This notebook displays visualizations for 10 subsequent runs of a LDA training on the same corpus of data.

In [1]:
```
from IPython.core.display import display, HTML
import ktm_prepviz, pyLDAvis
vis = ktm_prepviz.prepviz("rep_gensim")
```

```
In [4]: pyLDAvis.display(vis[0])
```

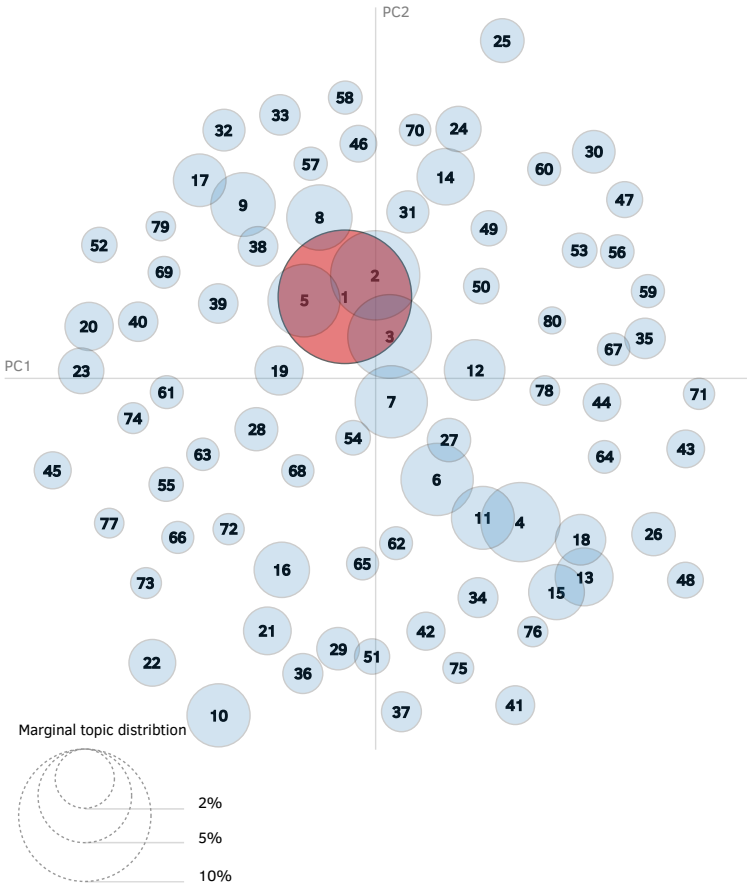
Out[4]:

Selected Topic: 1 | Previous Topic | Next Topic | Clear Topic

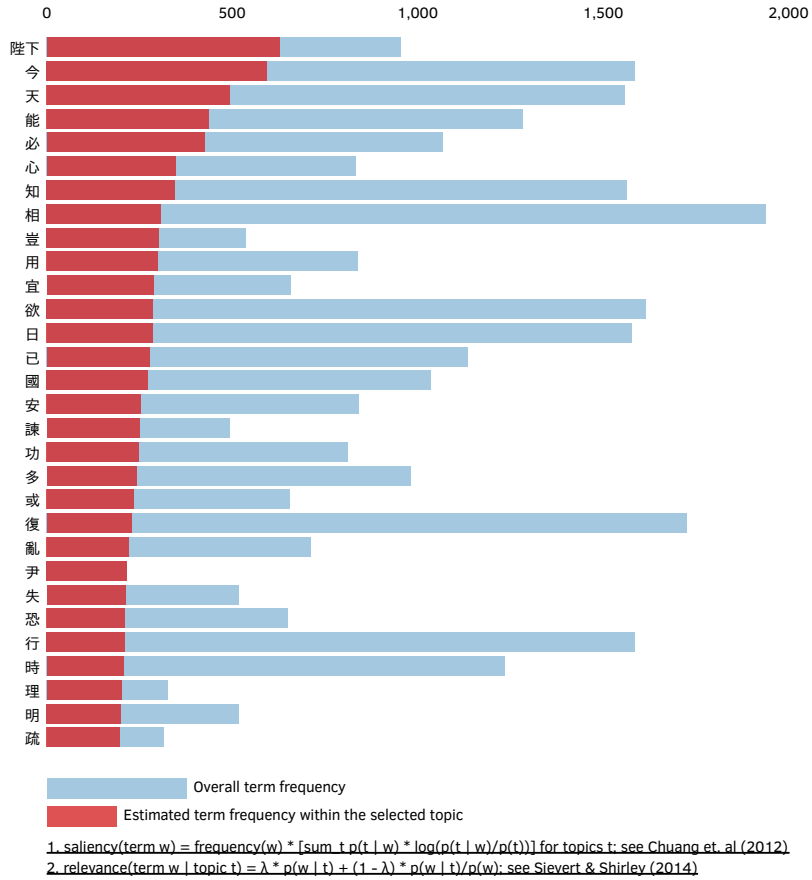
Slide to adjust relevance metric:(2) λ = 1

0.0 0.2 0.4 0.6 0.8 1

Intertopic Distance Map (via multidimensional scaling)



Top-30 Most Relevant Terms for Topic 1 (10.1% of tokens)



1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

```
In [5]: pyLDAvis.display(vis[1])
```

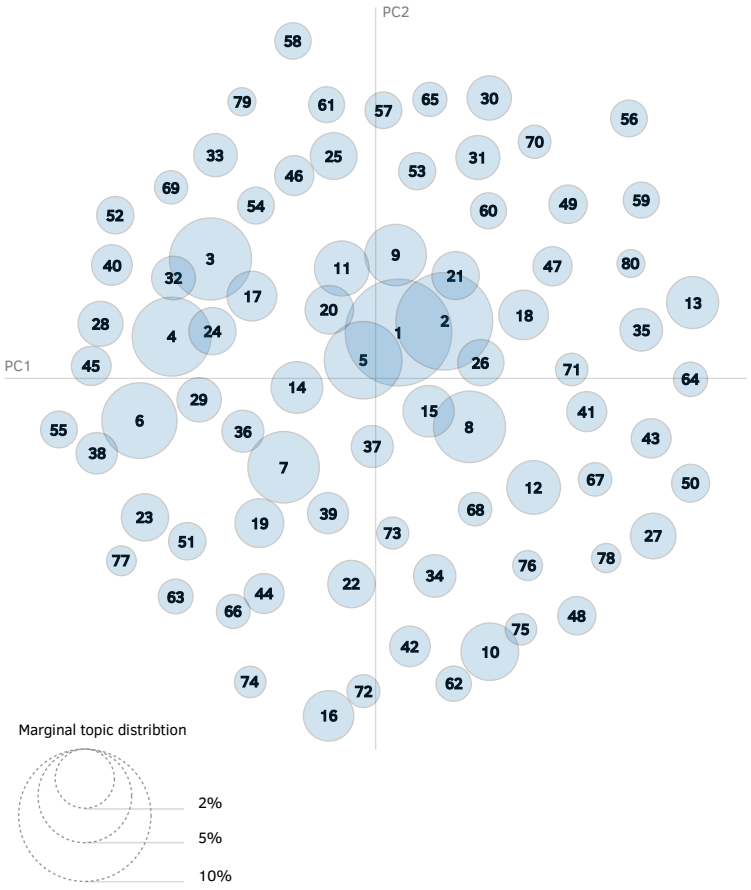
Out[5]:

Selected Topic: 0 | Previous Topic | Next Topic | Clear Topic

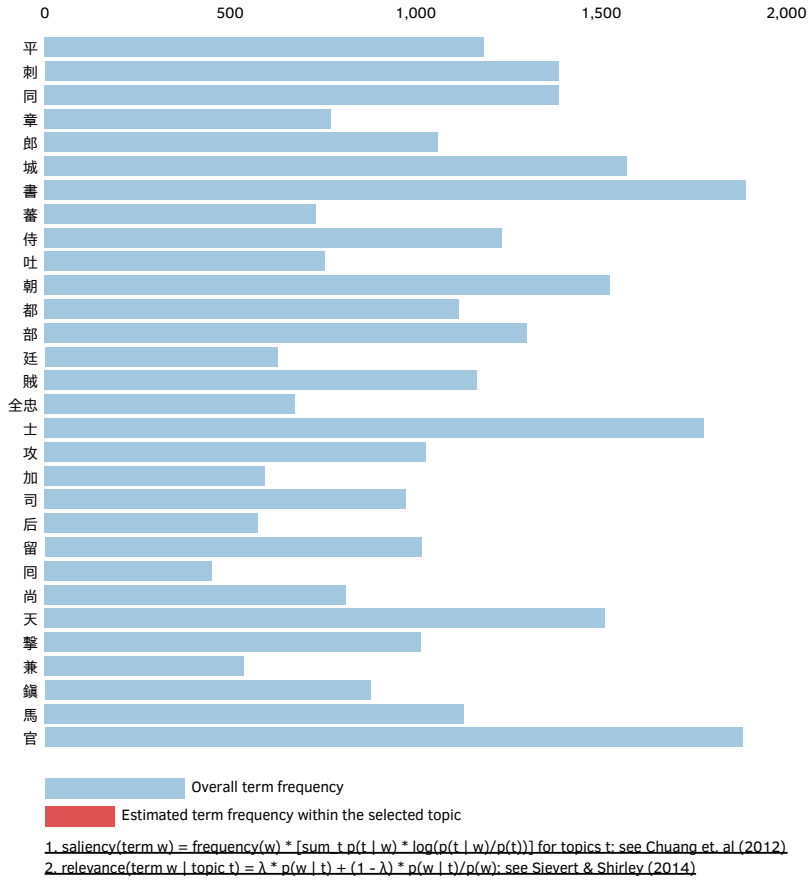
Slide to adjust relevance metric:(2) λ = 1

0.0 0.2 0.4 0.6 0.8 1

Intertopic Distance Map (via multidimensional scaling)



Top-30 Most Salient Terms1



1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

```
In [6]: pyLDAvis.display(vis[2])
```

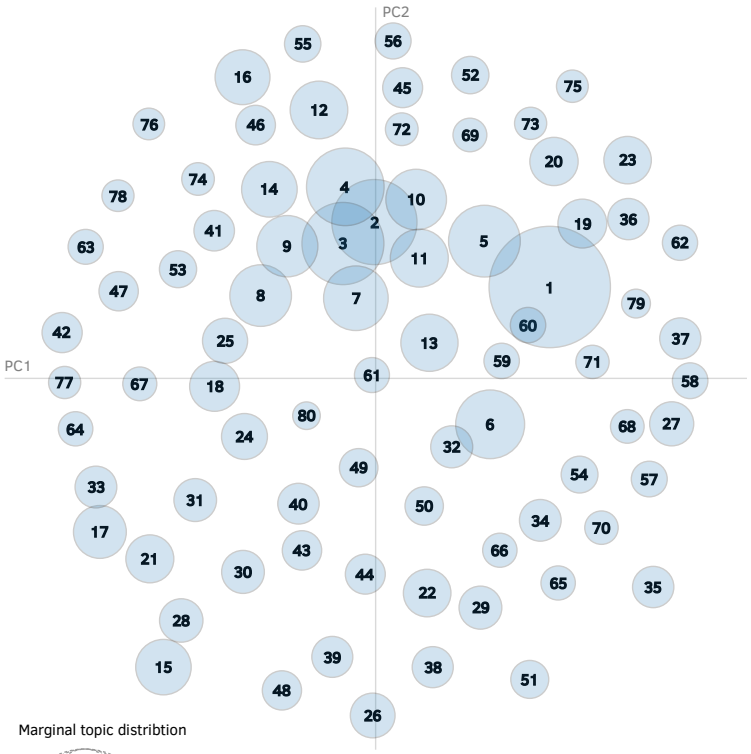
Out[6]:

Selected Topic: 0 Previous Topic Next Topic Clear Topic

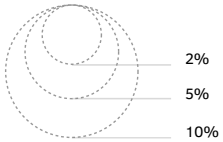
Slide to adjust relevance metric:(2) λ = 1

0.0 0.2 0.4 0.6 0.8 1

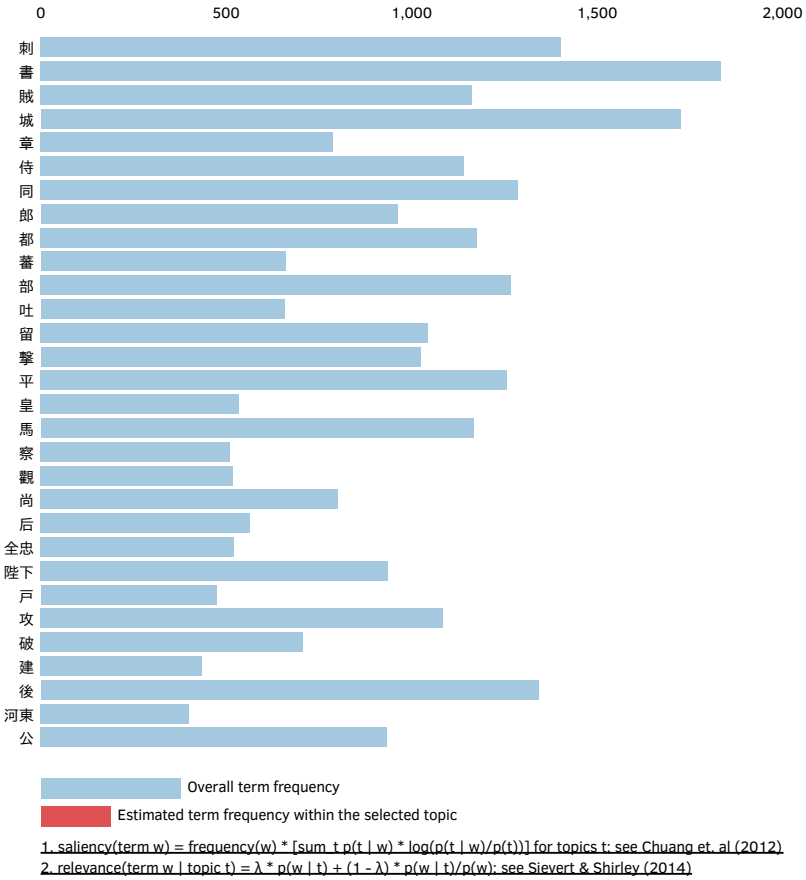
Intertopic Distance Map (via multidimensional scaling)



