Topic modeling considering context distinguishes temporal and sequential orders of stories

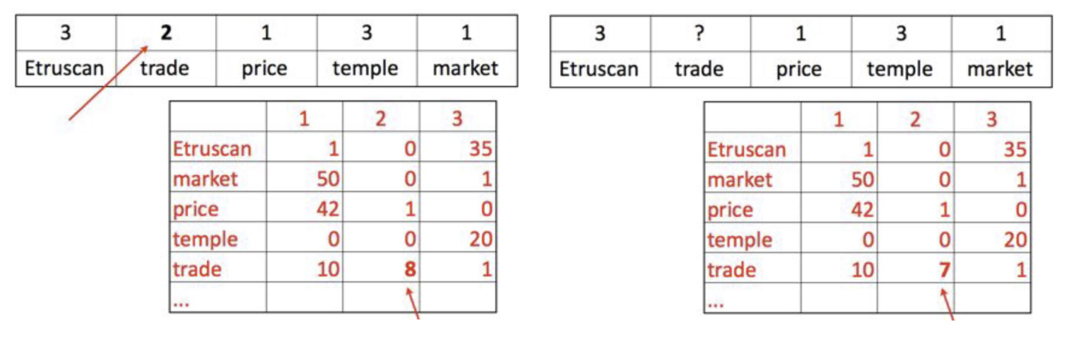
Jongeun Park

Is it possible to create model that not only understand human’s narrative but also evaluate? Our journey begins with this purpose to create model that distinguish various latent themes in many stories and compare it to human’s recalls of each story. Our purpose is concentrated in three parts. First one is to make model that could judge a variety of stream of the story. The second is the model to figure out sequential orders of each story. last is to make a model that can find useless or inadequate parts out in a human’s recall. Specifically, we use topic modeling of LDA method.

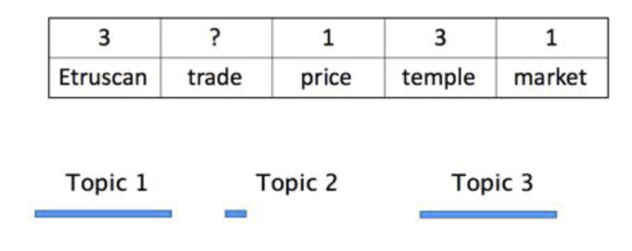
**Problem**

LDA(Latent Drichlet Allocation) is a representative algorism of topic modeling. It is assumed that documents have been comprised of mixture of topics and each topic makes vocabulary of words based on probability distributions. To be more specific, LDA works to maximize joint probability of P(topic t | document d) P(word w | topic t). It can be summarized in 3 steps.

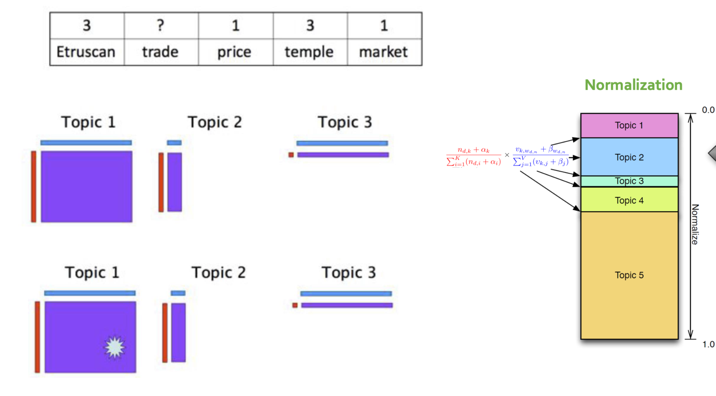
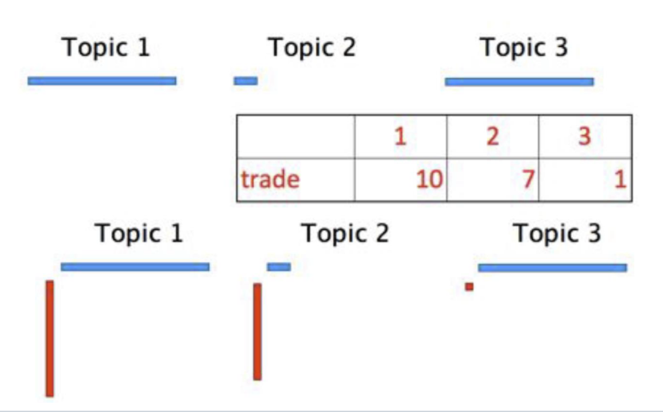
1. Allocate all words to each topic among k(variable allocated by users)



