School of Electrical & Electronic Engineering

Who Wrote the Letter to Hebrews?

Data Mining for Authorship Detection

Final Seminar

Date: 5th Apr 2012

Presented By:
Kai He, Yan Xie
Zhaokun Wang

Supervisor: Derek Abbott
Co-Supervisor: Brian Ng

THE UNIVERSITY of ADELAIDE

THE UNIVERSITY of ADELAIDE

Presentation Structure

- Project Objectives
- Background
- Previous Research
- Project Progress
- Work Management
- Conclusion
- Reference

THE UNIVERSITY of ADELAIDE

Project Objectives

- Study the controversy over the authorship of *The Letter to Hebrews*.
- Study Data Mining techniques and implement them to authorship attribution problems.
- Provide conclusion to the authorship controversy based on researching findings.

THE UNIVERSITY of ADELAIDE

Background

THE UNIVERSITY of ADELAIDE

The Controversy
- Who wrote The Letter to Hebrews?

The Letter to Hebrews:

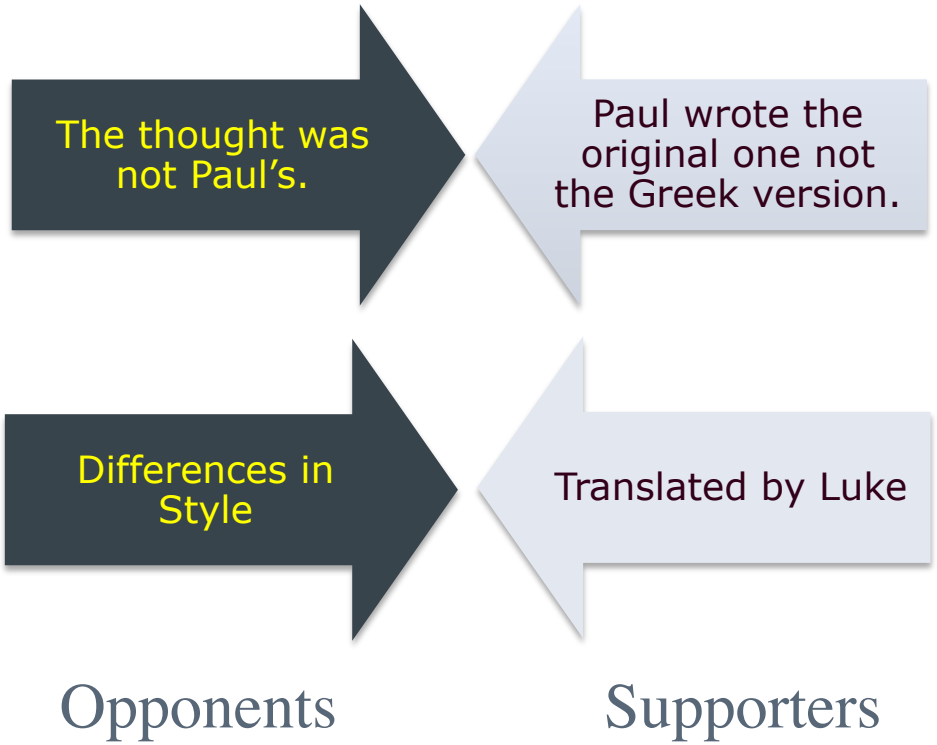
- is one of the books in the New Testament.
- its main content is to exhort Christians to persevere in the face of persecution.
- unknown author.

The Church largely agreed Paul is the author until the Reformation.

THE UNIVERSITY of ADELAIDE

The Controversy (Cont'd)

Paul's authorship of *The letter to Hebrews* is in doubt.



Possible Author of Hebrews

6 authors from New Testament:

- **Apollos**
- **Barnabas**
- **Clement**
- **Luke**
- **Paul**
- **Peter**

THE UNIVERSITY of ADELAIDE

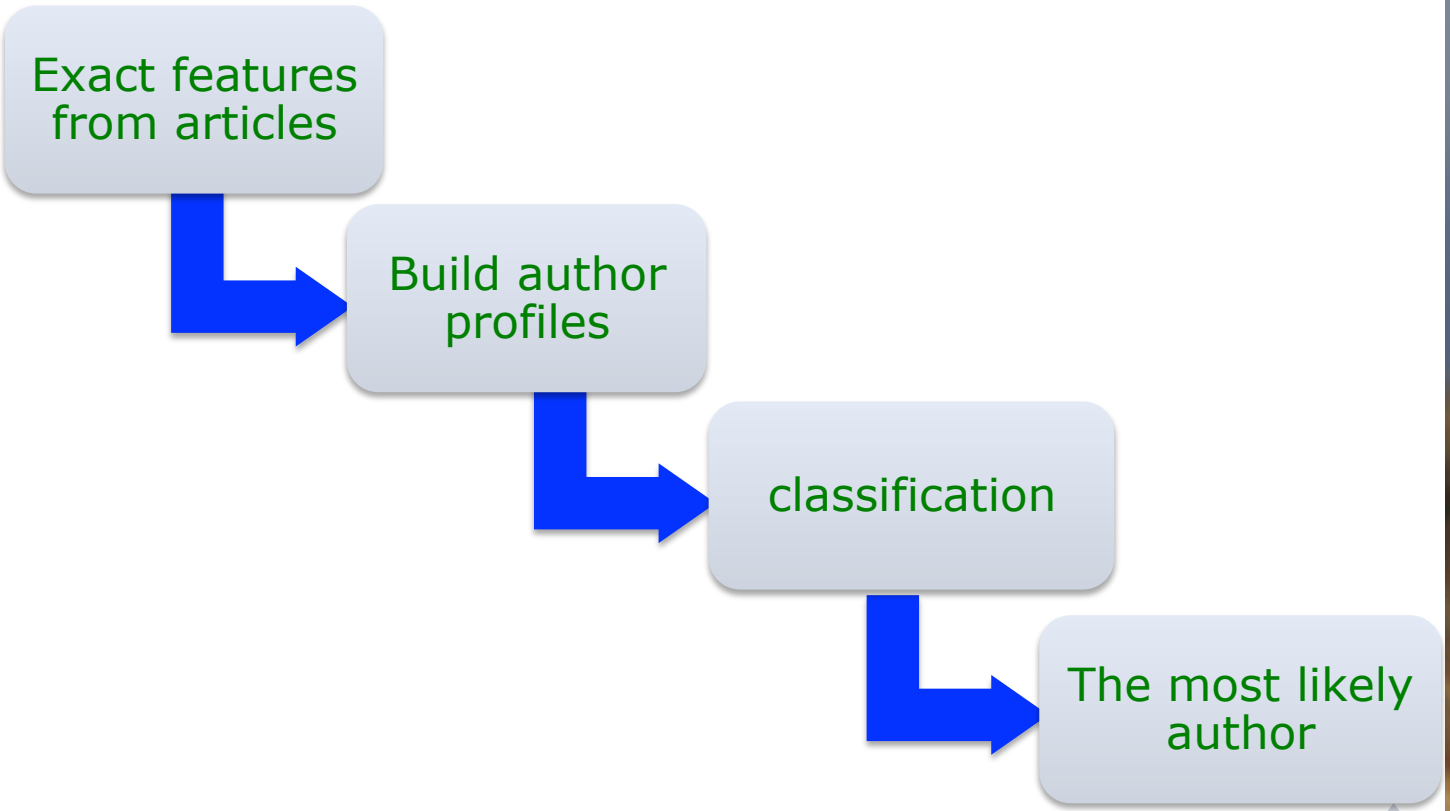
Past Research

Past Research

Syntactic
Characteristic Curve

Lexical
Most Frequent Function Words
Partial Matching Compression
Word Recurrence Interval
Trigram Markov Model

Approach for Authorship Attribution



THE UNIVERSITY of ADELAIDE

Project Progress

THE UNIVERSITY of ADELAIDE

Proposed methods

- Extraction Algorithms:
  - Maximal Frequent Word Sequence
  - Common N-gram

- Text Classifiers:
  - Naïve Bayes Classifier (Probability calculation)
  - Support Vector Machine (Machine self-decision)
  - Dissimilarity Calculation (Content comparison)

THE UNIVERSITY of ADELAIDE

Maximal Frequent Word Sequence & Naïve Bayes Classifier

Maximal Frequent Word Sequences

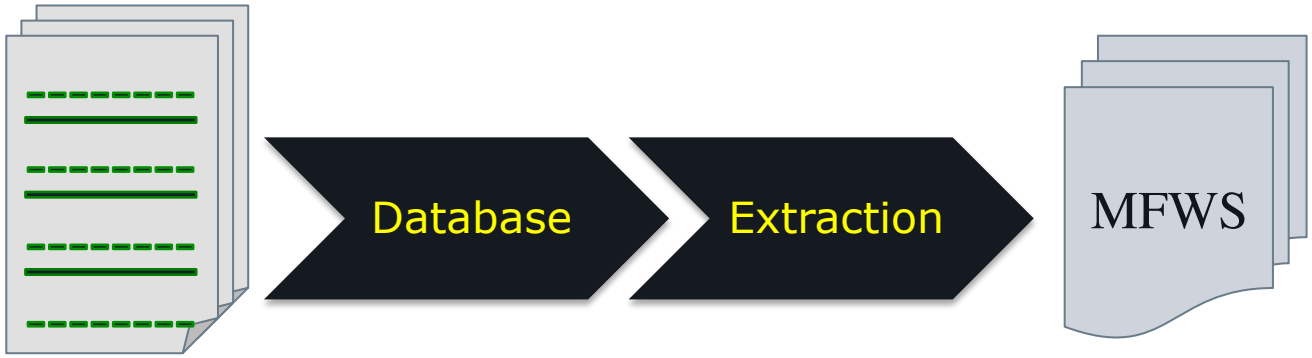
D is a set of texts, $D=\{d_1,d_2,d_3,\ldots,d_k\}$.

- A word sequence $s=\{w_1,w_2,w_3,\ldots\}$ is frequent if s can be found in at least *f* texts of the set D, where *f* is the given frequency threshold ($f \leq k$).

- The sequence s is a maximal frequent word sequence if there is no other frequent sequence S such that $s \subseteq S$.

Monday Tuesday Wednesday Thursday Friday Saturday Sunday Jar
February March April May June July August September October Nove
December Monday Tuesday Wednesday Thursday Friday Saturday Su
January February March April May June July August September Oc
November December Monday Tuesday Wednesday Thursday Friday Satu
Sunday January February March April May June July August September Oc
November December Monday Tuesday Wednesday Thursday Friday Satu
Sunday January February March April May June July August September Octo

Maximal Frequent Word Sequence Mining



Preprocessed Texts

Steps:

1. Preprocess input texts.
2. Construct a database from the texts.
3. Apply Extraction Algorithm.

A simple example...

Two input texts, find the MFWS with a threshold *f*=2.

T1=He knows I know you.

T2=I know you know him!

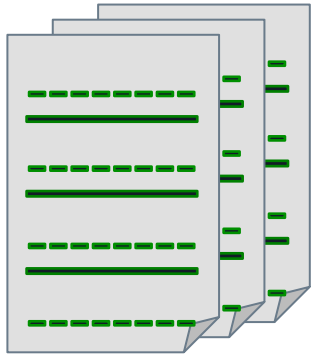
MFWS ?

Step 1. Preprocess input text

- Remove all symbols and punctuations.
- Remove duplicated space.
- Change Uppercase to lowercase.

The input texts become:

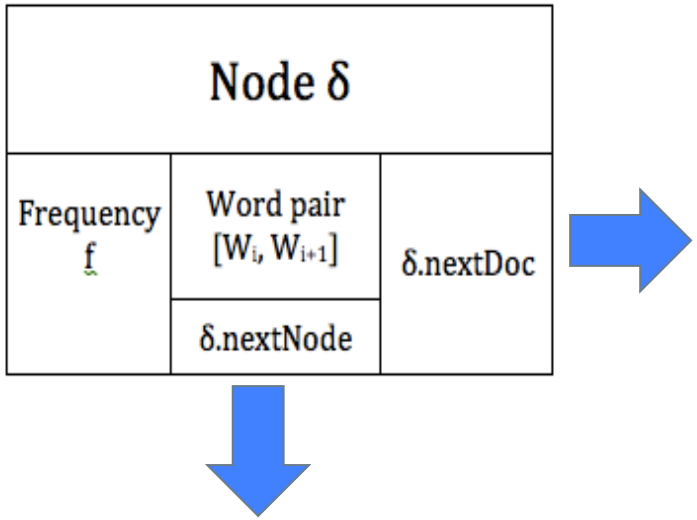
T1=He knows I know you.
T2=I know you know him!



THE UNIVERSITY of ADELAIDE

Step 2. Construct the database

- The database is a list of Nodes.
- A Node contains
  - A contiguous word sequence $[W_i, W_{i+1}]$
  - Links connected to other Nodes
  - Frequency f



Step 2. Construct the database (Cont'd)

Input text: t1= he knows i know you

The database looks like this:

Index	$[W_i, W_{i+1}]$	f	Link
1	he knows	1	
2	knows i	1	
3	i know	1	
4	know you	1	

t1

Current Index
1
2
3
4

Step 2. Construct the database (Cont'd)

Input text: t2= i know you know him

The database will look like this:

Index	$[W_i,W_{i+1}]$	f	Link
1	he knows	1	
2	knows i	1	
3	i know	2	
4	know you	2	
5	know him	1	

t1

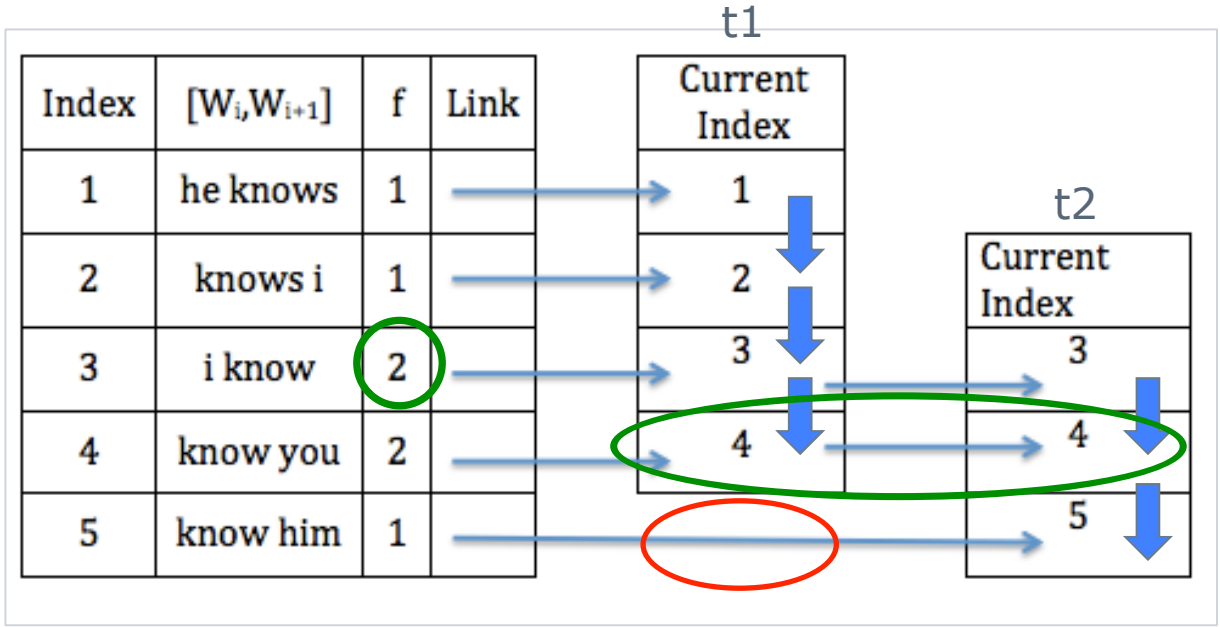
Current Index
1
2
3
4

t2

Current Index
3
4
5

Step 3: Extract MFWS

- Scan through the list
- Find frequent sequences that can grow



End of the example.

Two texts:

T1=he knows i know you

T2=i know you know him

The MFWS for threshold f=2 is:

- i know you

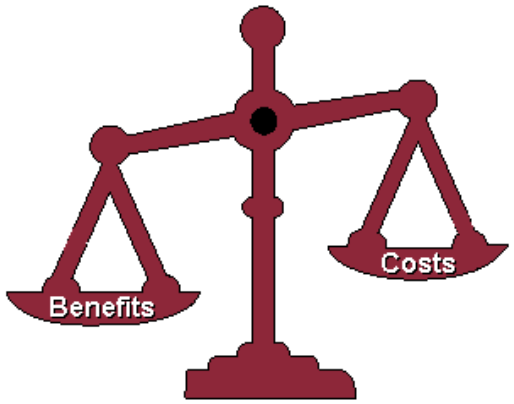
Reasons to choose Maximal Frequent Word Sequences

Combination of functional and content words

- Functional words have been approved to be an sufficient style marker. (Hilton; Mosteller & Wallace)
- Content words might contain unique meanings.
- Writers tend to use particular collocations of words.

Pattern extraction

- Language independent.
- No grammar constraints.



Limitation of Maximal Frequent Word Sequences

Size of the document collection

- Unbalanced number of documents
- Insufficient number of documents

e.g.

In New Testament, Paul has 14 articles whereas Luke only has 2.

Length of text

- Effect on the number of functional words

THE UNIVERSITY of ADELAIDE

A different approach...

Find the maximal frequent word sequence within one article.

A Trade-off:

✔ Potentially solve the problem of insufficient number of documents.

? Text length has larger impact on the result.

? Extracted MFWS might lose its characteristic.

THE UNIVERSITY of ADELAIDE

Naïve Bayes Classifier

Introduction to Naïve Bayes Classifier

The possibility of a document *d* belonging to a category c_i given *d* has a set of features $F=\{f_1,f_2,f_3,\ldots,f_k\}$ is:

$$P(c_i \mid d) = \frac{P(d \mid c_i)P(c_i)}{P(d)}$$

To simplify the equation, we assume that all the features of F are independent given c_i,

$$P(c_i \mid d) = P(c_i)\prod_{j=1}^{|F|} P(f_j \mid c_i)$$

Where

$$P(c_i) = \frac{N_i}{N} \quad \& \quad P(f_j \mid c_i) = \frac{1+N_{ji}}{|F| + \sum_{k=1}^{|F|} N_{ki}}$$

Introduction to Naïve Bayes Classifier (Cont'd)

$$P(c_i) = \frac{N_i}{N} \qquad P(f_j \mid c_i) = \frac{1 + N_{ji}}{|F| + \sum_{k=1}^{|F|} N_{ki}}$$

- N_i is the number of documents in category c_i
- N is the number of documents in the whole collection
- N_{ji} is the number of documents from c_i having feature f_j.
- $|F|$ is the number of all features.
- 1 is Laplace add one smoothing.
  - Deal with Zero probability

Reasons to use Naïve Bayes classifier

- Naive Bayes classifier has been proven successful in text classification.
  - Rosa María 2006, Lewis 1998, McCallum and Nigam 1998, Domingos and Pazzanni (1997)

- Simple to implement.

Algorithm Testing

Data Base – English fictions

A total of 132 articles, from 6 authors, 22 articles per author.

Author	Average text length per article
Sir Arthur Conan Doyle	6680 words
B. M. Bower	6355 words
Charles Dickens	7033 words
Henry James	7945 words
Richard H. Davis	6396 words
Zane Grey	6517 words

Algorithm Testing

Test 1

5 articles per author (training)

2 articles per author (Testing)

Test 2

10 articles per author (training)

2 articles per author (Testing)

Test Results for MFWS

Test 1:

Threshold	MWFS	Average words per sequence	Accuracy
2	2795	2.59	66.7%
3	2460	2.33	75.0%
4	2112	2.17	66.7%
5	1170	1.78	58.3%

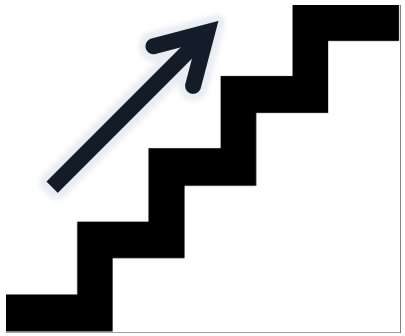
Test 2:

Threshold	MWFS	Average words per sequence	Accuracy
2	5017	2.63	66.67%
3	4653	2.37	66.67%
4	3139	2.21	75.0%
5	2268	1.90	75.0%

THE UNIVERSITY of ADELAIDE

Future development

- Reduce the impact of threshold f.
  - Combine features from different threshold f.
- Further tests and analyses on the 132 English fictions, Federalist paper and Greek New Testament.
- Compare results with other algorithms



Common N-gram & Support Vector Machine

Common N-gram (CNG)

- Contiguous sequence of n items from a given sequence of text
- Apply to language text by extraction of n-gram
- A n-gram of size 1 is referred to as a "unigram",
  size 2 is a "bigram",
  size 3 is a "trigram",
  size 4 is a "fourgram"…
- Byte Level & Language independence

Common N-gram (CNG)

- An example to demonstrate

  Consider "HELLO WORLD", by varying the value n ($2 \leq n \leq 5$)

N-gram Size			
N=2	N=3	N=4	N=5
HE	HEL	HELL	HELLO
EL	ELL	ELLO	ELLO_
LL	LLO	LLO_	LLO_W
LO	LO_	LO_W	LO_WO
O_	O_W	O_WO	O_WOR
_W	_WO	_WOR	_WORL
WO	WOR	WORL	WORLD
OR	ORL	ORLD	
RL	RLD		
LD			

Common N-gram Programming

- Purpose
- Java
- Core Methods: removePunctuation, removeDuplicateSpace, convertToLowerCase, processString
- Limitation (Length, input format, computational expensive)

Computational Expensive			
	132 English Text (6 authors, Each author has 22 books)	Federalist Paper (86 books)	Greek New Testament (27 books)
Running Time	6 hours	1hour 12mins	2hour 30mins

An example for extracting bigram (132 English Text)

The input file:

In the third week of November, in the year 1895, a dense yellow fog settled down upon London. From the Monday to the Thursday I doubt whether it was ever possible from our windows in Baker Street to see the loom of the opposite houses. The first day Holmes had spent in cross-indexing his huge book of references. The second and third had been patiently occupied upon a subject which he hand recently made his hobby–the music of the Middle Ages. But when, for the fourth time, after pushing back our chairs from breakfast we saw the greasy, heavy brown swirl still drifting past us and condensing in oily drops upon the window- panes, my comrade's impatient and active nature could endure this drab existence no longer. He paced restlessly about our sitting room in a fever of suppressed energy, biting his nails, tapping the furniture, and chafing against inaction.

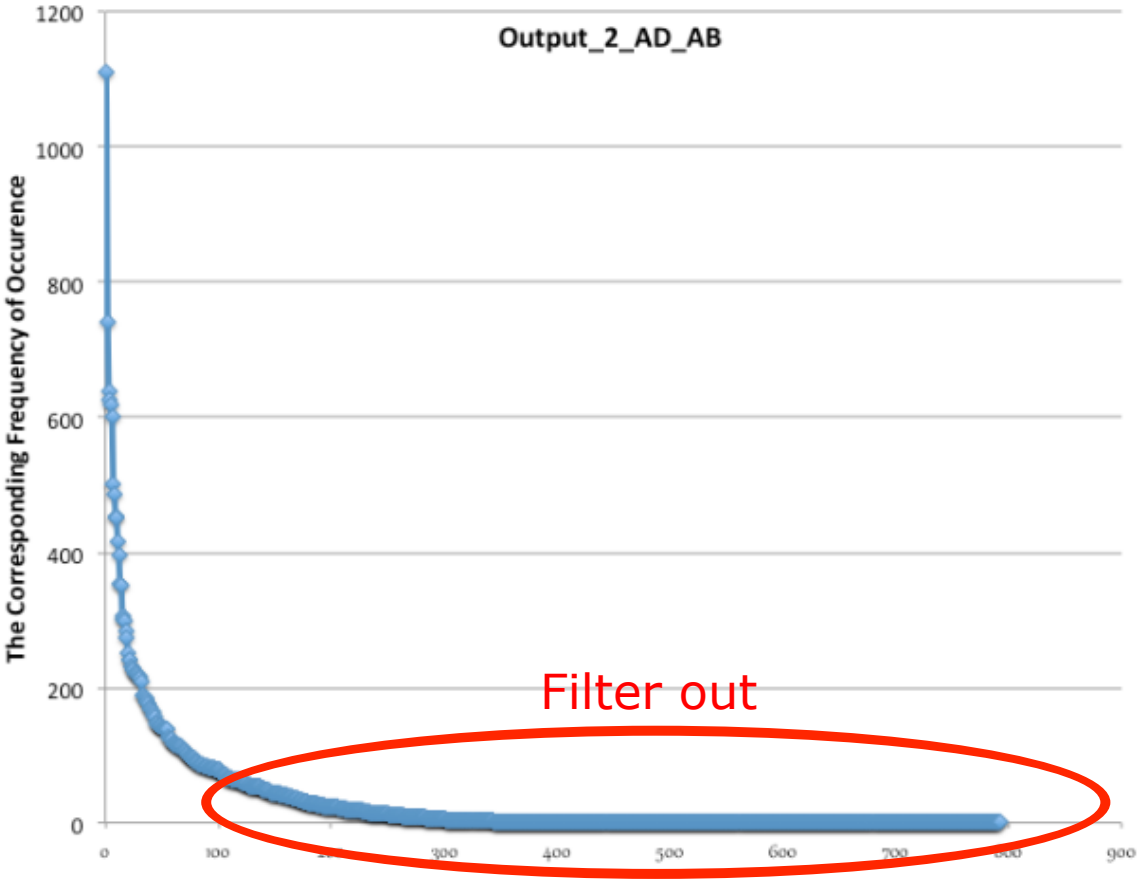
The output file:

N = 2	
e_	1112
_t	742
s_	638
th	626
he	619
t_	601
_a	501
d_	487
n_	455
_h	451
in	416
_w	396
er	354
_s	352
_o	307

THE UNIVERSITY of ADELAIDE

Results for CNG (132 English Text)

- Plot the most frequent and its occurrence



Support Vector Machine

Introduction to Support Vector Machine (SVM)

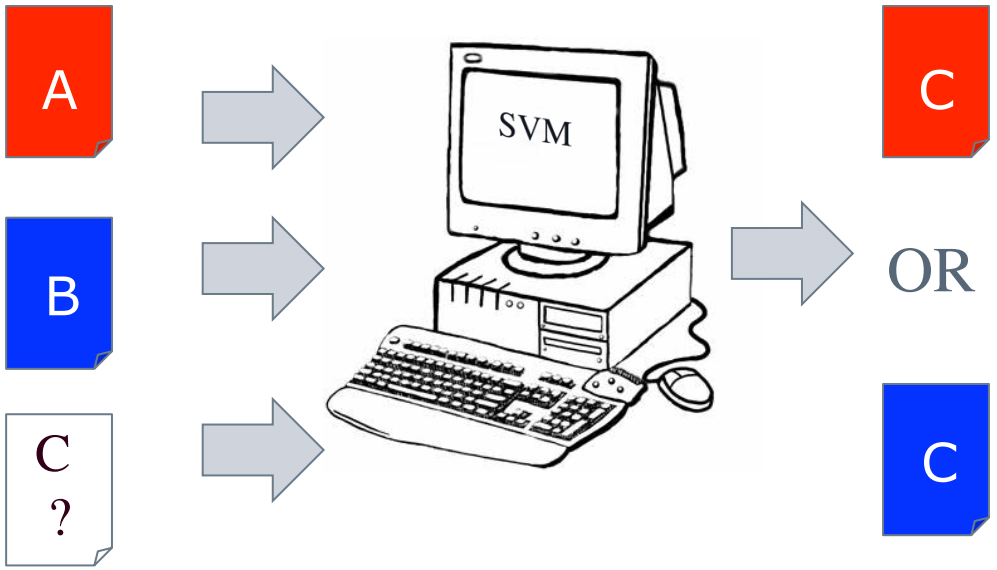
- A promising tool for data classification
- Perform accurate result
- Easy implement by using Matlab
- Classification Kernel Function:

  -- Linear Kernel Function

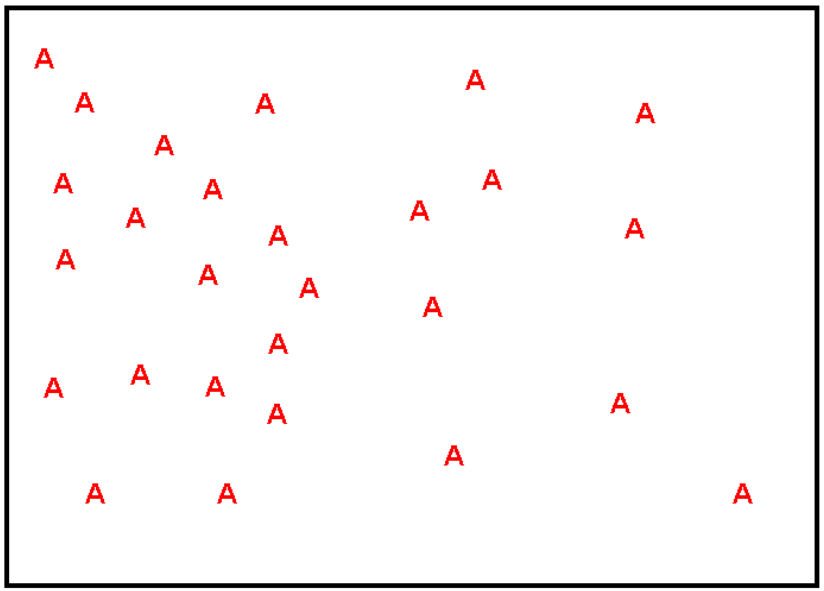
THE UNIVERSITY of ADELAIDE

How SVM works?

- Training Input: two different author profile: A & B (from CNG)
- Testing Input: one disputed author profile: C (from CNG)
- Output: C belongs to A, or
  C belongs to B



How SVM works? (Cont'd)



Input: A

Test Results for Common N-gram

Test 1:

5 articles per author for training 2 articles per author for testing			
N-gram	Threshold f	Common N-gram feature	Accuracy
2	15	237	41.67%
3	15	457	50.0%
4	15	300	66.67%

Test 2:

10 articles per author for training 2 articles per author for testing			
N-gram	Threshold f	Common N-gram feature	Accuracy
2	15	458	50.0%
3	15	791	66.67%
4	15	652	75.0%

THE UNIVERSITY of ADELAIDE

Future improvement

- Complete the analysis of 132 English Text, Federalist paper and Greek new testament
- Examine text lengths from training and testing data
- Explore the effects of combination of extracted features
- Compare results with Maximal Frequent Word Sequences
- Compare results with past algorithms

THE UNIVERSITY of ADELAIDE

Common N-gram & Dissimilarity Classifier

Dissimilarity Calculation

- What is Dissimilarity Calculation:
  - To determine the pairwise difference between samples.
- Why choose it:
  - Widely used
  - Simple to implement
- Applications
  - Used to find the Genetic Dissimilarity among genotype.