Marginal topic distribtion



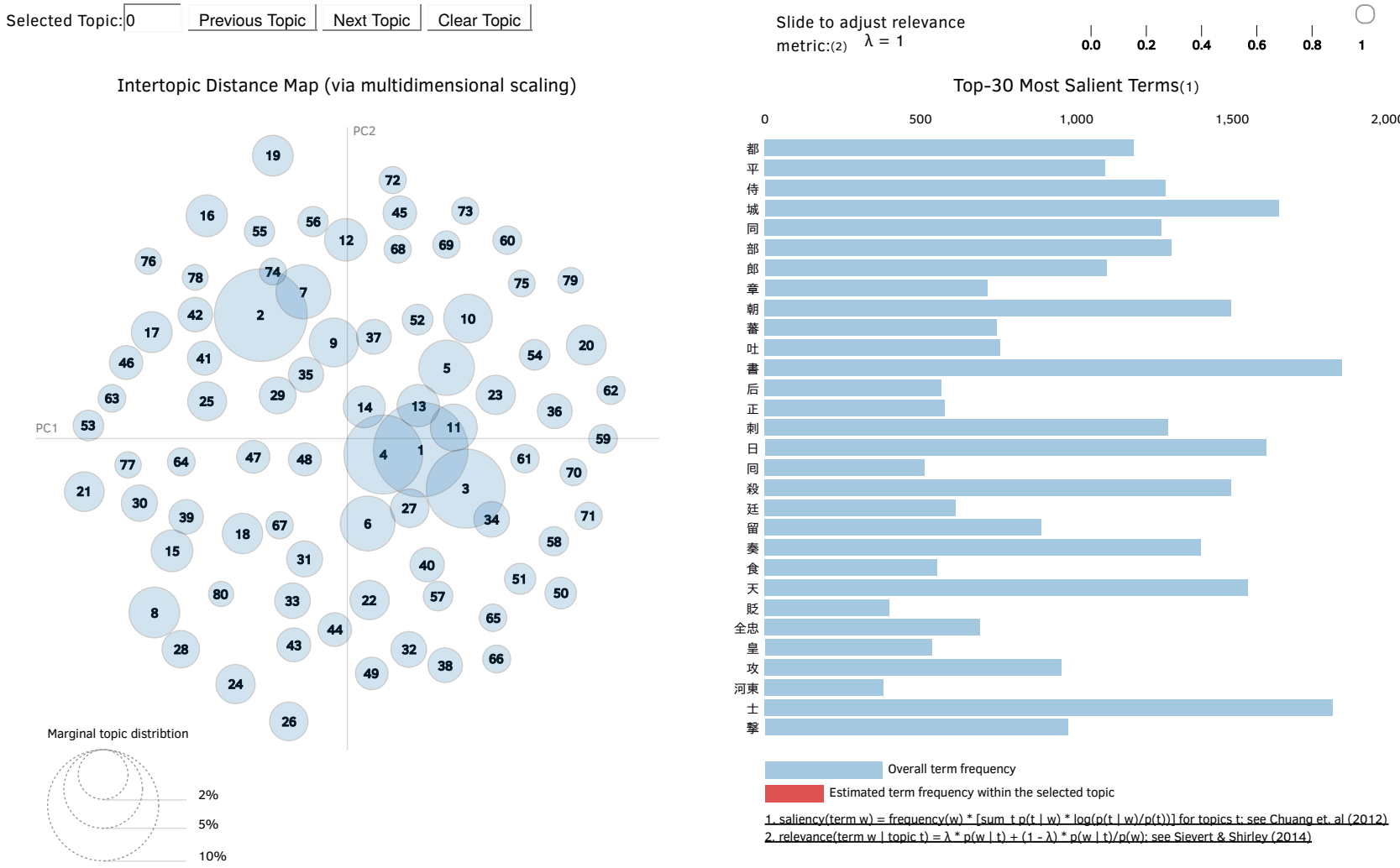
Top-30 Most Salient Terms1



1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

```
In [7]: pyLDAvis.display(vis[3])
```

Out[7]:



```
In [8]: pyLDAvis.display(vis[4])
```

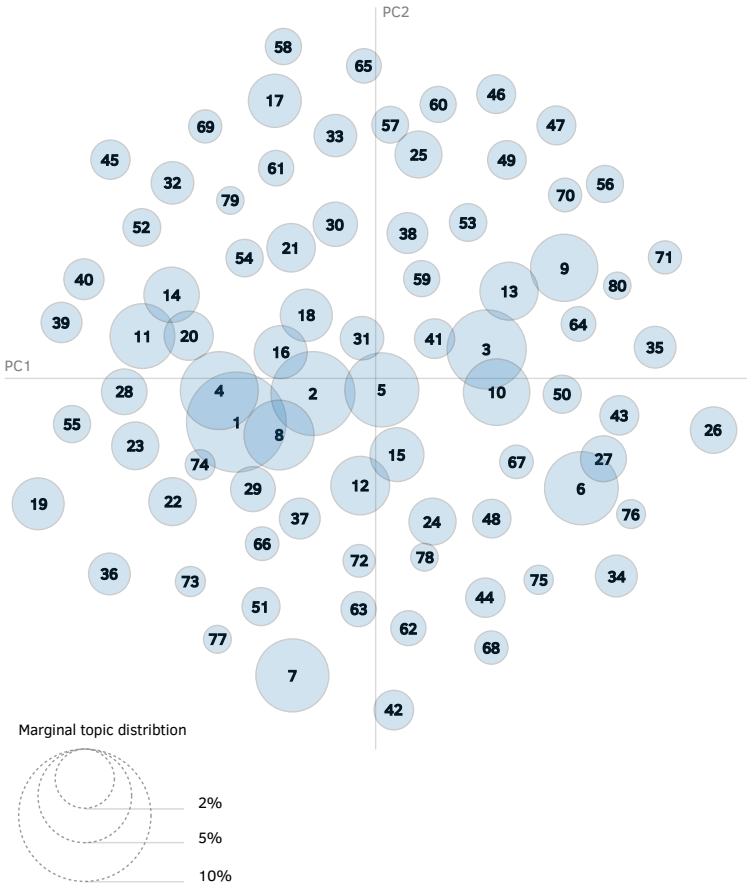
Out[8]:

Selected Topic: 0 | Previous Topic | Next Topic | Clear Topic

Slide to adjust relevance metric:(2) λ = 1

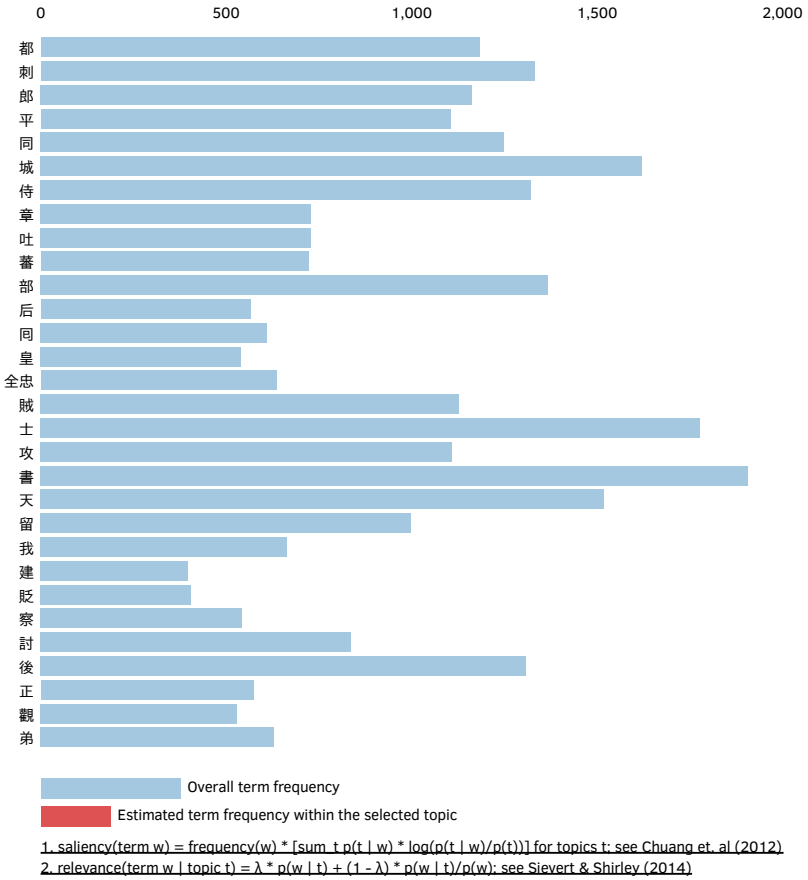
0.0 0.2 0.4 0.6 0.8 1

Intertopic Distance Map (via multidimensional scaling)



Marginal topic distribtion

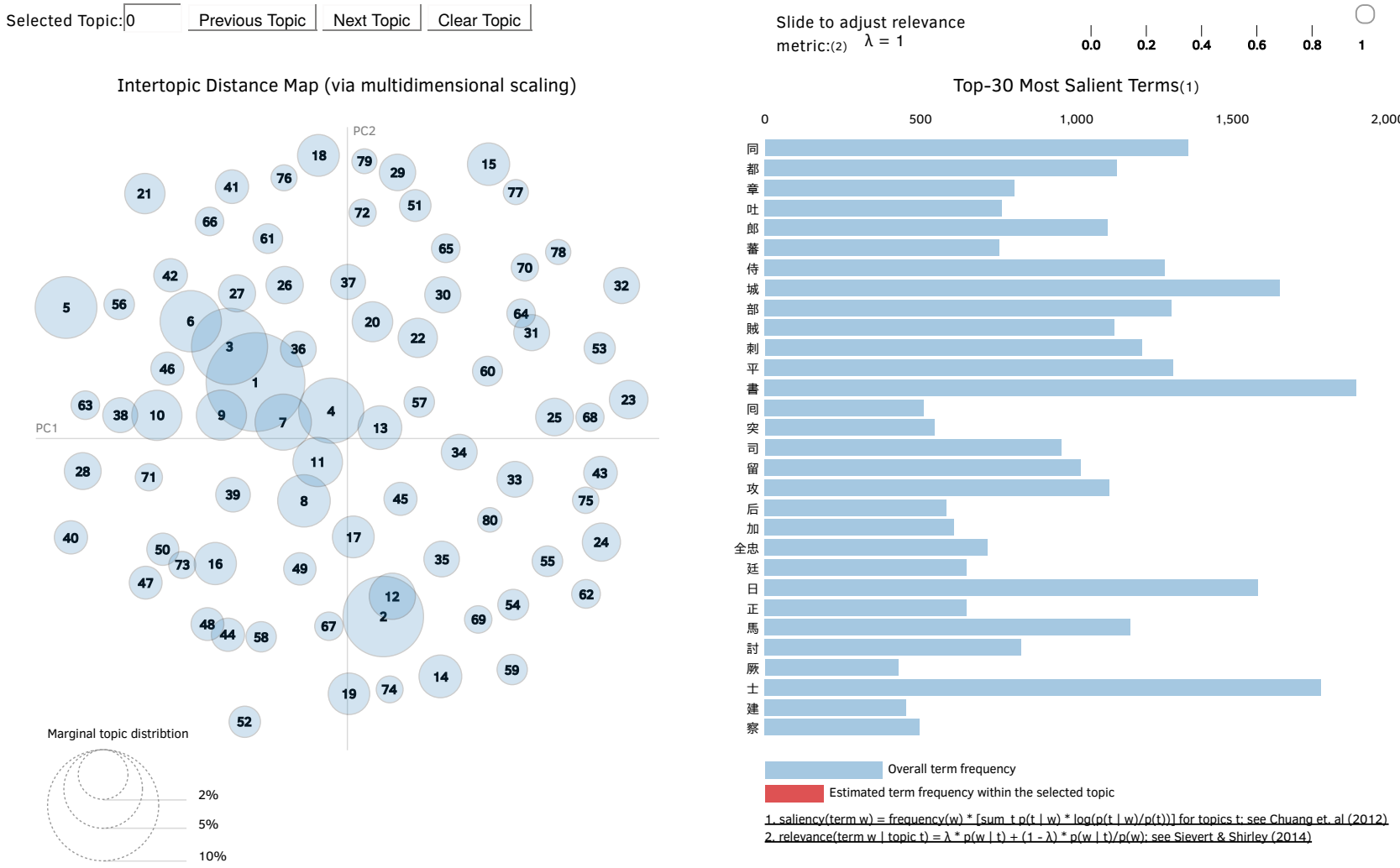
Top-30 Most Salient Terms(1)



1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

```
In [9]: pyLDAvis.display(vis[5])
```