1. Calculate how much this document likes each topic.

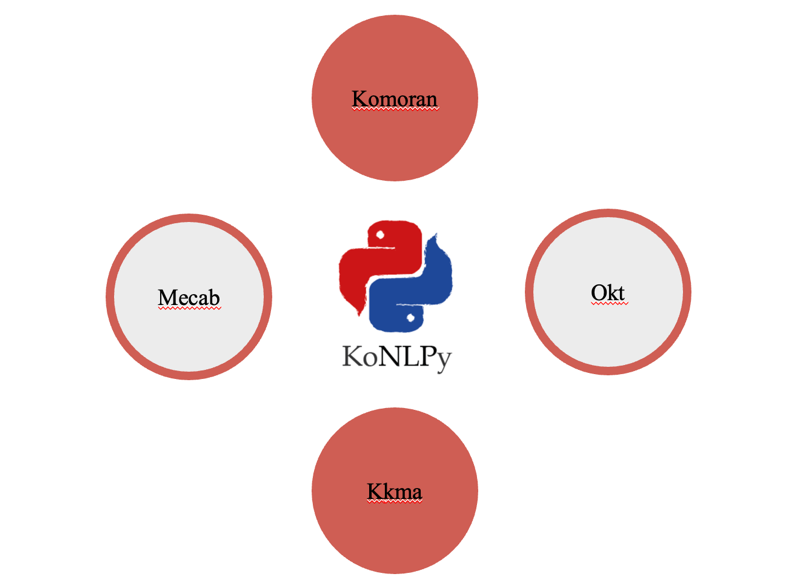


1. Calculate how much each topic likes the word. Then, according to the box area below, topic is allocated randomly to the word.



<Philsung Kang, “Topic modeling Inference”, ppt no.5 >

LDA is a useful algorithm to fine out latent topics. However, it has a limitation that because LDA had been created giving much weight in English circumstances, it does not have many tokenizers for slicing and figuring out sentences. Hence, in this project we are going to use KoNLPy(Korean NLP in Python) library that offers many other morpheme analyzers such as Kkma, Okt, Mecab and Komoran(for comparing quality each other, [see](https://iostream.tistory.com/144#header-n14)). Specifically, Mecab was used in this project because it has fast calculation speed and excellent quality for analyzing Korean morphemes.



**Method**

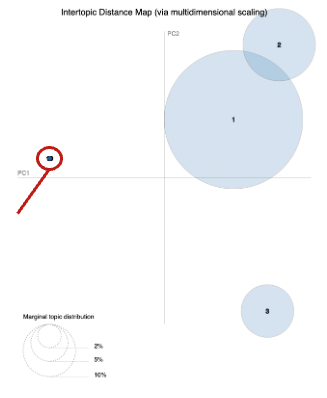
The experiment was conducted by a story called PuruPuru’s egg. To distinguish the dynamic content of the PuruPuru story and compare with subjects’ recalls, we used a topic modeling to discover story’s latent theme. Topic model takes as inputs a sentence of documents and a collections of document and generate two output matrics. First one is a topics matrix whose rows are topics and whose columns are words of vocabulary. The entries indicate how each word is important by each topic. The Second matrix is a topic-proportions matrix, which show how each document is comprised with mixture of topics.

The dataset is collected as original story of PuruPuru’s egg hand-segmented to consider the context. In this process, there was a problem of how long it should be divided when dividing documents into sentences. Therefore, when training the topic model, a verification process was required to find out what the appropriate length of the sentence was.

