Methods for automatic dating of documents

Leo Leppänen

University of Helsinki
Department of Modern Languages
Language Technology

March 19, 2014

Section 1

Introduction

We need timestamps

- More and more information is easily available online.

We need timestamps

- More and more information is easily available online.
- Also available is more and more "noise".

We need timestamps

- More and more information is easily available online.
- Also available is more and more "noise".
- Once relevant info is now outdated.

We need timestamps

- More and more information is easily available online.
- Also available is more and more "noise".
- Once relevant info is now outdated.
- There's a need for ways of distinquishing between relevant, once relevant and irrelevant.

What we have is not enough

- Filesystem timestamps only set an upper limit.

What we have is not enough

- Filesystem timestamps only set an upper limit.
- Indexing timestamps only set an upper limit.

What we have is not enough

- Filesystem timestamps only set an upper limit.
- Indexing timestamps only set an upper limit.
- Text seldom contains timestamps...

What we have is not enough

- Filesystem timestamps only set an upper limit.
- Indexing timestamps only set an upper limit.
- Text seldom contains timestamps...
- ... and even then they can be in all kinds of different formats.

What we have is not enough

- Filesystem timestamps only set an upper limit.
- Indexing timestamps only set an upper limit.
- Text seldom contains timestamps...
- ... and even then they can be in all kinds of different formats.
- Or no timestamps are available.

What we have is not enough

- Filesystem timestamps only set an upper limit.
- Indexing timestamps only set an upper limit.
- Text seldom contains timestamps...
- ... and even then they can be in all kinds of different formats.
- Or no timestamps are available.

We need automatic tools for dating documents.

Old tools are not enough

- We need efficient, automatic and accurate tools.

Old tools are not enough

- We need efficient, automatic and accurate tools.
- Getting upper limits from existing timestamps is not enough.

Old tools are not enough

- We need efficient, automatic and accurate tools.
- Getting upper limits from existing timestamps is not enough.
- Need to use text itself to determine a more accurate timestamp.

Section 2

Methods of automatic dating

Two groups of methods

1. Language Model based methods

Two groups of methods

1. Language Model based methods
2. Time Expression based methods

But first, Terminology!

- *chronon*
- *tf-idf*

- *nllr*

But first, Terminology!

- *chronon* - An atomic interval of a timeline, the smallest division of time.
- *tf-idf*
- *nllr*

But first, Terminology!

- *chronon* - An atomic interval of a timeline, the smallest division of time.
- *tf-idf* - Text frequency, inverse document frequency.
- *nllr*

But first, Terminology!

- *chronon* - An atomic interval of a timeline, the smallest division of time.
- *tf-idf* - Text frequency, inverse document frequency.
  - An algorithm for determining most "relevant" words from a document.
- *nllr*

But first, Terminology!

- *chronon* - An atomic interval of a timeline, the smallest division of time.
- *tf-idf* - Text frequency, inverse document frequency.
    - An algorithm for determining most "relevant" words from a document.
- *nllr* - Non-linear logarithmic likelihood ratio.

But first, Terminology!

- *chronon* - An atomic interval of a timeline, the smallest division of time.
- *tf-idf* - Text frequency, inverse document frequency.
  - An algorithm for determining most "relevant" words from a document.
- *nllr* - Non-linear logarithmic likelihood ratio.
  - "How much is this thing like this other thing"

Subsection 2

Language Model based methods

Unigram NLLR (de Jong et al., 2005)

- Use a reference corpus of documents with known dates divided to sub-corpuses by chronons.

Unigram NLLR (de Jong et al., 2005)

- Use a reference corpus of documents with known dates divided to sub-corpuses by chronons.
- Compare a document with unknown date to the chronons using NLLR.

Unigram NLLR (de Jong et al., 2005)

- Use a reference corpus of documents with known dates divided to sub-corpuses by chronons.
- Compare a document with unknown date to the chronons using NLLR.
- Pick the chronon with highest NLLR result.

Unigram NLLR (de Jong et al., 2005)

- Use a reference corpus of documents with known dates divided to sub-corpuses by chronons.
- Compare a document with unknown date to the chronons using NLLR.
- Pick the chronon with highest NLLR result.
- Reportedly $20 - 24\%$ accuracy with three month chronons and newspaper articles.

