process Safety, *Guidelines for Implementing Process Safety Management*, 2nd Edition, 2016.
2. International Electrotechnical Commission: *IEC 61508-7:2000 Functional Safety of Electrical Electronic/ Programmable Electronic Safety-Related Systems — Part 7, Overview of Techniques and Measures*, 2000.
3. Judea Pearl: *Causality: Models, Reasoning and Inference*, Cambridge University Press, (2000).
4. K. Higuchi, KH Coder, http://khc.sourceforge.net/, 2014.

BIOGRAPHIES

Toru Nakata, PhD
Artificial Intelligence Research Center,
National Institute of Advanced Industrial Science and Technology
Aomi 2-4-7
Koto-ku, Tokyo, 135-0064 Japan
City, State or Province, Postal Code, Country

e-mail: toru-nakata@aist.go.jp

Toru Nakata is the leader of Knowledge Information Research Team of Artificial Intelligence Research Center, at National Institute of Advanced Industrial Science and Technology (AIST). He is also a member of national committee on international safety standard such as IEC 61508. He has long carrier of research on human factor, human error prevention, and risk analysis for industrial safety.