1. Verification Process (a long sentence vs a short sentence)

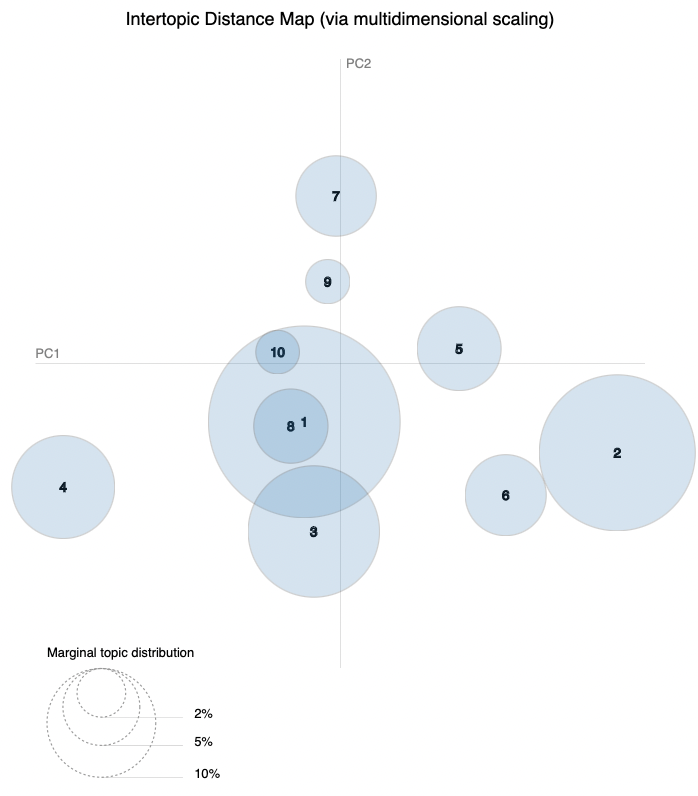
1. Long sentence

First of all, we tested when we learned the entire document. The variable k (number of topics) is assigned to 10, and the results are as follows.



Looking at the results, it can be seen that other than the three topics are not properly distinguished. It can be seen that the remaining seven themes are aggregated in the red circle. This means that the model worked in a different direction from our purpose.

1. Short sentence

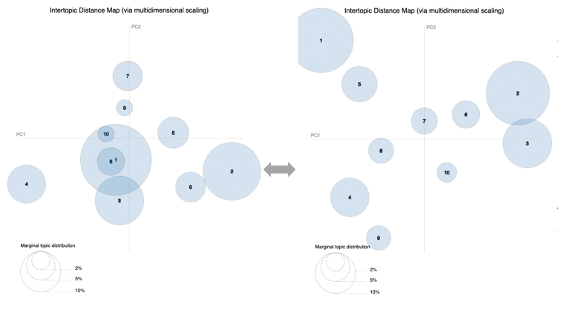


The results are definitely different compared to a long sentence version. It can be seen that the topics are more divided. The reason for showing these results lies in the way the LDA operates. LDA considers the proportions of topics in the document when training, and this is a natural result because the proportions of each topic is different when the short sentence is one document and the long sense is one document. As a result, we found it efficient to break sentences properly and briefly as a single document.

Earlier, it was said that topic modeling receives sentences from documents as input. To be exact, a token sequence is input, and what to set the token is another key.

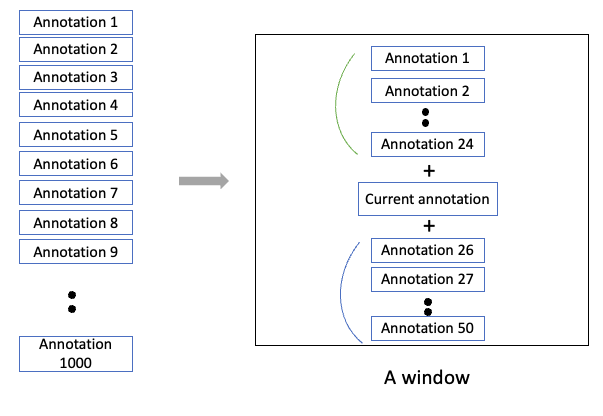
2. Verification Process (Noun vs Morpheme)

In most topic modeling, it is sufficient to understand the meaning of the topic by simply extracting nouns. However, since we are doing topic modeling in one story, we can predict that other parts (mother, stem, conjunction, exclamation) will be important.

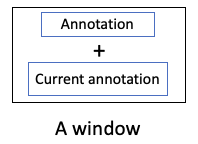


As shown in the results, when setting up tokens on a morpheme basis, the subject can be more clearly distinguished.

Our purpose was to do topics modeling that identify the flow of the story and compare and analyze it with the subject's recalls Therefore, related prior studies have been reviewed. Among them, there was a [paper](https://www.nature.com/articles/s41562-021-01051-6) (Heusser et al.) comparing episodes of the drama Sherlock to subjects' recalls through topic modeling, and our study referred to their methods. They used a different approach to increase the amount of input data and allow the model to consider the context when training the topic model. They first wrote 1,000 annotations, which are explanations of the drama. Each was written according to the passage of time for each episode. They then grouped the annotations to a window to be used as actual input data called a document so that there are up to 50 annotations. In this process, they trained one window with 24 annotations for the prior scene, 25 annotations for the current scene, and 25 annotations for the posterior scene, so that the model would consider the context.