Filtered NLLR(Kanhabua & Nørvåg 2008, 2009)

- Builds upon Unigram NLLR.

Filtered NLLR(Kanhabua & Nørvåg 2008, 2009)

- Builds upon Unigram NLLR.
- Use tf-idf to determine top N_t tokens and only use NLLR on those.

Filtered NLLR(Kanhabua & Nørvåg 2008, 2009)

- Builds upon Unigram NLLR.
- Use tf-idf to determine top N_t tokens and only use NLLR on those.
- Also used NLP-methods:
  - Part-of-Speech tagging.
  - Word sense disambiguation.
  - Collocation extraction.
  - Word filtering.
  - Concept extraction.

Filtered NLLR(Kanhabua & Nørvåg 2008, 2009)

- Builds upon Unigram NLLR.
- Use tf-idf to determine top N_t tokens and only use NLLR on those.
- Also used NLP-methods:
  - Part-of-Speech tagging.
  - Word sense disambiguation.
  - Collocation extraction.
  - Word filtering.
  - Concept extraction.
  - Article doesn't really tell how or to what effect.

Filtered NLLR(Kanhabua & Nørvåg 2008, 2009)

- Builds upon Unigram NLLR.
- Use tf-idf to determine top N_t tokens and only use NLLR on those.
- Also used NLP-methods:
    - Part-of-Speech tagging.
    - Word sense disambiguation.
    - Collocation extraction.
    - Word filtering.
    - Concept extraction.
    - Article doesn't really tell how or to what effect.
- Non-NLP improvements:

Filtered NLLR(Kanhabua & Nørvåg 2008, 2009)

- Builds upon Unigram NLLR.
- Use tf-idf to determine top N_t tokens and only use NLLR on those.
- Also used NLP-methods:
  - Part-of-Speech tagging.
  - Word sense disambiguation.
  - Collocation extraction.
  - Word filtering.
  - Concept extraction.
  - Article doesn't really tell how or to what effect.
- Non-NLP improvements:
  - Better interpolation.
  - Temporal Entropy.

Kumar et al. (2011, 2012), Dalli & Wilks (2006)

- A slightly different algorithm instead of NLLR.

Kumar et al. (2011, 2012), Dalli & Wilks (2006)

- A slightly different algorithm instead of NLLR.
- Otherwise pretty much the same things as Unigram NLLR and Filtered NLLR.

Kumar et al. (2011, 2012), Dalli & Wilks (2006)

- A slightly different algorithm instead of NLLR.
- Otherwise pretty much the same things as Unigram NLLR and Filtered NLLR.
- Comparison of effectiveness not possible due to different corpuses used.

Subsection 3

Time Expression based methods

Chambers (2012)

- Look for time expressions.

Chambers (2012)

- Look for time expressions.
  - *since 2005*
  - *until February 2000*

Chambers (2012)

- Look for time expressions.
  - *since 2005*
  - *until February 2000*
- Use those to determine likely dates.

Section 3

Weaknesses of LM based dating methods

Need of a good reference corpus

- Similarity required between non-dated texts and the reference corpus.

Need of a good reference corpus

- Similarity required between non-dated texts and the reference corpus.
- Newspapers user homogenous language, perhaps LM-methods don't generalize?

Need of a good reference corpus

- Similarity required between non-dated texts and the reference corpus.
- Newspapers user homogenous language, perhaps LM-methods don't generalize?
- Newspapers also have high overlap within chronons if multiple papers are used in the same corpus: Same event in all the sources.

Need of a good reference corpus (cont.)

- A perfect reference corpus would comprise of all the texts from a chronon.

Need of a good reference corpus (cont.)

- A perfect reference corpus would comprise of all the texts from a chronon.
- But then we would already know when all the texts were dated...

Need of a good reference corpus (cont.)

- A perfect reference corpus would comprise of all the texts from a chronon.
- But then we would already know when all the texts were dated...
- Therefore, the reference corpus is incomplete and contains an amount of error.

NLLR requires interpolation

- NLLR is undefined if a token is not present in the reference corpus (division by zero).

NLLR requires interpolation

- NLLR is undefined if a token is not present in the reference corpus (division by zero).
- Need to interpolate non-zero values where zero values would be.