Out[9]:



```
In [10]: pyLDAvis.display(vis[6])
```

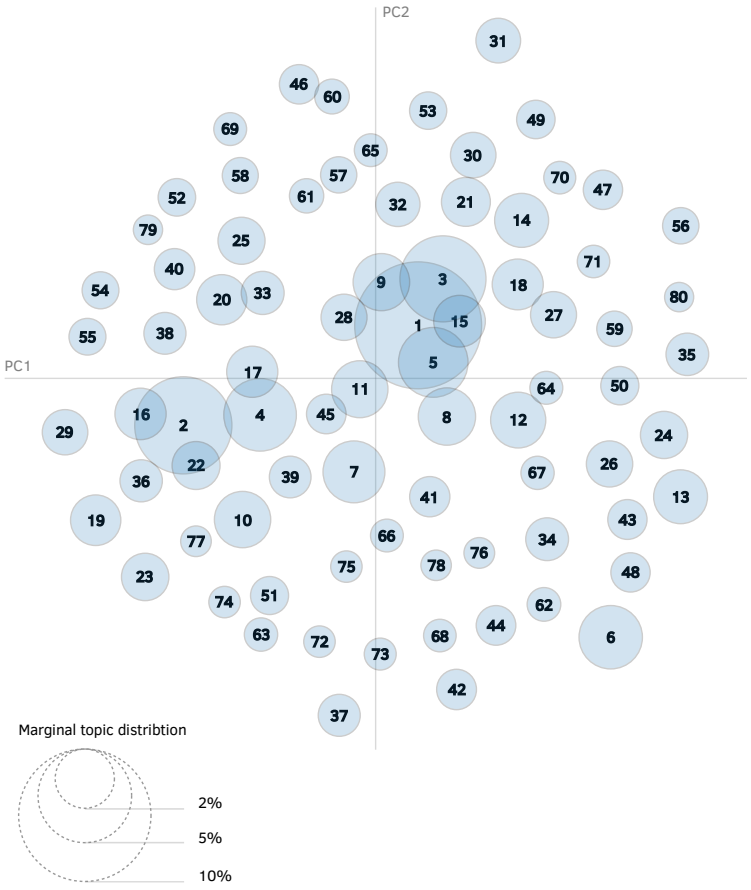
Out[10]:

Selected Topic: 0 Previous Topic Next Topic Clear Topic

Slide to adjust relevance metric:(2) λ = 1

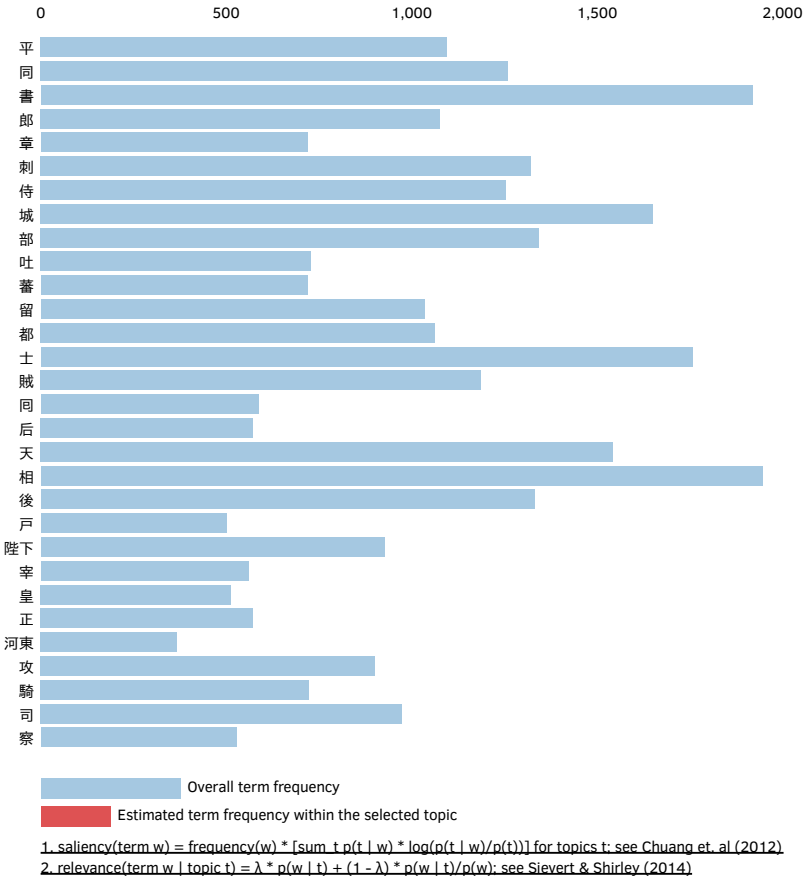
0.0 0.2 0.4 0.6 0.8 1

Intertopic Distance Map (via multidimensional scaling)



Marginal topic distribtion

Top-30 Most Salient Terms1



1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

In [11]:
```
pyLDAvis.display(vis[7])
```

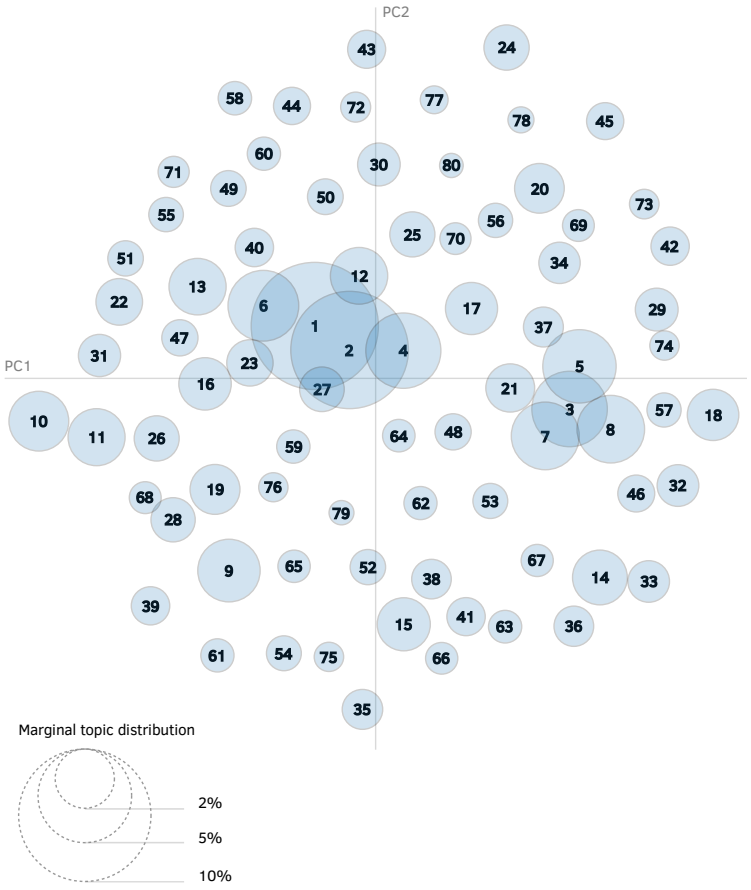
Out[11]:

Selected Topic: 0 | Previous Topic | Next Topic | Clear Topic

Slide to adjust relevance metric:(2) λ = 1

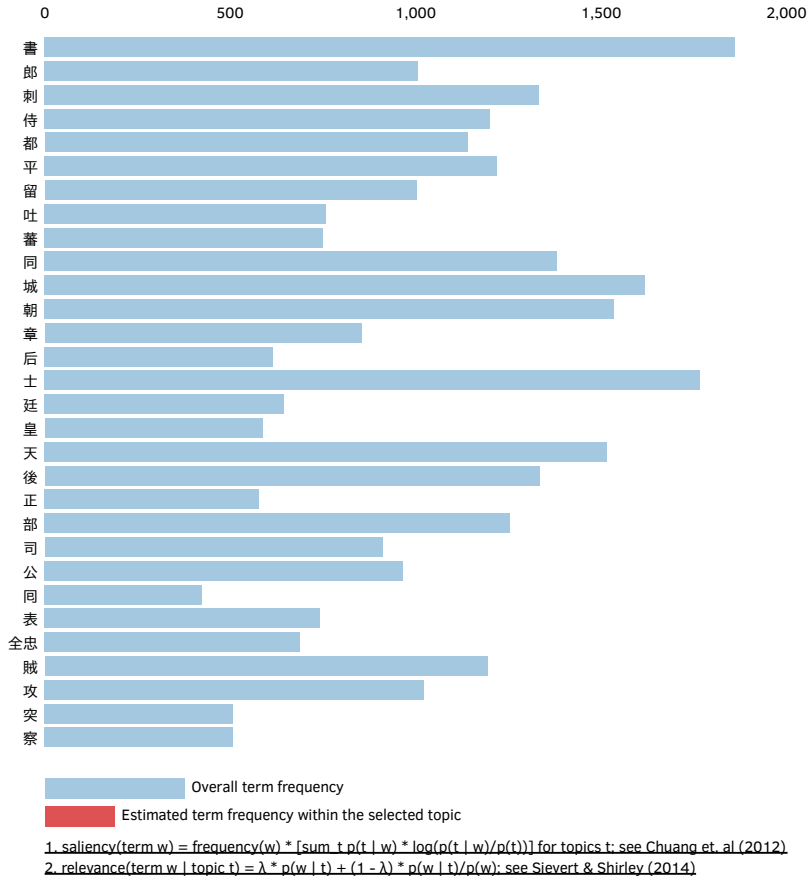
0.0 0.2 0.4 0.6 0.8 1

Intertopic Distance Map (via multidimensional scaling)



Marginal topic distribution

Top-30 Most Salient Terms1



1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

```
In [12]: pyLDAvis.display(vis[8])
```

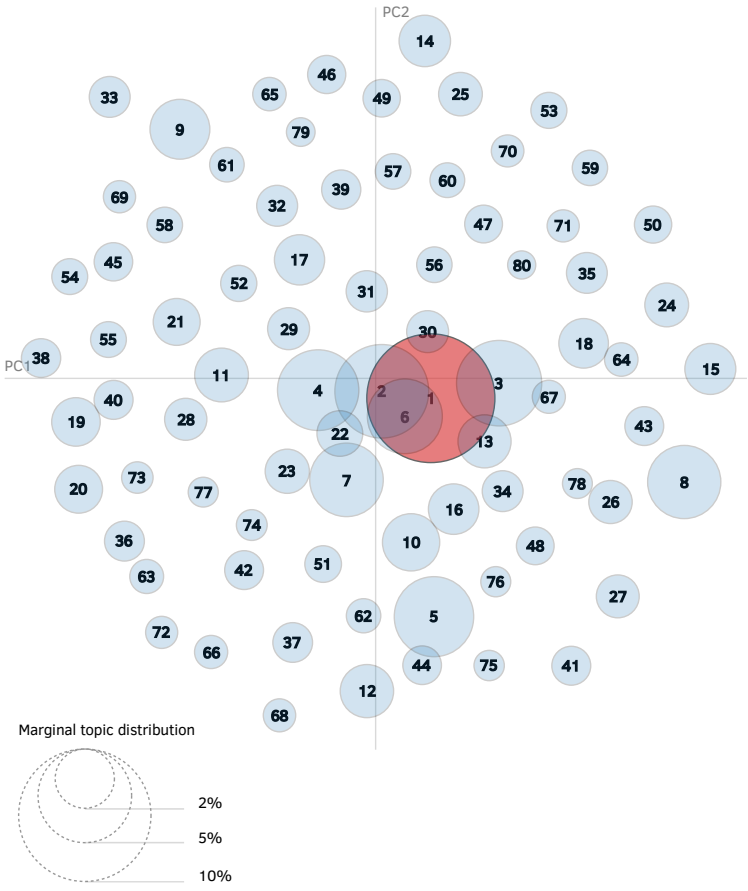
Out[12]:

Selected Topic: 1 | Previous Topic | Next Topic | Clear Topic

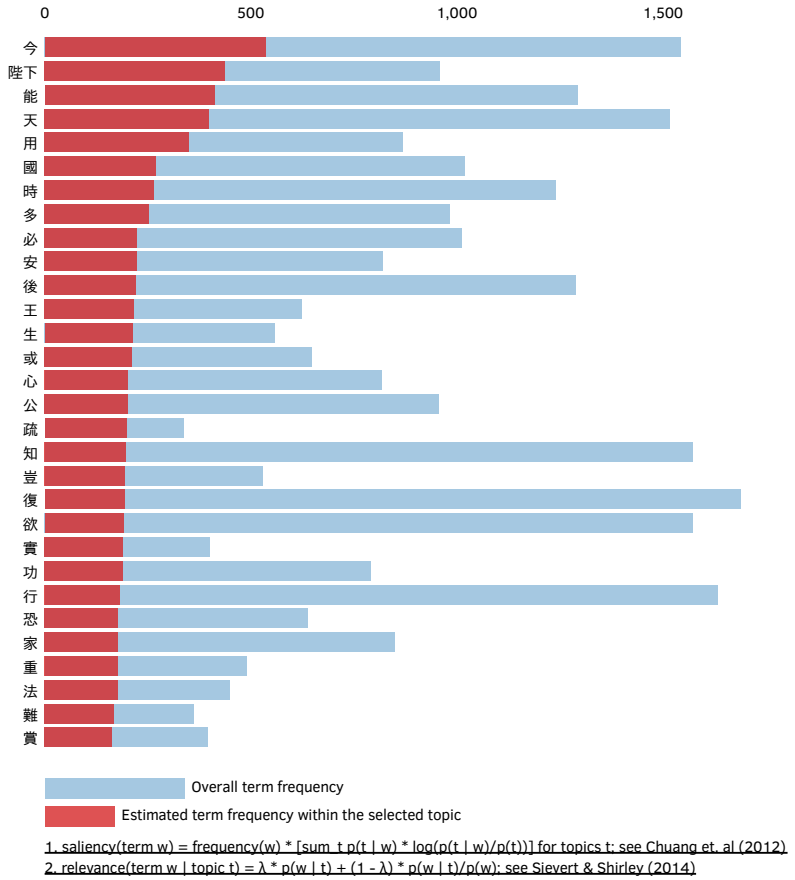
Slide to adjust relevance metric:(2) λ = 1