To model the context for annotations near the beginning of the episode (that is, within 25 of the beginning or end), they created overlapping sliding windows that grew in size from one annotation to the full length. They also tapered the sliding-window lengths at the end of the episode, whereby time segments within fewer than 24 annotations of the end of the episode were assigned sliding windows that extended to the end of the episode. Our project also constructed training data in this way. However, there are only 18 annotations that can be used, so only one annotation is attached to the current annotation so that a total of two annotations are one document.



**Preprocessing**

In topic modeling, data preprocessing is essential because it improves the performance of the model and optimizes the model. For pre-processing purposes, we used a total of three lists. The dataset is located in the data folder in the [Git hub](https://github.com/summer2788/NLP_topic_modeling).

1. Word replacement list

-File name: [replace\_list\_hip.xlsx](https://github.com/summer2788/NLP_topic_modeling/blob/main/data/replace_list_hip.xlsx)

Because LDA topic modeling provides results around frequent vocabulary, unifying words with the same meaning into one word is an effective way to perform semantic analysis in text. For example, '러닝' and '런닝' are both words like '달리기'. One can judge that the words all have the same meaning, but the computer recognizes them all as different words. This can result in the occurrence of a keyword being missed because the number of appearances is less as each word is used as a different word, even though the word of a particular meaning is a frequent vocabulary. Therefore, in text mining techniques where the frequency of word appearance is important, such as LDA topic modeling, preceding such word replacement tasks is one of the ways to increase the effectiveness of data analysis.

1. Stopword list

-File name: [stopword\_list\_hip.xlsx](https://github.com/summer2788/NLP_topic_modeling/blob/main/data/stopword_list_hip.xlsx)

A stopword is a vocabulary that is frequently used in text mining, but it is a word such as a predicate or postposition that is far from people's reactions or opinions. For example, there are words such as '~입니다', '~이다', '~와', and '에서'. These words are common in several recalls, but they are considered unnecessary information in this project of the topic modeling. Therefore, it is necessary to organize these disused terms well in the preprocessing stage.

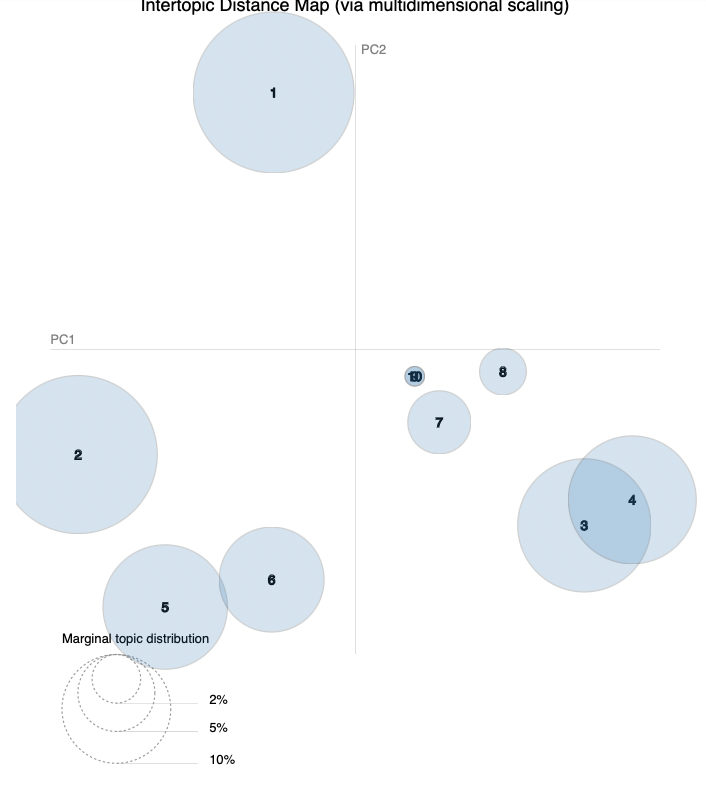
1. One char list

-File name: [one\_char\_list\_hip.xlsx](https://github.com/summer2788/NLP_topic_modeling/blob/main/data/one_char_list_hip.xlsx)