NLLR requires interpolation

- NLLR is undefined if a token is not present in the reference corpus (division by zero).
- Need to interpolate non-zero values where zero values would be.
- Interpolation causes fuzziness and introduces error.

NLLR requires interpolation

- NLLR is undefined if a token is not present in the reference corpus (division by zero).
- Need to interpolate non-zero values where zero values would be.
- Interpolation causes fuzziness and introduces error.
- Sources of error are not additive but multiplicative.

Section 4

Stemming as a preprocessing step

The logic behind stemming

- From a dating perspective, *tsunami* and *tsunamis* are the same thing.

The logic behind stemming

- From a dating perspective, *tsunami* and *tsunamis* are the same thing.
- No benefit from distinguishing between them.

The logic behind stemming

- From a dating perspective, *tsunami* and *tsunamis* are the same thing.
- No benefit from distinguishing between them.
- Differentiation is detrimential from an NLLR perspective.

The problem

- In agglunative languages, stemming removes information.

The problem

- In agglunative languages, stemming removes information.
- Consider the following sentences:
    1. *Presidentti Niinistö lähti tänään Mäntyniemestä Sotšiin*
    2. *Presidentti Niinistö lähti tänään Sotšista Mäntyniemeen*

The problem

- In agglunative languages, stemming removes information.
- Consider the following sentences:
  1. *Presidentti Niinistö lähti tänään Mäntyniemestä Sotšiin*
  2. *Presidentti Niinistö lähti tänään Sotšista Mäntyniemeen*
- In the stemmed sentences, all the difference is lost:
  1. *Presidentti Niinistö lähteä tänään Mäntyniemi Sotši*
  2. *Presidentti Niinistö lähteä tänään Sotši Mäntyniemi*

Is it usefull?

- For stemming to be useful, the benefit must outweight the negative effect from an accuracy perspective.

Is it usefull?

- For stemming to be useful, the benefit must outweight the negative effect from an accuracy perspective.
- Seems fair enough for English:
  - Little inflection.
  - Information such as shown above would already be lost since word order is lost.

Is it usefull?

- For stemming to be useful, the benefit must outweight the negative effect from an accuracy perspective.
- Seems fair enough for English:
  - Little inflection.
  - Information such as shown above would already be lost since word order is lost.
- Less clear for Finnish:
  - Highly agglunative.
  - Stemming removes information that would otherwise be present.

Section 5

A case study

NLLR - a Java program

- A Java program was created (http://github.com/ljleppan/nllr)

NLLR - a Java program

- A Java program was created (http://github.com/ljleppan/nllr)
- Takes a CSV-reference corpus.

NLLR - a Java program

- A Java program was created (http://github.com/ljleppan/nllr)
- Takes a CSV-reference corpus.
- Can use chronons of one day, one week, two weeks or a month.

NLLR - a Java program

- A Java program was created (http://github.com/ljleppan/nllr)
- Takes a CSV-reference corpus.
- Can use chronons of one day, one week, two weeks or a month.
- User can choose if stemming is used.
  - Snowball (Porter2) stemmer.
  - English and Finnish rules selectable.

NLLR - a Java program

- A Java program was created (http://github.com/ljleppan/nllr)
- Takes a CSV-reference corpus.
- Can use chronons of one day, one week, two weeks or a month.
- User can choose if stemming is used.
  - Snowball (Porter2) stemmer.
  - English and Finnish rules selectable.
- Can analyze both single texts (from manual input) or corpuses (from files).

NLLR - a Java program

- A Java program was created (http://github.com/ljleppan/nllr)
- Takes a CSV-reference corpus.
- Can use chronons of one day, one week, two weeks or a month.
- User can choose if stemming is used.
  - Snowball (Porter2) stemmer.
  - English and Finnish rules selectable.
- Can analyze both single texts (from manual input) or corpuses (from files).
- Keeps track of correct and incorrect answers for corpus analysis, provided that the analyzed corpus contains real dates for the texts.

The corpus

- A part of Reuters-21578.

The corpus

- A part of Reuters-21578.
- Articles from Reuters, spanning a few months in 1987.

The corpus

- A part of Reuters-21578.
- Articles from Reuters, spanning a few months in 1987.
- Used a python script to convert the original SGML to CSV, then manual labour to remove some badly formatted lines.

The corpus

- A part of Reuters-21578.
- Articles from Reuters, spanning a few months in 1987.
- Used a python script to convert the original SGML to CSV, then manual labour to remove some badly formatted lines.
- $n = 14899$ with a 2-12 split:
  - Smaller set of $n = 2768$
  - Larger set of $n = 12131$

Results for 12-2 split

- Bigger set as training data (reference corpus), smaller set as test data.

Results for 12-2 split

- Bigger set as training data (reference corpus), smaller set as test data.
- Run for all four chronons, both with and without stemming.

Results for 12-2 split

- Bigger set as training data (reference corpus), smaller set as test data.
- Run for all four chronons, both with and without stemming.
- Difference between stemmed and non-stemmed runs is weird.

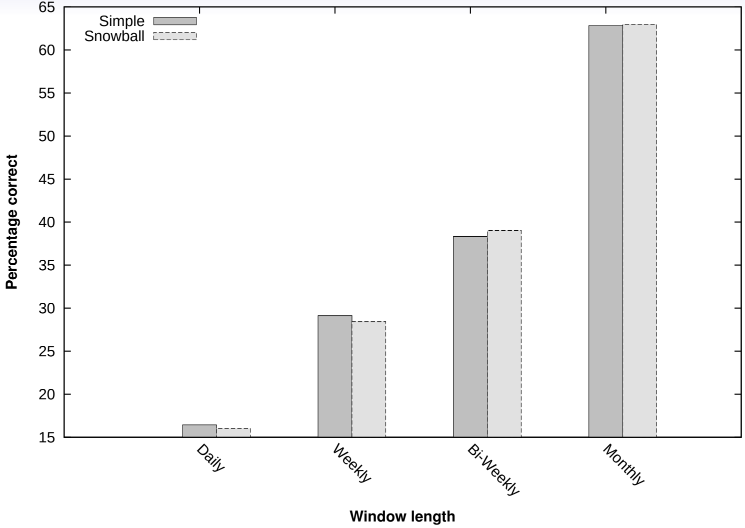

Figure: Results using 12-2 split

Try a new run, this time with 2-12 split, to confirm the results.

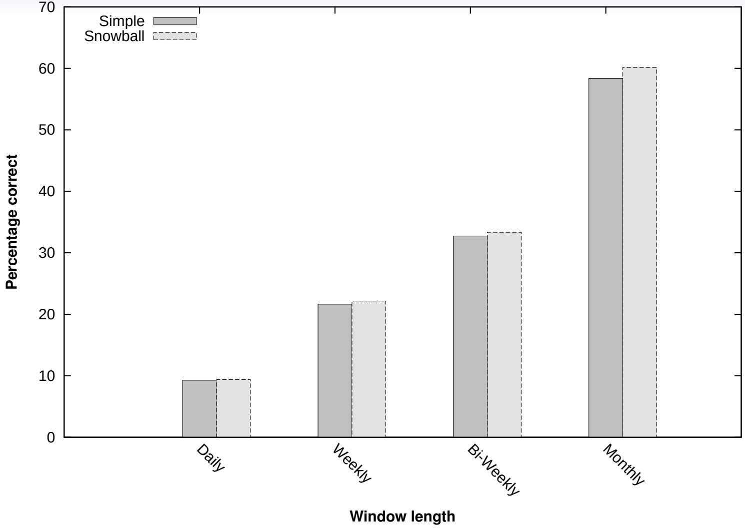

Figure: Results using 2-12 split

A little bit more sane

- This time, something of a trend in favor of Snowball.

A little bit more sane

- This time, something of a trend in favor of Snowball.
- Still less improvement than expected.

A little bit more sane

- This time, something of a trend in favor of Snowball.
- Still less improvement than expected.
- What does this mean for the concerns about stemming presented earlier?

Conclusions

- Stemming doesn't seem to do much even with English.

Conclusions

- Stemming doesn't seem to do much even with English.
- If the benefit from stemming is so small with English, can the benefit compensate for the larger loss of information when dealing with Finnish.

Conclusions

- Stemming doesn't seem to do much even with English.
- If the benefit from stemming is so small with English, can the benefit compensate for the larger loss of information when dealing with Finnish.
- Trying to get hold of a sufficiently large Finnish corpus with known dates for the texts.