0.0 0.2 0.4 0.6 0.8 1

Intertopic Distance Map (via multidimensional scaling)



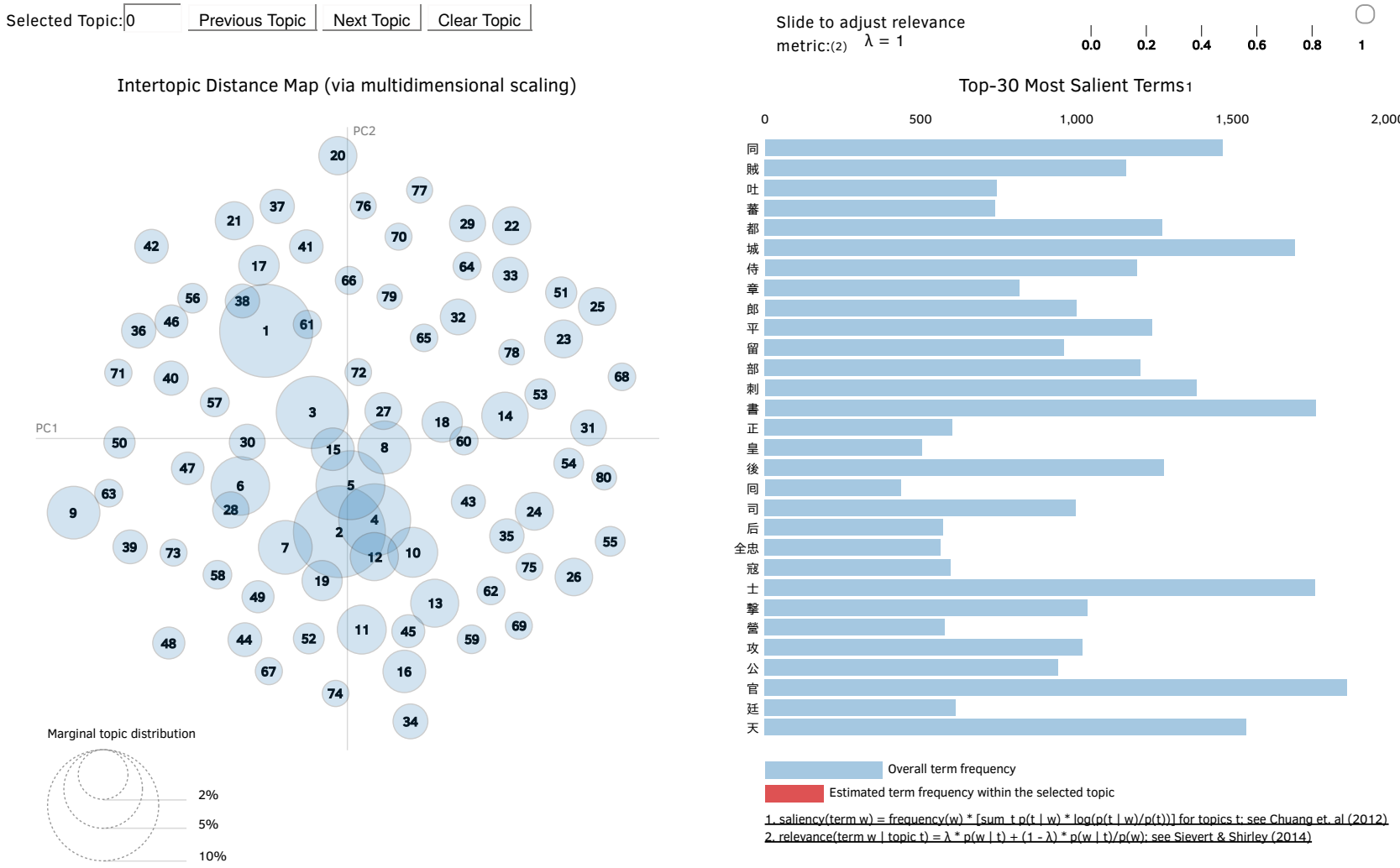
Top-30 Most Relevant Terms for Topic 1 (9.4% of tokens)



1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

```
In [13]: pyLDAvis.display(vis[9])
```

Out[13]:



In [15]:
```
from IPython.core.display import display, HTML
hdata = pyLDAvis.prepared_data_to_html(vis[9])
display(HTML('<h1>The 9th run</h1>'+hdata))
```