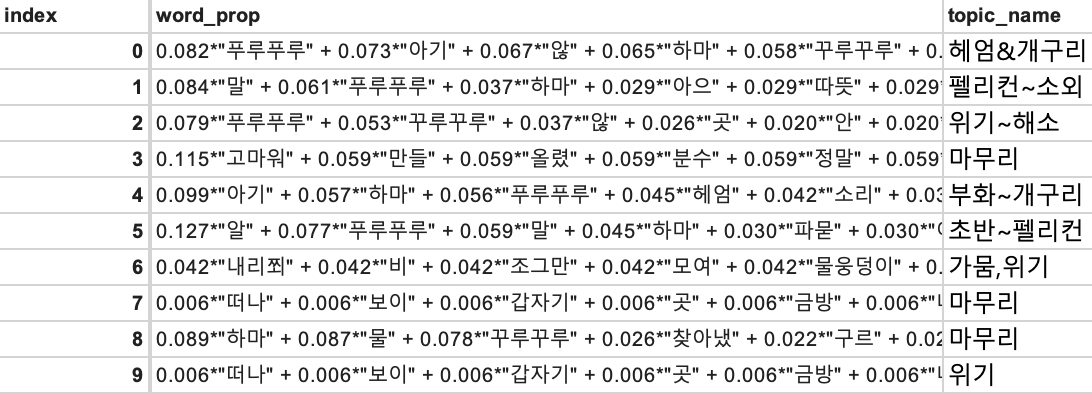
Earlier, it was said that stopwords are a set of unnecessary words in semantic analysis through text mining. After a tokenization operation, a single-letter token is generally treated as an stopwords term. Therefore, we often analyze meaning by using only tokens that have been removed from two or more characters or words. This is the same in English other than Korean. Because it's hard to have a big meaning with just one letter of the alphabet. However, in Korean, it is a single character, but it is sometimes a keyword depending on the domain you analyze. For example, keywords that are importantly related to story development such as '물', '알', and '화' appear. Therefore, in the preprocessing stage, these single-letter but keyword words are exceptionally not removed from the preprocessing stage.

Conclusion

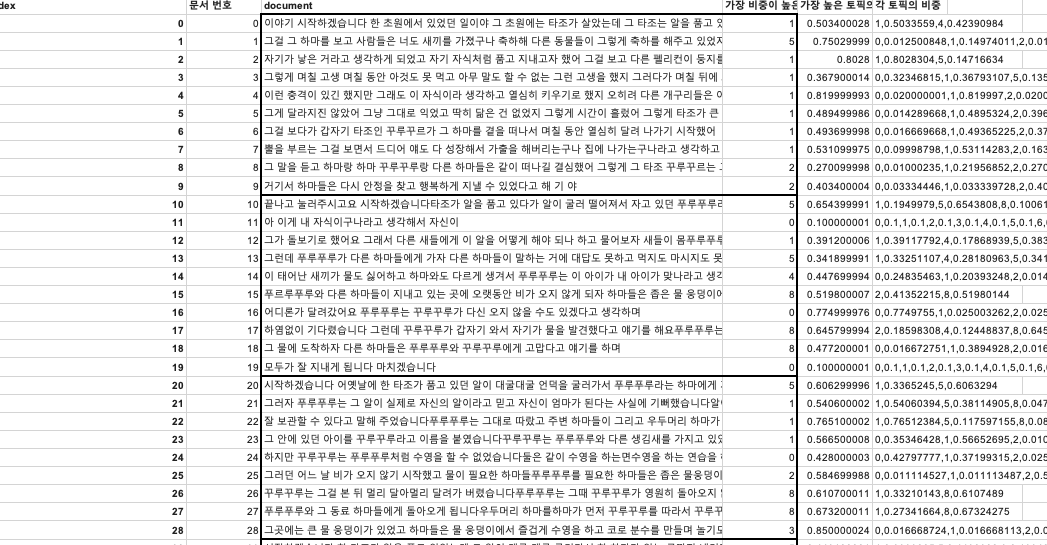
We coded, and to extract a total of 10 topics in the topic model. The number 10 is appropriate number according to the configuration requirements of the story. The model extracted a total of 10 topics, two of which were excluded because they were very fine. A total of eight significant topics were extracted, which was exactly the same number as the scene segments we originally expected.



We mapped each topic to a scene segment and named each topic. After that, the subjects' recalls were analyzed. The subjects' recalls were accurately expressed in the order of the story. Accordingly, topic model was able to accurately grasp the order of the story. As shown in the figure, it can be seen that the order of the story is divided by topic for each subject. It can be seen that topic 5 appears a lot in the beginning of the recall, and topic 2 and 8 appear in the second half of the recall.



Then we named each topic. The topic contained the contents of a specific part of the story.



See the [file](https://github.com/summer2788/NLP_topic_modeling/blob/main/data/topic_word_prop.xlsx) for detailed analysis results

After that, the results of the analysis based on the subject's recall are as follows. The flow of each recall and the flow of the story were largely consistent, and it was also possible to infer what part of the story the subject was talking about. Through this, it was possible to verify the accuracy of the subject's recall.