Example – Step 1: Arranging Features Extraction

t1

Feature	Occurrence
a_	164
ac	94
ad	154
ai	97
al	147
am	58
an	357
ar	261
as	252
at	335
av	88

t2

Feature	Occurrence
a_	121
ac	54
ad	97
ai	106
al	140
an	299
ar	143
as	186
at	226
av	84

Example – Step 2: Construct dissimilarity matrix

Define dissimilarity matrix:

$A = \{ (X_{1,} f_{1A}), (X_2, f_{2A}),\ldots(X_{n,} f_{nA}) \};$

$B = \{ (X_{1,} f_{1B}), (X_2, f_{2B}),\ldots(X_{n,} f_{nB}) \}$

f_{nA} and f_{nB} is occurrence of feature X_n from text A and B

Dissimilarity matrix for text t1 and t2:

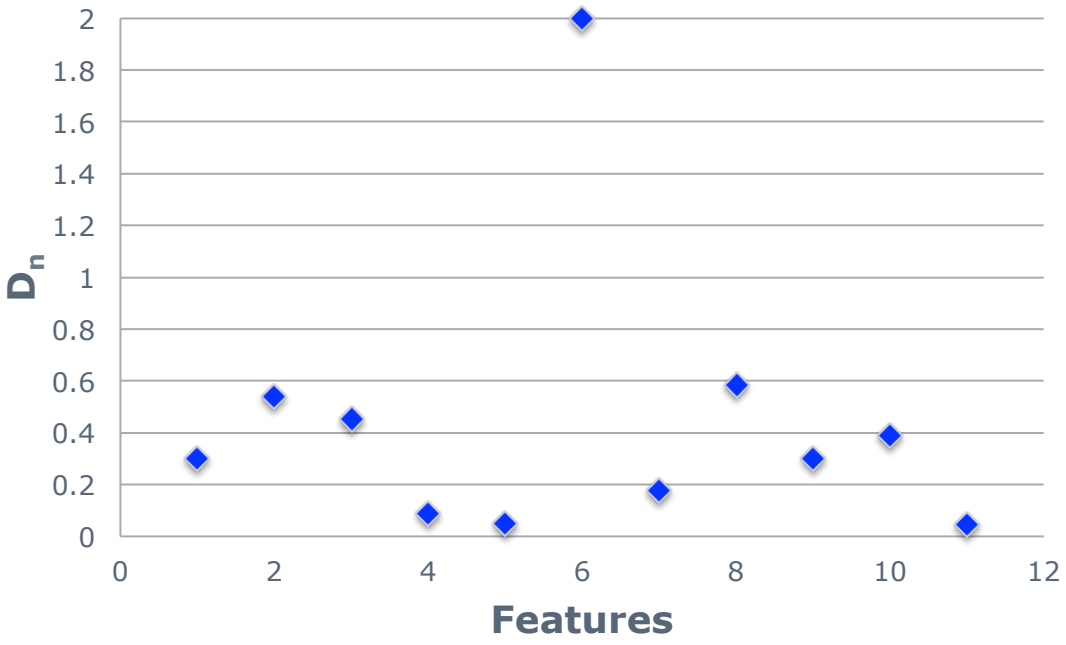
M_t1 = { (a_, 121), (ac, 54),…(av, 84) }

M_t2= { (a_, 164), (ac, 94),…(av, 88) }

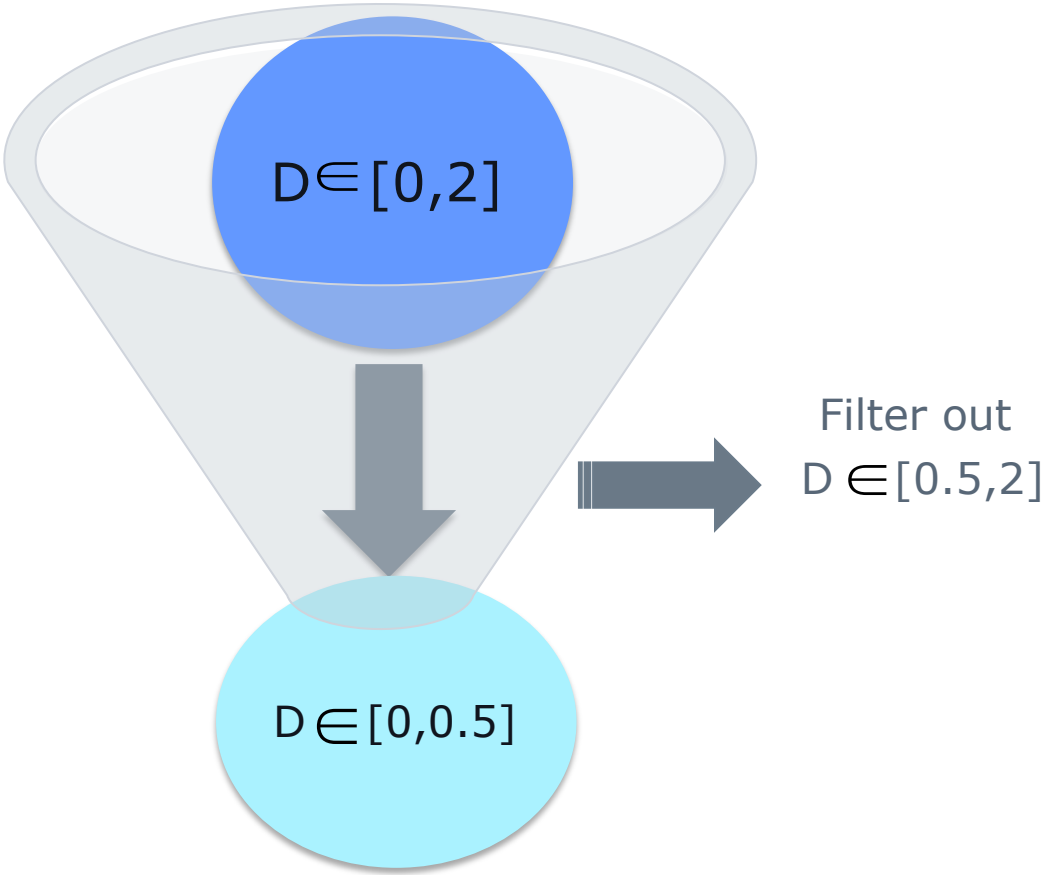
Example – Step 3: Dissimilarity Calculation

Mathematical Formula:

$$D_n = \frac{| f_{nA} - f_{nB} |}{\frac{f_{nA} + f_{nB}}{2}}$$



Example – Step 4: Evaluation D_n



Testing Results for dissimilarity classifier

Test 1:

N-gram	Common N-gram feature	Threshold D_n	Accuracy
2	237	0.5	41.67%
3	457	0.5	50.0%
4	300	0.5	66.67%

Test 2:

N-gram	Common N-gram feature	Threshold D_n	Accuracy
2	458	0.5	50.0%
3	791	0.5	66.67%
4	652	0.5	75.0%

THE UNIVERSITY of ADELAIDE

Further Improvement

- Complete the testing the 132 English fictions, Federalist paper and Greek new testament.
- Examine the effects with different threshold.
- Compare results with other algorithms.

THE UNIVERSITY of ADELAIDE

Work Management

THE UNIVERSITY of ADELAIDE

Content

- Project Schedule
- Role Allocation
- Risk Assessment
- Project Budget
- Deliverable
- Project's possible implications

Project Schedule

Events	Date	Status
Proposal Seminar	12th Aug 2011	✔
Stage 1 report	26th Aug 2011	✔
Stage 2 report	28th Oct 2011	✔
Exhibition Information	9th Mar 2012	✔
Final Seminar	5th Apr 2012	processing
Final report	25th May 2012	
Poster	29th May 2012	
Project Exhibition	1st Jun 2012	

THE UNIVERSITY of ADELAIDE

Role Allocation

	Yan Xie	Kai He	Zhaokun Wang
MFWS Programming		✔	
C-Ngram Programming	✔		
Meeting organizer	✔		
Group Leader		✔	
Document archive and management			✔
Classifiers Programming	✔	✔	✔
Progress report	✔	✔	✔
Meeting chair, secretary	✔	✔	✔
Document revision and formatting	✔	✔	✔

Risk Assessment

Risk	Priority	Probability Rating	Impact Rating
Changed schedule of critical events	32	4	8
Data Lost	15	3	5

Preventive Measures

- (1) Begin tasks ahead
  (2) Regularly review the project schedule

- (1) Use iCloud
  (2) Send every group members a copy of work
  (3) One group member is in charge of document archive.

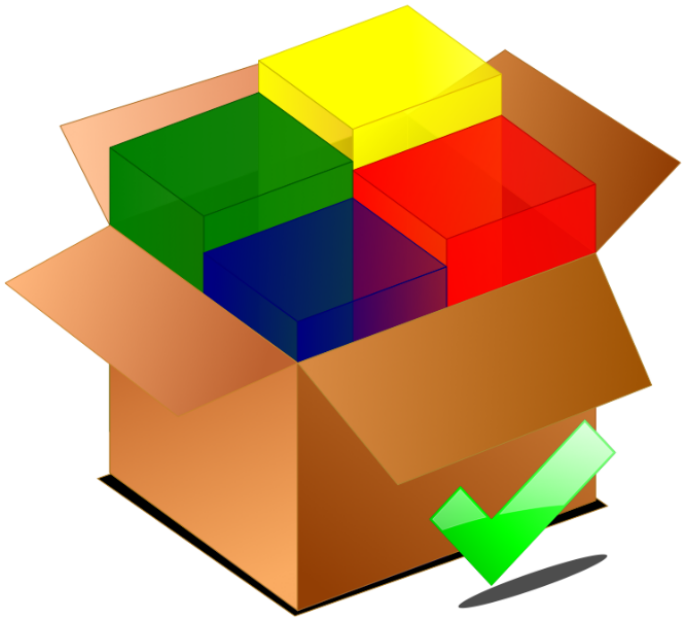
Project Budget

- Allocated Total Budget: $750
  ----- $250 per team member
- Expected Expenses: $50
  ----- Poster: $50
- Actual Expenses so far: $0

THE UNIVERSITY of ADELAIDE

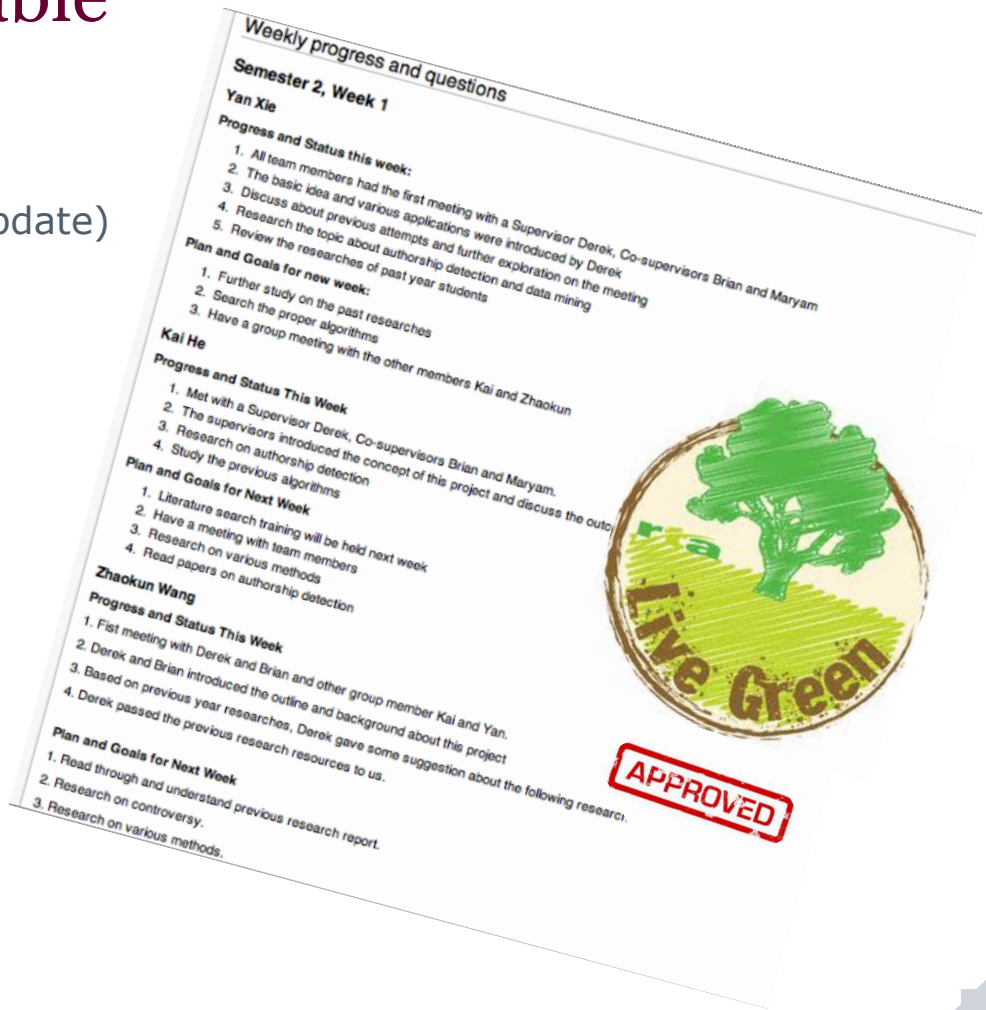
Deliverable

- Documents:
  --Proposal Seminar Power Point Slide
  --Stage One Group Report
  --Stage Two Individual Report
  --Group meeting minutes
  --Software updates log
- Wiki page
- Youtube Video
- Poster



Deliverable

- Wiki
  (Weekly update)



Weekly progress and questions

Semester 2, Week 1

Yan Xie

Progress and Status this week:

1. All team members had the first meeting with a Supervisor Derek, Co-supervisors Brian and Maryam
2. The basic idea and various applications were introduced by Derek
3. Discuss about previous attempts and further exploration on the meeting
4. Research the topic about authorship detection and data mining
5. Review the researches of past year students

Plan and Goals for new week:

1. Further study on the past researches
2. Search the proper algorithms
3. Have a group meeting with the other members Kai and Zhaokun

Kai He

Progress and Status This Week

1. Met with a Supervisor Derek, Co-supervisors Brian and Maryam.
2. The supervisors introduced the concept of this project and discuss the outc
3. Research on authorship detection
4. Study the previous algorithms

Plan and Goals for Next Week

1. Literature search training will be held next week
2. Have a meeting with team members
3. Research on various methods
4. Read papers on authorship detection

Zhaokun Wang

Progress and Status This Week

1. Fist meeting with Derek and Brian and other group member Kai and Yan.
2. Derek and Brian introduced the outline and background about this project
3. Based on previous year researches, Derek gave some suggestion about the following researc
4. Derek passed the previous research resources to us.

Plan and Goals for Next Week

1. Read through and understand previous research report.
2. Research on controversy.
3. Research on various methods.

Deliverable

- Youtube Video Presentation 2011:

  Who wrote the Letter to the Hebrews?

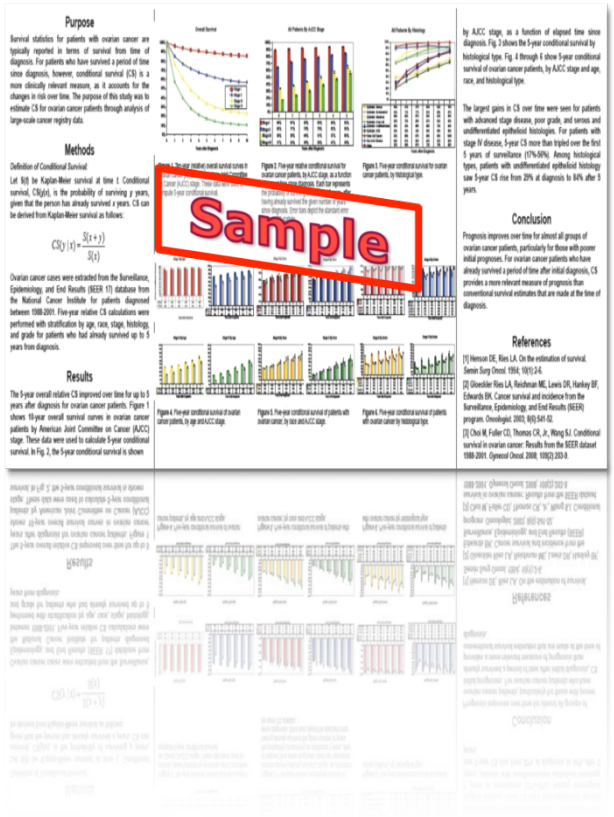
Image Reference: http://www.googleplay.cc/wp-content/uploads/2012/03/youtube-600x347.png

Deliverable

- Poster:

  --Project description

  --Image



THE UNIVERSITY of ADELAIDE

Project's possible implications

- Ethical
  --Automatic student plagiarism detection
  --Duplicate Publication
- Social
  --Assessing the credibility of content on the web
  --Measuring quality of web content
- Cultural
  --Widely use in the other language, such as, Chinese, Indian…

THE UNIVERSITY of ADELAIDE

Conclusion

Reference

- Baayen, H., Halteren, H. V., Neijt, A. & Tweedie, F., An experiment in authorship attribution, 6th JADT, 2002
- Eddy H. T., The Characteristic Curves of Composition, <http://www.jstor.org/stable/1763509>, viewed August 2010
- Holmes, D. I., & Forsyth, R. S., The 'Federalist' Revisited: New Directions in Authorship Attribution, Literary and Linguistic Computing, 10, 111-127, 1995
- Juola, P. & Baayen, H., A controlled corpus experiment in authorship attribution by crossentropy, Proceedings of ACH/ ALLC- 2003, 2003
- Putnins, T. J., Signoriello, D. J., Jain, S. Berryman, M. J., & Abbott, D., Who wrote the Letter to the Hebrews? Data mining for detection of text authorship, University of Adelaide, 2005
- Smith, M. W. A., Recent experience and new developments of methods for the determination of authorship, ALLC Bulletin, 11:73–82, 1983.

THE UNIVERSITY of ADELAIDE

Reference

- Stamatatos, E., Fakotakis, N. & Kokkinakis, G., Automatic Text Categorization in Terms of Genre and Author, Computational Linguistics, vol. 26, no. 4, pp. 471-495(25), December 2000
- Sabordo,M., Shong, C. Y., Berryman, M. J. & Abbott, D., Who Wrote the Letter to the Hebrews? – Data Mining for Detection of Text Authorship, SPIE vol. 5649 pp. 513 – 524, 2004
- Smith, M. W. A., Recent experience and new developments of methods for the determination of authorship, ALLC Bulletin, 11:73–82, 1983
- Weston, J, S Mukherjee, O. Chapelle, M. Pontil, T. Poggio & V. Vapnik. "Feature Selection for SVMs" Barnhill BioInformatics Savannah, Georgia USA 2007

THE UNIVERSITY of ADELAIDE

Thank you !

THE UNIVERSITY of ADELAIDE

Questions?