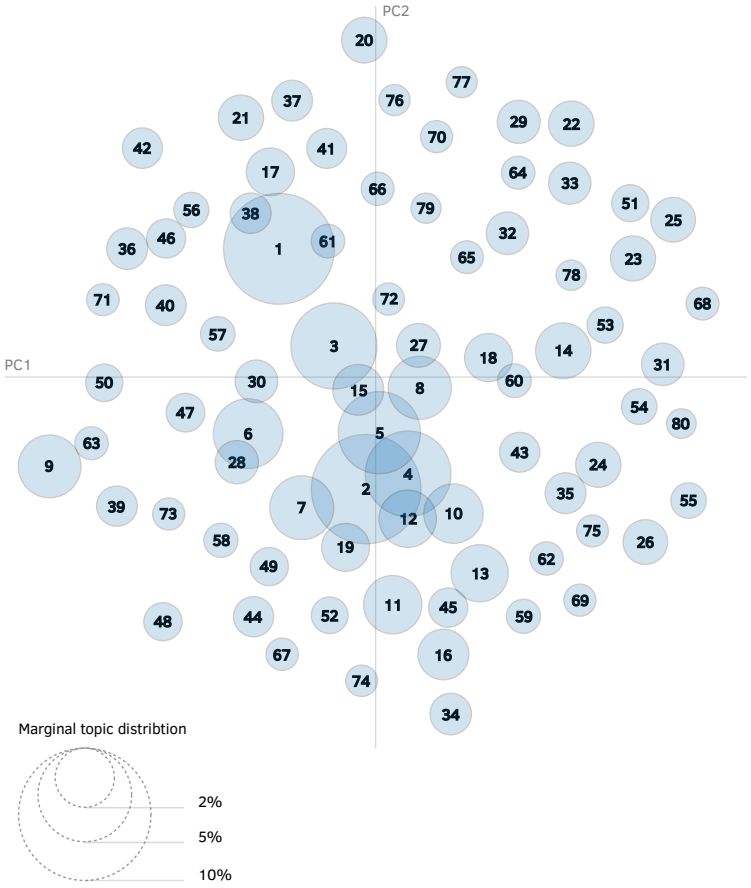
The 9th run

Selected Topic: 0 Previous Topic Next Topic Clear Topic

Slide to adjust relevance metric:(2) λ = 1

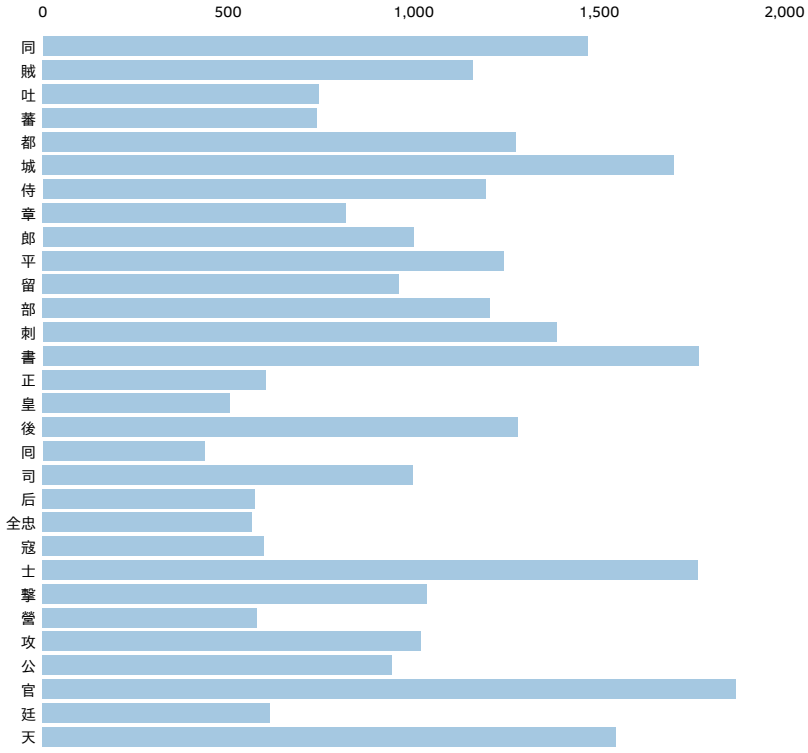
0.0 0.2 0.4 0.6 0.8 1

Intertopic Distance Map (via multidimensional scaling)



Marginal topic distribtion

Top-30 Most Salient Terms1



Overall term frequency

Estimated term frequency within the selected topic

1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

In [17]:
```
reload(ktm_prepviz)
ktm_prepviz.display_with_header(vis[9], "Run 10")
```

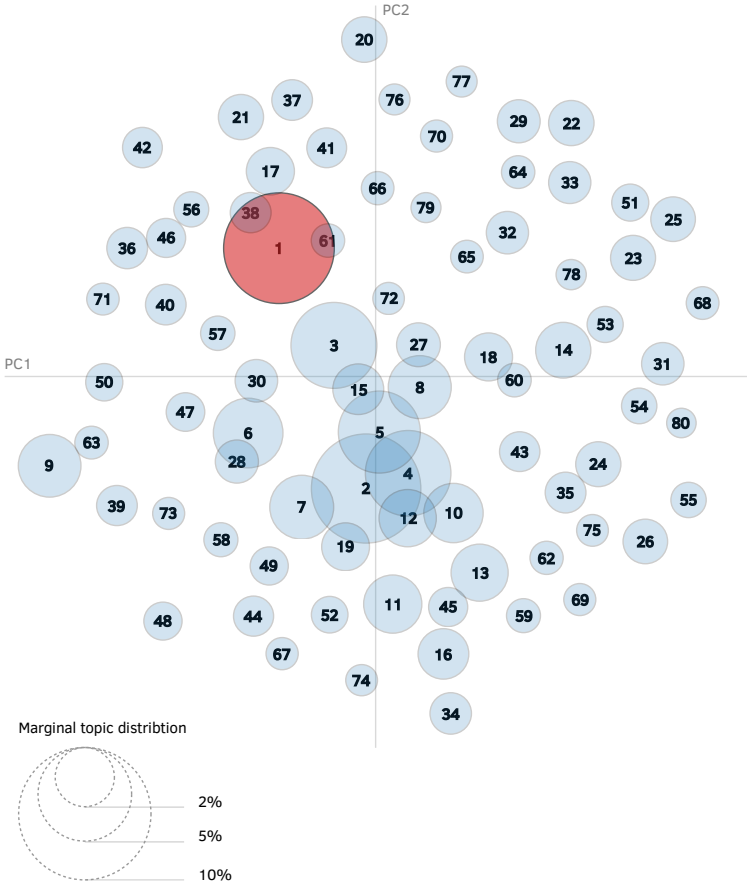
Run 10

Selected Topic:1 Previous Topic Next Topic Clear Topic

Slide to adjust relevance metric:(2) λ = 1

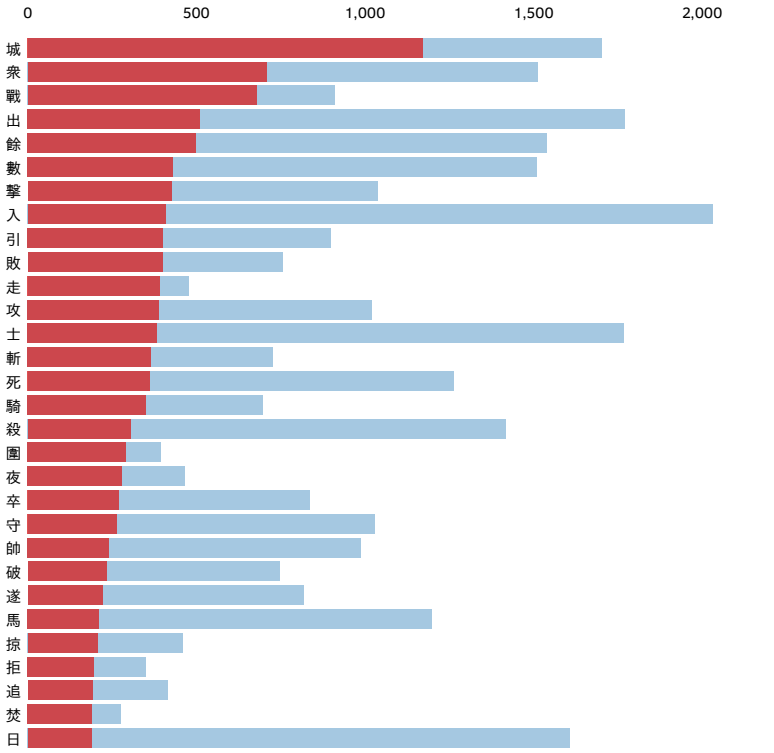
0.0 0.2 0.4 0.6 0.8 1

Intertopic Distance Map (via multidimensional scaling)



Marginal topic distribtion

Top-30 Most Relevant Terms for Topic 1 (6.9% of tokens)



Overall term frequency

Estimated term frequency within the selected topic

1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

In [ ]: