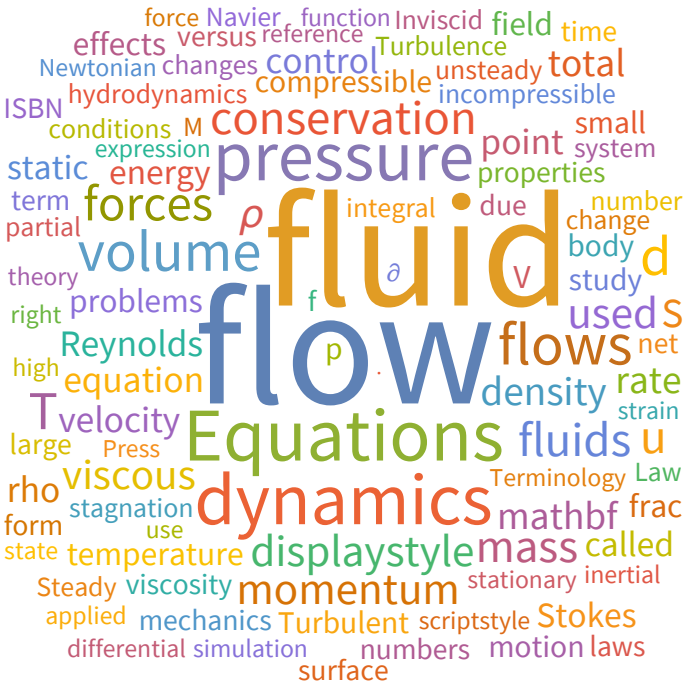
Example of Text Analysis

In[•]:=
```
webpage = WikipediaData["fluid dynamics"];
```

In[•]:=
```
WordCloud[DeleteStopwords[webpage]]
```

Out[•]=



This word cloud makes me think that this conversation is not very interesting. There seems to be no real discussion other than an assortment of people and topics

In[•]:=
```
contents = TextContents[webpage, VerifyInterpretation → True];
```

In[•]:=
```
counts = ReverseSort@CountsBy[contents, Type &]
```

Out[•]=

Type	206

In[•]:= **contents**

Out[•]=

String	Type	Position	Proba
petroleum	Chemical	{443, 451}	0.509
First	Organization	{1439, 1443}	0.865
Law	GovernmentAgency	{1445, 1447}	0.736
Thermodynamics	GovernmentAgency	{1452, 1465}	0.753
Reynolds	TopologicalSpaceType	{1599, 1606}	0.818
Reynolds transport theorem	PhysicalConstant	{1599, 1624}	0.887
theorem	AstronomicalObjectType	{1618, 1624}	0.691
one point	Quantity	{2067, 2075}	0.9
speed	PhysicalConstant	{2307, 2311}	0.826
light	PhysicalConstant	{2316, 2320}	0.841
Newtonian	PhysicalConstant	{2350, 2358}	0.973
Newtonian	TopologicalSpaceType	{2350, 2358}	0.859
Navier–Stokes	PhysicalConstant	{2375, 2387}	0.816
Stokes	TopologicalSpaceType	{2382, 2387}	0.863
perfect gas	PhysicalConstant	{3132, 3142}	0.751
gas constant	PhysicalConstant	{3560, 3571}	0.983
Stokes'	PhysicalConstant	{4176, 4182}	0.865
Stokes' theorem	FamousMathProblem	{4176, 4190}	0.996
S	Element	{5101, 5101}	0.685
S	Element	{5168, 5168}	0.660

rows 1–20 of **206**

```
In[•]:= persons = Normal[Select[contents, Type === "Person" &][[All, "Interpretation"]]]
Out[•]=
        {}

In[•]:= WordCloud[Counts[Flatten@EntityValue[persons, EntityProperty["Person", "Occupation"]]]]
Out[•]=
```

In[•]:=
```
WordCloud[Counts[persons], ImageSize → 500]
```
Out[•]=

In[•]:=
```
Show[WordCloud[counts]]
```
Out[•]=

In[•]:=
```
countries = Normal[Select[contents, Type === "PhysicalConstant" &][[All, "Interpretation"]]]
```
Out[•]=

```
{}
```

In[•]:=
```
WordCloud[countries, ImageSize → 500]
```

Out[•]=

In[•]:=
```
WordCloud[Normal[Select[contents, Type === "MilitaryConflict" &][[All, "Interpretation"]]],
 ImageSize → Large]
```

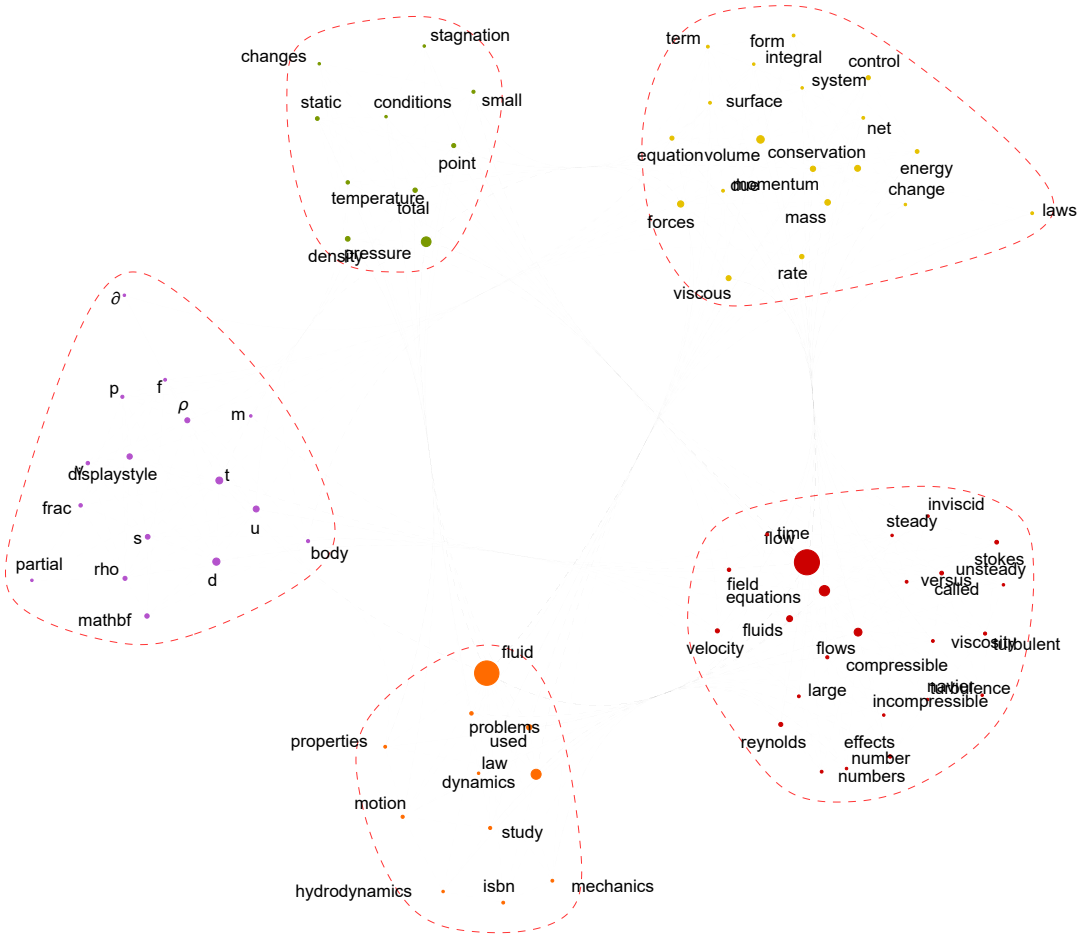
Out[•]=

KeywordGraph

```
In[•]:= Options[KeywordsGraph] = {DirectedEdges → False, EdgeWeight → Automatic, "LowerCase" → True,
    "StopWords" → True, VertexLabels → Automatic, VertexWeight → Automatic};

  KeywordsGraph[text_String, number_Integer?Positive, blist_List : {},
    rlist_List : {}, opts : OptionsPattern[{KeywordsGraph, Graph}]] :=
   Module[{keycounts, keywords, edges, edgeCount, data =
      Replace[DeleteCases[TextWords[If[TrueQ[OptionValue["StopWords"]], DeleteStopwords,
           Identity][If[TrueQ[OptionValue["LowerCase"]], ToLowerCase, Identity][text]]],
        Alternatives @@ blist], rlist, {1}]}, keycounts = Counts[data];
    Quiet[Check[keywords = TakeLargest[keycounts, number],
      Return[Failure["KeywordCount", <|"MessageTemplate" →
          "Number of specified keywords `1` exceeds the actual number of keywords `2`.",
        "MessageParameters" → {number, Length[keycounts]}|>], Module]]];
    edges = Partition[Cases[data, Alternatives @@ Keys[keywords]], 2, 1];
    edgeCount = If[TrueQ[Replace[OptionValue[DirectedEdges], Automatic :> False]],
      KeySelect[Counts[DirectedEdge @@@ edges], #[[1]] ≠ #[[2]] &],
      KeySelect[Counts[Sort /@ UndirectedEdge @@@ edges], #[[1]] ≠ #[[2]] &]];
    Graph[Keys[keywords], Keys[edgeCount], FilterRules[Flatten[Join[{opts,
        VertexWeight → Replace[OptionValue[VertexWeight], Automatic :> Values[keywords]],
        EdgeWeight → Replace[OptionValue[EdgeWeight], Automatic :> Values[edgeCount]]},
       Options[KeywordsGraph]]], Options[Graph]]]]
```

In[•]:=
```
CommunityGraphPlot[KeywordsGraph[DeleteStopwords[webpage], 80,
  VertexSize → "VertexWeight", EdgeStyle → Directive[Black, Dashed, Opacity[0.01]]],
 CommunityBoundaryStyle → Directive[Red, Dashed, Opacity[0.8]],
 ImageSize → Large, GraphLayout → "RadialEmbedding"]
```

Out[•]=

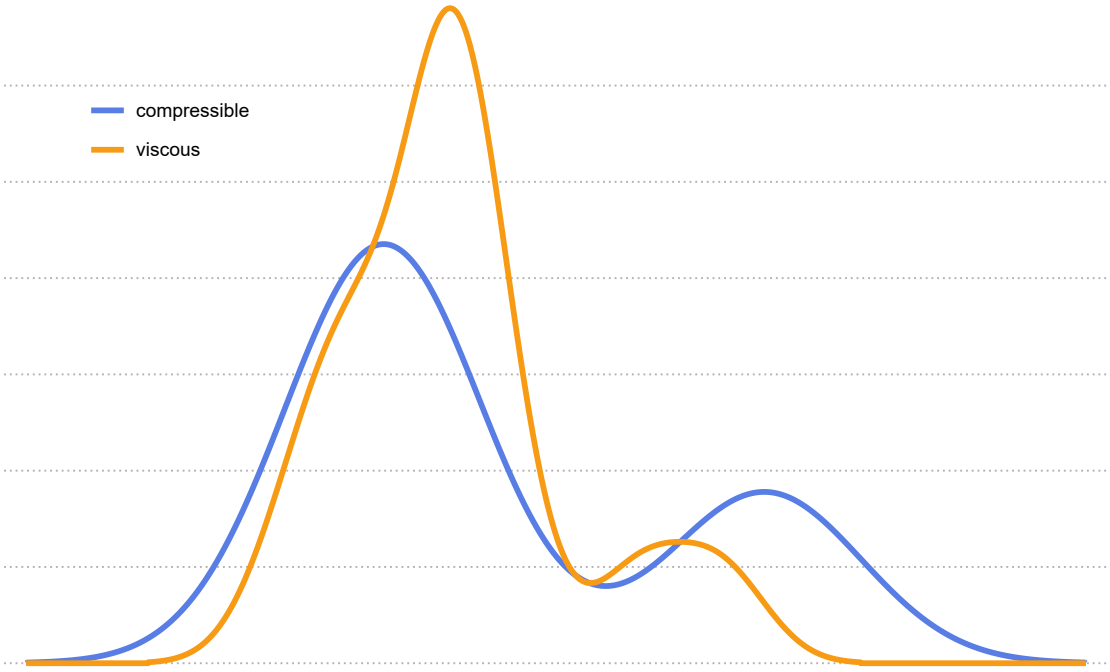


KeywordPlot

```
In[•]:= KeywordPlot // ClearAll;
Options[KeywordPlot] =
  Options[SmoothHistogram] ~Join~ {"PlotFunction" → Automatic, "TopN" → All};
KeywordPlot::nokwd = "Keyword \"`1`\" not found in text.";

(*Operatorforms*)
KeywordPlot[keyword_String, opts : OptionsPattern[]] := KeywordPlot[{keyword}, opts];
KeywordPlot[keywords_List, opts : OptionsPattern[]] :=
  Function[text, KeywordPlot[text, keywords, opts]];

(*Mainform*)
KeywordPlot[text_String, keyword_String, opts : OptionsPattern[]] :=
  KeywordPlot[text, {keyword}, opts];
KeywordPlot[text_String, keywords_List, opts : OptionsPattern[]] :=
 Module[{pltFun, topN, txt, kws, pos, badpos},
  pltFun = OptionValue["PlotFunction"] /. Automatic → SmoothHistogram;
  topN = OptionValue["TopN"];
  txt = RemoveDiacritics@ToLowerCase@text;
  kws = RemoveDiacritics@ToLowerCase@keywords;
  pos = StringPosition[txt, #][[All, 1]] & /@ kws;
  badpos = Position[pos, {}];
  If[badpos =!= {}, ResourceFunction["ResourceSystemMessage"][KeywordPlot::nokwd, #] & /@
    Extract[kws, badpos];
   If[Flatten[pos] == {}, Return@$Failed, (*nokeywordstoplot*)kws = Delete[kws, badpos];
    pos = Delete[pos, badpos];]];
  If[IntegerQ@topN, ord = OrderingBy[pos, -Length[#] &];
   pos = Take[pos[[ord]], UpTo@topN];
   kws = Take[kws[[ord]], UpTo@topN];];
  pltFun @@ {pos, Sequence @@ FilterRules[{opts}, Options[pltFun]],
    PlotLegends → kws, AspectRatio → 1 / GoldenRatio,
    PlotTheme → "Minimal", ImageSize → Automatic, Frame → None}]
```

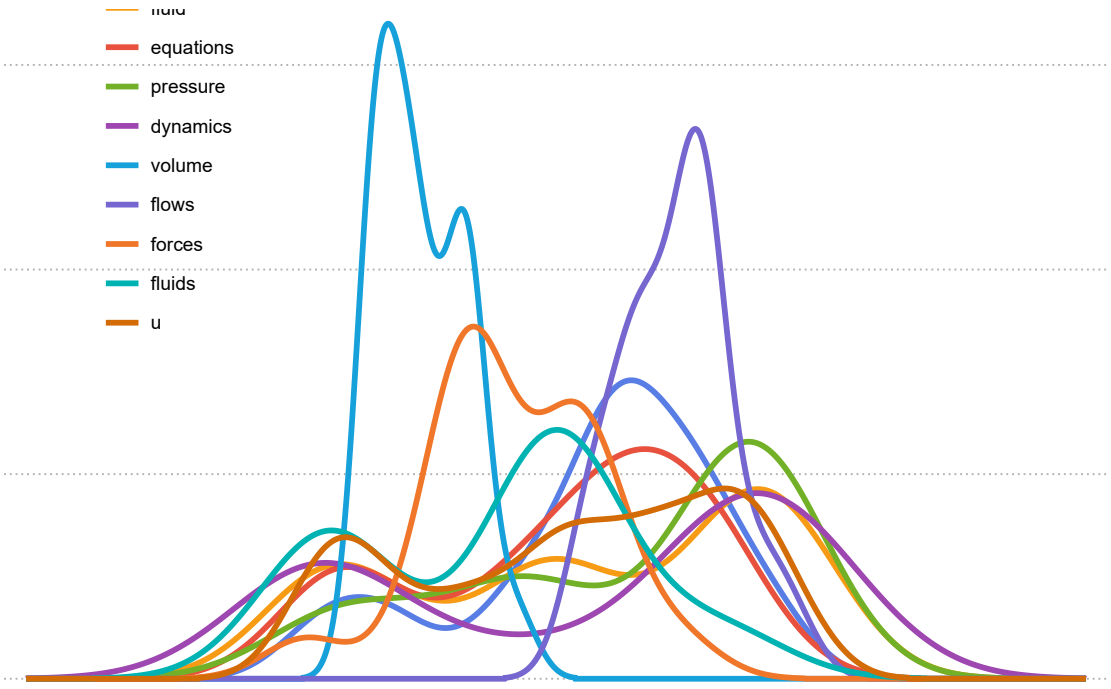
In[•]:=
```
keywords = {"compressible", "viscous"};

KeywordPlot[keywords, ImageSize → Large,
  PlotLegends → Placed[keywords, {{.15, .8}}], PlotTheme → "Business"]@webpage
```

Out[•]=



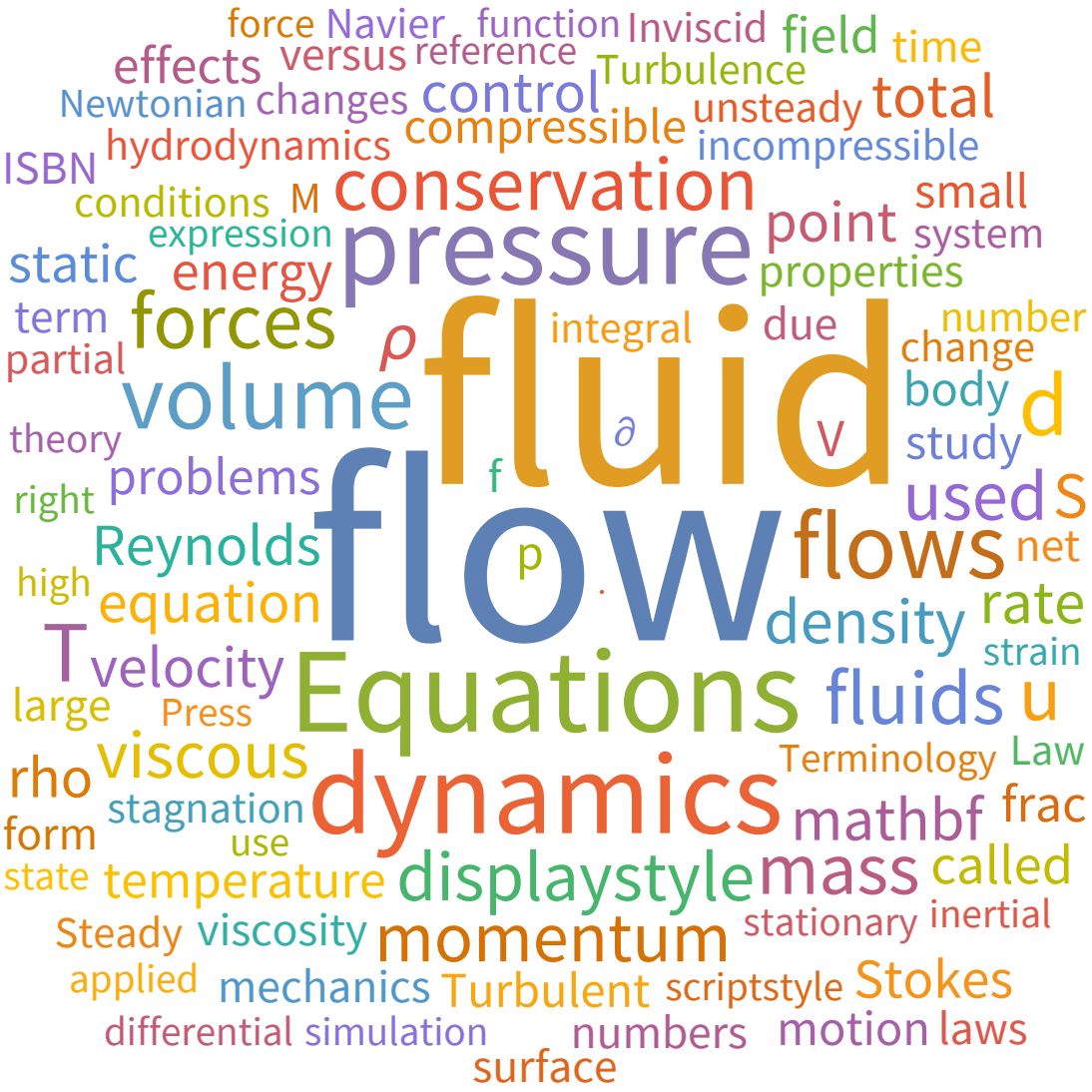
In[•]:=
```
KeywordPlot[
  Normal[Keys[TakeLargest[Counts[TextCases[DeleteStopwords[webpage], "Word"]], 10]]],
  ImageSize → Large, PlotLegends → Placed[
    Normal[Keys[TakeLargest[Counts[TextCases[DeleteStopwords[webpage], "Word"]], 10]]],
    {{.15, .8}}], PlotTheme → "Business"]@webpage
```

Out[•]=



In[•]:=
```
WordCloud[DeleteStopwords[webpage], ImageSize → Large]
```

Out[•]=



Nonfunctioning Resource Functions

In[•]:=
```
ResourceFunction["ReadabilityScore"][webpage]
```

Out[•]=

```
10.1247
```

In[•]:=
```
ResourceFunction["SynonymGraph"]["viscous", 2, ImageSize → Full]
```

Out[•]=