Human Annotation and Automatic Detection of Web Genres

by

Noushin Rezapour Asheghi

Submitted in accordance with the requirements for the degree of Doctor of Philosophy.

UNIVERSITY OF LEEDS

The University of Leeds
School of Computing

January 2015

The candidate confirms that the work submitted is her own and that the appropriate credit has been given where reference has been made to the work of others.

Declarations

The candidate confirms that the work submitted is her own, except where work which has formed part of jointly authored publications has been included. The contribution of the candidate and the other authors to this work has been explicitly indicated below. The candidate confirms that appropriate credit has been given within the thesis where reference has been made to the work of others.

- **Publication:** Noushin Rezapour Asheghi, Serge Sharoff, and Katja Markert. 2014. Designing and evaluating a reliable corpus of web genres via crowd-sourcing. In Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC14).
  My contributions: Principal author, conducted all the experiments and interpreted the results.
  Other authors contributions: Serge Sharoff, and Katja Markert provided supervision, feedback, general guidance and contributed to paper write-up.
  Chapters based on this work: Parts of Chapter 3.

- **Publication:** Noushin Rezapour Asheghi, Katja Markert and Serge Sharoff. 2014.Semi-supervised Graph-based Genre Classification for Web Pages. TextGraphs-9: 39.
  My contributions: Principal author, conducted all the experiments, interpreted the results and wrote the paper.
  Other authors contributions: Serge Sharoff, and Katja Markert provided supervision, feedback and general guidance.
  Chapters based on this work: Chapter 4 and 5.

Acknowledgements

Doing a PhD is like a journey with many hurdles along the way. This journey would not be completed without the help of many individuals. I would like to take this opportunity to acknowledge anyone who has helped me in this journey. First of all, I would like to express my deepest gratitude and appreciation to my supervisors, Dr. Katja Markert, Dr. Serge Sharoff and Dr. Brandon Bennett for their insightful advice, expert guidance and encouragement at all levels.

I am also very grateful for the friendship of all the members in my research group, especially Josiah Wang, Andrew McKinlay, Amal Alsaif, Saman Hina, Samuel Danso and Alicja Piotrkowicz. I immensely enjoyed the discussions that we had in the lab over the past four years.

I would also like to thank my husband, Sayyad. The generation of this thesis would not have been possible without his continuous support and encouragement. Finally, and most importantly, I am very grateful to my parents for their unconditional love and support throughout my life. Thank you both for providing me the opportunity for a great education.

Abstract

Texts differ from each other in various dimensions such as topic, sentiment, authorship and genre. In this thesis, the dimension of text variation of interest is genre. Unlike topic classification, genre classification focuses on the functional purpose of documents and classifies them into categories such as news, review, online shop, personal home page and conversational forum. In other words, genre classification allows the identification of documents that are similar in terms of purpose, even they are topically very diverse.

Research on web genres has been motivated by the idea that finding information on the web can be made easier and more effective by automatic classification techniques that differentiate among web documents with respect to their genres. Following this idea, during the past two decades, researchers have investigated the performance of various genre classification algorithms in order to enhance search engines. Therefore, current web automatic genre identification research has resulted in several genre annotated web-corpora as well as a variety of supervised machine learning algorithms on these corpora.

However, previous research suffers from shortcomings in corpus collection and annotation (in particular, low human reliability in genre annotation), which then makes the supervised machine learning results hard to assess and compare to each other as no reliable benchmarks exist. This thesis addresses this shortcoming. First, we built the Leeds Web Genre Corpus Balanced-design (LWGC-B) which is the first reliably annotated corpus for web genres, using crowd-sourcing for genre annotation. This corpus which was compiled by focused search method, overcomes the drawbacks of previous genre annotation efforts such as low intercoder agreement and false correlation between genre and topic classes.

Second, we use this corpus as a benchmark to determine the best features for closed-set supervised machine learning of web genres. Third, we enhance the prevailing supervised machine learning paradigm by using semi-supervised graph-based approaches that make use of the graph-structure of the web to improve classification results. Forth, we expanded our annotation method successfully to Leeds Web Genre Corpus Random (LWGC-R) where the pages to be annotated are collected randomly by querying search engines. This randomly collected corpus also allowed us to investigate coverage of the underlying genre inventory. The result shows that our 15 genre categories are sufficient to cover the majority but not the vast majority of the random web pages.

The unique property of the LWGC-R corpus (i.e. having web pages that do not belong to any of the predefined genre classes which we refer to as noise) allowed us to, for the first time, evaluate the performance of an open-set genre classification algorithm on a dataset with noise. The outcome of this experiment indicates that automatic open-set genre classification is a much more challenging task compared to closed-set genre classification due to noise. The results also show that automatic detection of some genre classes is more robust to noise compared to other genre classes.

Contents

List of Figures

List of Tables

Abbreviations

The following table illustrates the abbreviations used throughout the thesis.

Abbreviation	Description
AGI	Automatic Genre Identification
BNC	British National Corpus
FGC	Functional Genre Classification
POS	Part-of-Speech
NLP	Natural Language Processing
Acc	Accuracy
LWGC	Leeds Web Genre Corpus
LWGC-B	Leeds Web Genre Corpus Balanced-design
LWGC-R	Leeds Web Genre Corpus Random
HGC	Hierarchical Genre Collection
MGC	Multi-labelled Genre Collection
URL	Uniform Resource Locators
SVM	Support Vector Machine
RFSE	Random Feature Sub-spacing Ensemble
NYTAC	New York Times Annotated Corpus
PHP	Personal Homepage
COM	Company/ Business homepage
EDU	Educational Organization Homepage
BIO	Biography
FAQ	Frequently Asked Questions
MTurk	Amazon Mechanical Turk
HITs	Human Intelligence Tasks
ICA	Iterative Classification Algorithm
FOIL	First Order Inductive Learner
MLNs	Markov Logic Networks
Edit	Editorials
LttE	Letters to the Editor
Med	Medicine
Def	Defence

Chapter 1

Introduction

Text classification has different branches such as topic classification, authorship recognition, sentiment analysis and genre identification. This thesis concentrates on Automatic Genre Identification (AGI). In AGI, documents are classified based on their genres rather than their topics or subjects. Genre classification attempts to answer the question: What is the goal or the purpose of the text? whereas the aim of topic classification is to answer the question: what is this text about? Genre classes such as editorial, interview and news which are recognizable by their distinct purposes can be on any topics.

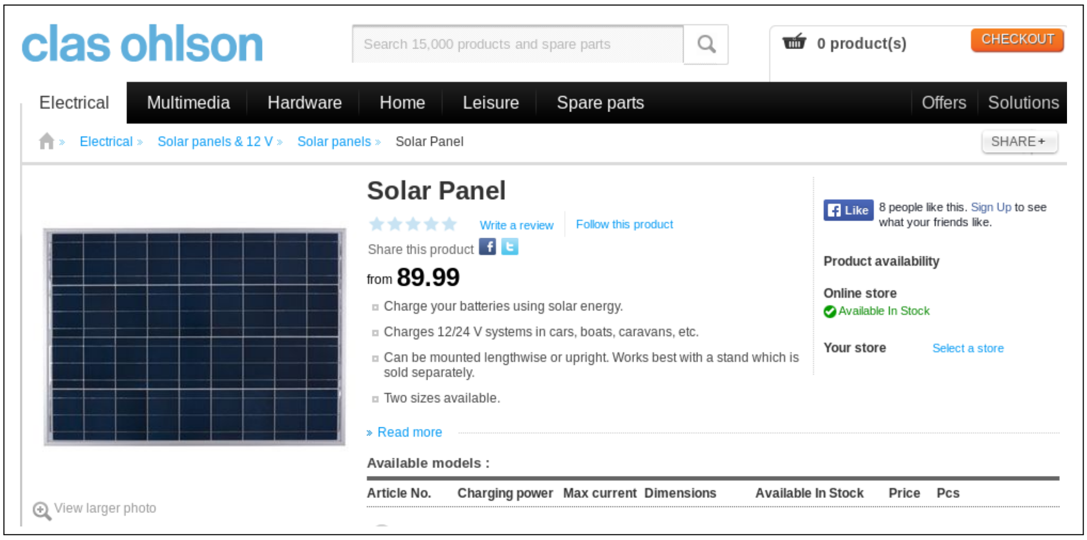
Writers vary their style, structure and vocabulary significantly depending on the genre of the document they write. For example, an interview, a news article and an editorial for a newspaper on the same topic will be presented very differently because they have different purposes. The purpose of news articles is to inform people and therefore they are written in an informative style, whereas, the editorials' main purpose is to express opinion and thus they are presented in an argumentative style. Meanwhile, the structure of interviews, which are usually in the form of face-to-face questions and answers, distinguishes them from both news articles and editorials. Automatic Genre Identification classifies documents into genres that encapsulate the main communicative function of a text.

1.1 Motivation

Automatic genre identification has the potential to be important for a number of applications. For example, AGI could improve Information Retrieval. Search engines such as Google retrieve web pages based on the queries composed of keywords. Since keywords to some extent indicate the subject of web pages, these results are mainly topical. The problem with this method is that it often cannot precisely retrieve the pages which match the users' needs. For example, querying a search engine with the term *solar panel* would result in retrieving web pages from various genres such as:

- On-line shops with the aim of selling this product (Figure 1.1)
- Conversation forums where people have an interactive conversation about this product (Figure 1.2)
- News about this product (Figure 1.3)
- Companies' home pages with the description of the services these companies provide regarding this product (Figure 1.4)
- Personal blogs where individuals share their experience regarding this product in their daily lives
- Academic papers which express deep specialized information about constructing and developing this product

However, if a user is only interested in reading reviews about *solar panels*, entering *solar panel review* as a query in a search engine could result in a list of web pages from any genre class, because the existence of these words in a web page does not guarantee that it contains a review. For instance, the words *review* and *solar panel* are present in the conversational forum web page in Figure 1.2. However, this page does not contain any review. Therefore, if a user could use the search engine to retrieve the web pages from a specific genre, the search results could be more beneficial. For example, specifying the genre as review in a genre-enabled search engine would give more precise results. In order to improve search engines, automatic genre identification of web pages has recently attracted significant interest [133].



Electrical » Solar panels & 12 V » Solar panels » Solar Panel

Solar Panel

Write a review | Follow this product

Share this product

from 89.99

- Charge your batteries using solar energy.
- Charges 12/24 V systems in cars, boats, caravans, etc.
- Can be mounted lengthwise or upright. Works best with a stand which is sold separately.
- Two sizes available.

» Read more

8 people like this. Sign Up to see what your friends like.

Product availability

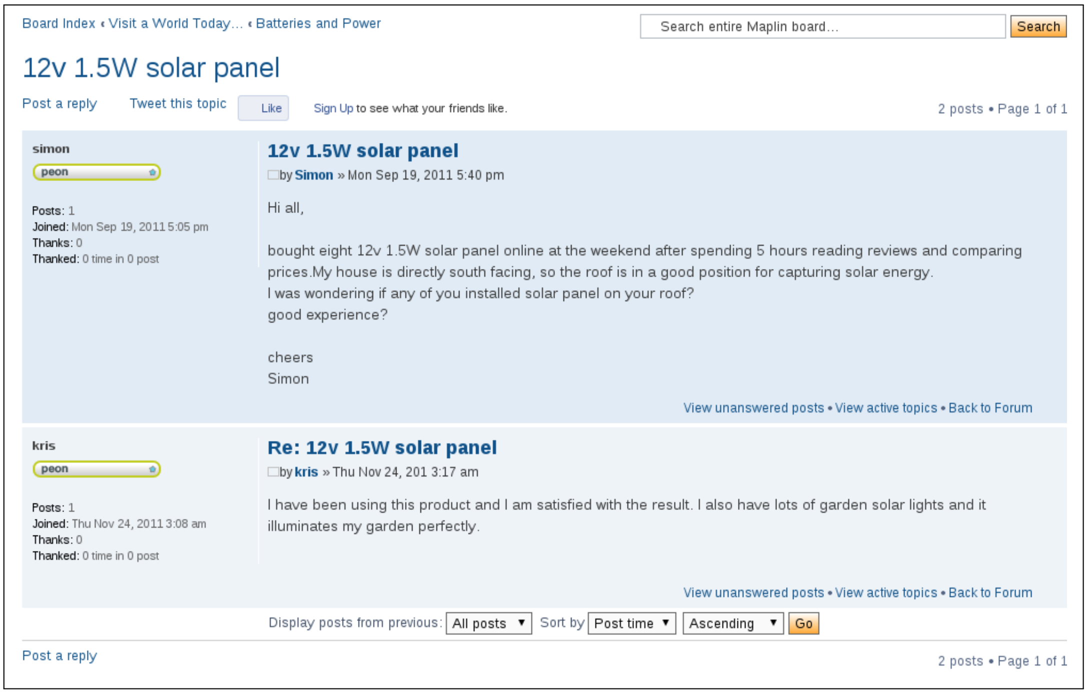
Online store
Available In Stock

Your store Select a store

Available models :

Article No.	Charging power	Max current	Dimensions	Available In Stock	Price	Pcs

Figure 1.1: Part of a shop web page retrieved with the query “solar panel”. URL: http://www.clasohlson.com/uk/Solar-Panel/36-4451?LGWCODE=364451000;83827;4797&gclid=CPObnY3RwsACFULmwgodkWoAiQ



Board Index ‹ Visit a World Today… ‹ Batteries and Power

12v 1.5W solar panel

Post a reply Tweet this topic Sign Up to see what your friends like. 2 posts • Page 1 of 1

simon

Posts: 1
Joined: Mon Sep 19, 2011 5:05 pm
Thanks: 0
Thanked: 0 time in 0 post

12v 1.5W solar panel

by Simon » Mon Sep 19, 2011 5:40 pm

Hi all,

bought eight 12v 1.5W solar panel online at the weekend after spending 5 hours reading reviews and comparing prices.My house is directly south facing, so the roof is in a good position for capturing solar energy.
I was wondering if any of you installed solar panel on your roof?
good experience?

cheers
Simon

View unanswered posts • View active topics • Back to Forum

kris

Posts: 1
Joined: Thu Nov 24, 2011 3:08 am
Thanks: 0
Thanked: 0 time in 0 post

Re: 12v 1.5W solar panel

by kris » Thu Nov 24, 201 3:17 am

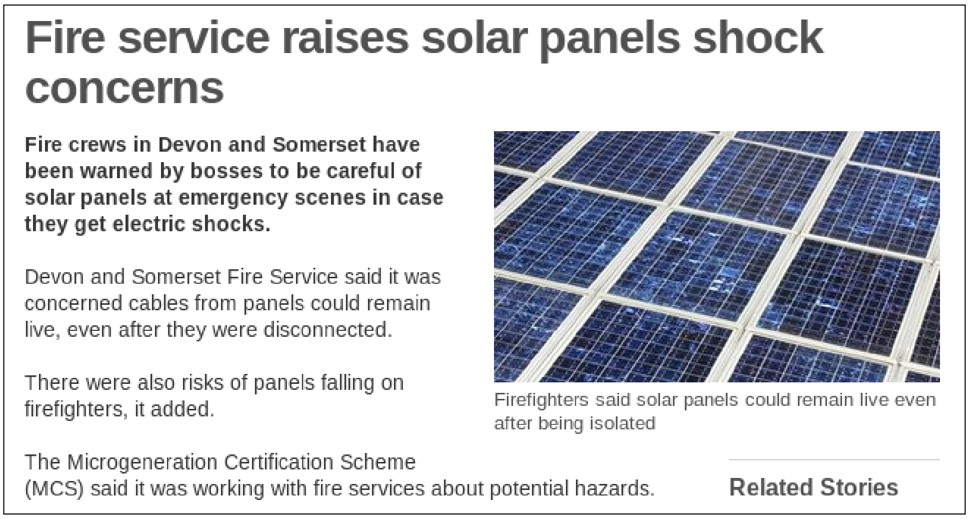
I have been using this product and I am satisfied with the result. I also have lots of garden solar lights and it illuminates my garden perfectly.

View unanswered posts • View active topics • Back to Forum

Display posts from previous: All posts Sort by Post time Ascending Go

Post a reply 2 posts • Page 1 of 1

Figure 1.2: Part of a conversation forum web page retrieved with the query “solar panel”. URL: http://www.solarpanels.com/showthread.php?12983-Question



Fire service raises solar panels shock concerns

Fire crews in Devon and Somerset have been warned by bosses to be careful of solar panels at emergency scenes in case they get electric shocks.

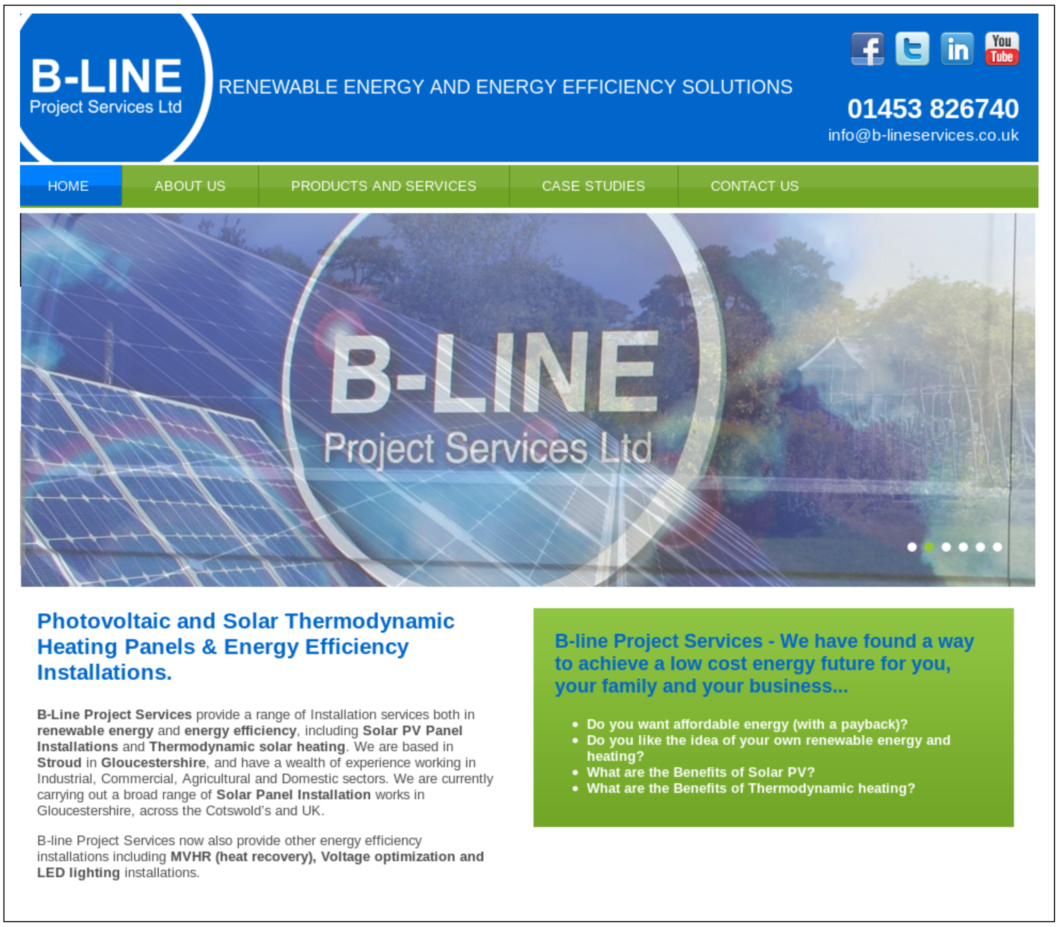
Devon and Somerset Fire Service said it was concerned cables from panels could remain live, even after they were disconnected.

There were also risks of panels falling on firefighters, it added.

Firefighters said solar panels could remain live even after being isolated

The Microgeneration Certification Scheme (MCS) said it was working with fire services about potential hazards.

Figure 1.3: Part of a News web page retrieved with the query "solar panel". URL: http://www.mcsdirectory.co.uk/fire-service-raises-solar-panels-shock-concerns/



Photovoltaic and Solar Thermodynamic Heating Panels & Energy Efficiency Installations.

B-Line Project Services provide a range of Installation services both in **renewable energy** and **energy efficiency**, including **Solar PV Panel Installations** and **Thermodynamic solar heating**. We are based in **Stroud** in **Gloucestershire**, and have a wealth of experience working in Industrial, Commercial, Agricultural and Domestic sectors. We are currently carrying out a broad range of **Solar Panel Installation** works in Gloucestershire, across the Cotswold's and UK.

B-line Project Services now also provide other energy efficiency installations including **MVHR (heat recovery), Voltage optimization and LED lighting** installations.

B-line Project Services - We have found a way to achieve a low cost energy future for you, your family and your business...

- **Do you want affordable energy (with a payback)?**
- **Do you like the idea of your own renewable energy and heating?**
- **What are the Benefits of Solar PV?**
- **What are the Benefits of Thermodynamic heating?**

Figure 1.4: Part of a company homepage retrieved with the query "solar panel". URL: http://www.solarpowergloucestershire.co.uk/

Detecting the genre of a discourse could also be beneficial in other areas of Natural Language Processing (NLP). For instance, AGI can help to choose more appropriate language models in part-of-speech (POS) tagging, word-sense disambiguation, detecting discourse relations and automatic summarization. Giesbrecht and Evert [52] showed that a change in the genre of a dataset can have a direct impact on accuracy of POS tagging. In their experiment, POS tagging achieves 96.9% accuracy on newspaper texts whereas it reaches only 85.7% accuracy on forums. Therefore, automatic identification of the genre of a text as a pre-processing step to POS tagging could help to choose more appropriate language model for POS tagging. As an additional example, Petrenz and Webber [112] found that the probability of the word "states" being a verb is about 20% in letters, whereas this probability drops to only about 2% in editorials.

Webber [162] showed that genre classes such as letters to the editor and newspaper articles are different in terms of the distribution of certain discourse relations. Since some discourse relations are more frequently found in texts belonging to a certain genre, prior knowledge of the genre of the texts could help to detect these relations more accurately.

Automatic genre identification could also be used to improve techniques for automatic summarisation. Stewart [150] shows that genre-oriented summarisation algorithms perform better than summarisation algorithms that do not take into account the genre of the documents. These techniques are based on the fact that the structure and the position of important information in a text correlates with its genre [55, 165]. For example, the first few sentences of a news article might be sufficient to summarise its content, whereas, this approach is not appropriate for summarising a story. Another application of AGI is the field of word-sense disambiguation. Martinez and Agirre [94] emphasise that work on word sense disambiguation must take into account genre as an important parameter, because genre can affect the results of word-sense disambiguation models, as certain senses may occur only in certain genre classes.

1.2 Objective of the Thesis

In this thesis, the main AGI questions to be addressed are:

1. **Is it possible to use crowd-sourcing to reliably annotate a web genre corpus?** Firstly, we wish to investigate whether humans can agree on our predefined genre classes of web pages.

2. **What set of features produces the best result in genre classification?** The second goal of this research is to find the best set of features that can be used in an automatic supervised classification method in order to distinguish genre classes with high accuracy.

3. **Can we exploit the graph structure of the web to increase genre classification accuracy?** We also investigate whether the connectivity of web pages could help us to implement a better genre classifier on the web.

4. **How does a genre classification model built on a designed web genre corpus perform on a random web genre corpus?** In order to answer this question, we, for the first time in genre classification, test the transferability of the best performing genre model from a designed corpus to a random corpus.

5. **How does an open-set genre classification algorithm perform on the real web?** Finally, we explore genre classification in a more realistic setting by simulating the web environment where not all web pages belong to a predefined genre class. In this setting, the classifier has to have the power to reject to classify a web page as one of the preselected genre categories.

1.3 Main Contributions of the Thesis

The significant contributions of our research are the construction of the first reliably annotated web genre corpus as well as the investigation and development of new algorithms for automatic genre identification. Below is a brief summary of our key research contributions.

- We present the Leeds Web Genre Corpus (LWGC) which is the first reliably annotated web genre corpus. This collection includes 15 genre classes which are reliably annotated via crowd-sourcing. The LWGC consists of two subcorpora: The first one (LWGC-B(alanced)) is a designed corpus, where web pages were collected using focused search for specific genres by following links in available web directories before them being submitted to the crowd-sourcing annotation. This method leads to a balanced distribution of genres in the corpus, ideal for automatic genre identification via machine learning methods that need sufficient training material for each genre (a property that many existing collections lack). In addition, we collect the corpus from a wide variety of sources, circumventing spurious topic-genre correlations existing in some prior corpora. Our second subcorpus (LWGC-R(andom)) then expands our method successfully to a corpus where the pages to be annotated are collected in a more arbitrary way among web pages returned by search engines. This sub-corpus also allowed us to investigate coverage of the underlying genre inventory.

- We compare the performance of different features in a supervised genre classification setup using a reliable and topic-diverse web genre-annotated corpus. The drawbacks of existing genre-annotated web corpora (e.g., low inter-coder agreement; false correlations between topic and genre classes) resulted in researchers' doubt on the outcomes of classification models based on these corpora [139]. Therefore, at the time of our research, the question which set of features produces the best result in automatic genre classification on the web was still an open question.

- We employ a semi-supervised graph-based algorithm as a novel technique in genre classification in order to learn from neighbouring web pages. The experimental results show that our semi-supervised algorithm significantly improves the overall genre classification accuracy produced by conventional supervised classification using features appeared within the web pages. However, it seems that some genre classes benefit more from this graph-based model than others.

- We explore an open-set genre classification technique for the first time on a random slice of the real web. A small number of studies proposed open-set classification techniques for genre classification. However, the ability of these techniques to detect web pages which do not belong to any of the predefined genre classes have not been tested due to the lack of such dataset.

1.4 Thesis Outline

This rest of this thesis is structured as follows:

In Chapter 2, we present the related work on automatic genre classification. This chapter provides a detailed picture of previous attempts at genre classification of documents by other researchers.

Chapter 3 describes the creation of the LWGC-B corpus which is compiled by focused search. We detail our motivation for creating the corpus and also describe our annotation guidelines and methodology. Then we present the results of our human agreement study as well as a statistical analysis of our final, gold standard corpus.

Chapter 4 presents supervised learning algorithms for automatically predicting the genre of web pages on the LWGC-B corpus which was created in Chapter 3. We compare the performance of various features proposed by other researchers for genre classification as well as investigate the usefulness of novel features in AGI.

Chapter 5 proposes a semi-supervised graph-based algorithm as a novel technique for genre classification. The motivation behind applying graph-based learning is to learn from the neighbouring web pages. However, since we do not have the genre labels of the neighbouring pages in our dataset, we employed them as unlabelled data using a semi-supervised setting. In this chapter, we use a semi-supervised min-cut algorithm as a novel technique in genre classification and show that this algorithm can improve genre classification results.

Chapter 6 investigates whether we can achieve high inter-annotator agreement on random web pages too. In order to answer this question, we compiled random web pages by querying search engines and applied exactly the same annotation scheme used in Chapter 2 for annotating the LWGC-B corpus, which resulted in creation of the LWGC-R corpus. This chapter also seeks to answer the question whether a model trained on our designed corpus (LWGC-B) can successfully be used for automatic genre classification on random web pages (LWGC-R corpus). In addition, this chapter investigates genre classification as an open-set classification task.

In Chapter 7, we summarize the work presented in the thesis. We also point out some potential directions for further research.

Chapter 2

Related Work

2.1 Introduction

The advent of computers has paved the way for a great increase in text analysis research. The new technology made it possible to electronically store and process large amounts of data. The Brown Corpus was the very first large computerised corpus created in the 1960s. It consists of 500 English texts which are classified into 15 genre classes such as reportage, editorial and fiction [97]. Later, the British National Corpus (BNC) was designed and created [84]. This collection includes 70 genre classes which are defined according to a range of parameters such as the type of medium (e.g., book or newspaper), audience (specialists or lay persons) and domain (e.g. leisure or applied science). With the arrival of the web, it became much easier to collect large corpora. The web also resulted in new genres not available before, such as personal home pages.

In the past two decades, researchers have tried to automatically classify documents based on their genres using both web-based and non-web-based corpora. Although in this thesis we focus on automatic genre identification on web corpora, we discuss related work in automatic genre classification in general. The outline of this chapter is as follows. We start by examining the concept of genre in general and on the web in Sections 2.2 and 2.3 respectively. In Section 2.4, we investigate the issues surrounding genre classification schemes. Section 2.5 introduces existing publicly available web genre corpora and their limitations. While Section 2.6 investigates the performance of various features in AGI, Section 2.7 focuses on different algorithms proposed for AGI. Section 2.8 examines the impact of factors such as topic and authorship on genre classification. Finally, Section 2.9 concludes this chapter.

2.2 The Concept of Genre

Many researchers have studied the notion of genre and various definitions of genre have been long debated in this field. Campbell and Jamieson [24] defined genre as:

> "a group of acts unified by a constellation of forms that recurs in each of its members. These forms, in isolation, appear in other discourses. What is distinctive about the acts in genre is a recurrence of the forms together in constellation." ([24, p. 20])

In this definition, the emphasis is on the "form" as it is identified as the main attribute that texts in a genre class share. A similar but extended definition is given by Miller [105, p.159] who argues that the definition of genre must not be limited to the form of the discourse only, but it should also include "the action it is used to accomplish". In other words, texts in a genre class have the same purpose or goal as well as similar patterns of form. Moreover, Biber [17] emphasises the importance of "purpose" in recognizing a genre class by stating:

> "I use the term genre to refer to text categorizations made on the basis of external criteria relating to author/speaker purpose."
> ([17, p. 68])

Swales [154]'s definition of genre is in line with Biber's as he also recognizes "purpose" as a principle attribute that instances in a genre class share. Later, Orlikowski and Yates [109] arrived with a more comprehensive definition of what genre actually is. They define genre as:

> "a distinctive type of communicative action, characterized by a socially recognized communicative purpose and common aspects of form." ([109, p. 543])

Orlikowski and Yates's [109] definition adds a new dimension to a genre class by clearly stressing that genres must be socially recognizable. In other words, genre classes exist only if they are identifiable by people in the society [1]. This definition also indicates purpose and form as the two main attributes in defining a genre category. In this thesis, we adopt this definition of genre which seems to be the most comprehensive definition.

It must be noted that in the literature, there is much confusion between the notion of genre and related concepts such as text type, style and register as these terms are often used interchangeably. Moessner [106] used the metaphor "terminological maze" to describe this confusion. Many researchers tried to provide clarification of the use of these terms. For instance, Biber [17] attempts to differentiate genre and text type by stating:

> "I use the term genre to refer to text categorizations made on the basis of external criteria relating to author/speaker purpose."
>
> ([17, p. 68])

> "I use the term text type on the other hand, to refer to groupings of texts that are similar with respect to their linguistic form, irrespective of genre categories." ([17, p.70])

Lee [84] agrees with Biber's [17] differentiation of these two terms. He, after an extensive survey of usage and definitions of these terms by the researchers, concludes that text type refers to rhetorical categories such as narrative, description, exposition and argumentation, because documents categorised as belonging to each of these classes are similar in terms of linguistic form. Therefore, texts can have the same text types but belong to different genres (e.g. both genre classes editorial and review are argumentative).

Moreover, Lee [84] defines the term style as a way of characterising the language use by individual authors, whereas he refers to register as linguistic patterns associated with different situations. Biber [18] sheds more light on differences between register and style by saying:

> "The style perspective is similar to the register perspective in its linguistic focus, analyzing the use of core linguistic features that are distributed throughout text samples from a variety. The key difference from the register perspective is that the use of these features is

> not functionally motivated by the situational context; rather, style features reflect aesthetic preferences, associated with particular authors or historical periods." ([18, p.2])

Thus, although both register and style concentrate on linguistic patterns, the former is interested in linguistic patterns associated with particular authors or particular historical periods while the latter is more concerned with the linguistic patterns associated with particular situation and communicative purpose. Additionally, Biber [18] clarifies the definitions of the two terms genre and register by pointing out their differences:

> "The genre perspective is similar to the register perspective in that it includes description of the purposes and situational context of a text variety, but its linguistic analysis contrasts with the register perspective by focusing on the conventional structures used to construct a complete text within the variety, for example, the conventional way in which a letter begins and ends." ([18, p.2])

Therefore, according to Biber, we often need to have a complete text in order to identify the linguistic characteristics associated with the genre (e.g. beginning and ending of a letter are very important for its recognition), whereas, text excerpts are enough to recognize the register of a text because linguistic features which reveal the register occur through out the text. Biber [18] also argues that genre features are often conventional rather than functional. He explains this further by providing an example: a news story from a genre perspective begins with a concise title; the first few sentences is a summary of the main event and the text usually includes multiple paragraphs which are short and self-contained. In contrast, a news story from a register perspective shows the linguistic patterns such as past tense verbs, passive voice verbs and reported speech, that are distributed throughout the text and are typical of the register of newspaper reporting.

We see in this example by Biber [18] that when he observes documents from the register point of view, he emphasises on linguistic forms or patterns common in these texts. However, he also stresses on the linguistic forms when he provides a definition for the text type. Moreover, in the definition of genre by Orlikowski and Yates [109] and Miller [105], form is one of the key attributes in recognising a genre class. Therefore, as Moessner [106] put it we face a "terminological maze" here and researchers use these terms interchangeably. In this thesis, we observe the texts through the genre lens and based on the genre definition we adopted in this thesis, both form and purpose of the documents are important in the recognition of the genre classes.

2.3 Genre on the Web

Since this thesis focuses on genres on the web, it is important to compare these genres with genres in traditional media. The World Wide Web which was created in 1989 is a communication medium for retrieving and displaying multimedia hypertext documents [14]. Yates et al. [164] recognized the advent of a new communications medium as one of the reasons for the emergence of variants of existing genres or of new genres.

The development of genres on the web has been studied and analysed by researchers in this field. Shepherd and Watters [141] introduced the notion of "cybergenre" for the first time to describe the genres in the new medium. They proposed a hierarchical taxonomy that classifies the genres on the web. According to this classification, cybergenres can be extant (i.e.“ based on existing genres") or novel (i.e. "not like any existing genre in any other medium"). On-line newspaper is an example of extant genres and personal homepage is an example of novel genres. Dillon and Gushrowski [39] also argued that the personal homepage is a novel genre on the web which has no equivalent in the world of print. Moreover, extant genres are further divided into two sub-classes: replicated (i.e.“based on genres existing in other media") and variant (i.e.“a modification of existing genres"). Novel genres are also separated into two groups: emergent (i.e. "derived but significantly different from existing genres"), and spontaneous (i.e. "never employed in other media"). They refer to personalized newspaper and frequently asked questions as examples of emergent and spontaneous genres respectively.

Crowston and Williams [33] proposed a similar taxonomy for the genres on the web. They conducted a survey on 1000 random web pages and distinguish four different types of genres: reproduced genres, adapted genres, novel genres and unclassified web pages (Table 2.1). The first type which comprised 60% of the data set consists of pages which to a great extent replicate the genres in the traditional media. The second type which is referred to as adapted genres are

Type of genre	Percentage
reproduced genres	60.6%
adapted genres	28.6%
novel genres	5.3%
unclassified web pages	5.6%

Table 2.1: Percentage of types of genres on the web reported by Crowston and Williams [33]

genres which evolved from existing genres on the paper world to new genres on the web by using the capability of the new medium. For example, a list of items which makes use of the hyper-link capability of the web to link to other pages with more information on those items is creating both a list and an index. Therefore, these pages have a new functionality and purpose which is a combination of a list and an index. The third type of genres on the web is the novel genres which are exclusive to the web.Home pages are the examples of this group.

Although the proportion of the novel genres in this study is very low, it is possible that nowadays this group of genres comprise a bigger percentage of the web due to the rapid growth and easy accessibility of the web.

In Crowston and Williams's survey there are pages which remained unclassified due to two main reasons: not knowing the name of the genre and the difficulty of determining the purpose of the web page. Some of these unclassified web pages could be the example of genres still in formation. Therefore, in the process of building a genre-annotated web corpus, we would expect to find some web pages without any genre label.

The outcomes of these studies provide an insight into the composition of genres on the web. These two classifications show that in general the web is hosting genres which have counterparts in other media as well as new genres which are recognized and acknowledged by many web users (e.g. social forum sites and shopping websites [68]) and genres which are still developing.

2.4 Genre Classification Schemes

Researchers have always been interested in classification of various entities with the aim of bringing order to the chaotic world, and the world of documents and in particular the world of web documents are no exception. The objective of a genre classification scheme is to help people make sense of different types of communicative actions. A variety of genre classification schemes have been developed by different researchers using mainly two different techniques: top-down and bottom-up [34].

In the first technique, researchers use existing sources or classifications, or rely on their own judgement, knowledge and understanding of genres in order to define genre classes. The example of top-down genre classification schemes are: the very fine-grained genre categorization system created by Gorlach [56] which consists of 2000 genres; seventy genre classes in the British National Corpus (BNC) [84]; Dewe et al.'s [37] web genre classification scheme which includes about 24 categories; seventy genre classes presented in [16]; 500 genre labels listed in [40]; the web genre classification scheme created by Crowston and Williams [33] which includes about seventy genre categories; a list of twenty genre classes by Vidulin et al. [159] and the genre categorization scheme by Sharoff [138] which consists of only seven very broad genre categories. The creators of some of these classification schemes claim their completeness. For example, the last two examples (i.e. [159] and [138]) claim that any web page can be categorized into one of the genre classes in their proposed genre classification schemes.

In the second technique (i.e. the bottom-up method), a group of people are asked to produce genre labels for the classification scheme. The genre classification scheme with 292 genre classes presented in [32] was produced based on this approach. Crowston et al. [32] argue that the bottom-up approach is a better way of creating genre classification systems, because the top-down approach is not capable of discovering new genre classes such as very domain-specific genre categories. Also they note that "it is important to capture the users' own language and understanding of these genres" [32, p.7]. However, they express their disappointment with the results of their study with respect to its usefulness in building a classification scheme of genre terms for further application. One of the challenges that they encountered during their study was that there appears to be no common understanding on what a genre is. Another challenge was that it was not always easy for the participants to identify the purpose of the web pages. They also pointed out that sometimes respondents found it hard to name a genre for particular web pages, because they could not think of the term that describes the purpose of the pages. Therefore, they concluded that it is very hard to study genres naturalistically.

Some researchers combined both top-down and bottom-up approaches in order to benefit from both techniques. For example, Roussinov et al. [128] conducted a study based on the combination of these two methods in order to identify genres on the web. In this study, a number of interviewers asked 184 individual web users (respondents) to describe the reasons for searching the web and to assign a genre class to each of the web pages that matches their information needs. In this study, the initial genre classes were taken from [33]. However, the web users could add new genres to it to produce the final genre classification scheme. After examining 1234 web pages, 116 different genres were identified of which 44 were new ones. In the next step of the study, the interviewers annotated the web pages with the list of genre categories produced in this study. However, the percentage agreement between the interviewers and the respondents was only 49.63%. This study shows that genre classification is not an objective procedure and it is difficult to get a high agreement on the genre of web pages.

A different approach called faceted classification was proposed by Crowston and Kwasnik [34] who recognized the complexity of the concept of genre. Facets are attributes or properties associated with a genre which people use as clues to identify the genre of a document. In this approach, instead of directly assigning a genre to a web page, it is viewed from various angles and through different facets. The starting point of this classification system is identifying the basic facets which are used to differentiate genres. Once these attributes have been identified, one can go to the next phase, which is analysing documents using the facets.

The first advantage of this approach is its flexibility. Since each document is viewed through a number of independent attributes, these attributes can be put together in various combinations at the time of retrieval to produce new categories. For example, combining the attributes “communicative”, “personal” , “informal” and “emotive” may result in retrieving personal letters or emails expressing emotion of the writer whereas combining “communicative”, “formal”, “personal” and “testimonial” may result in recommendation letters or emails. The second advantage of this classification system is that it is hospitable to new genres [34]. We can define emerging genres which we did not previously encounter based on established and known facets [34]. However, the disadvantage lies in the difficulty to establish appropriate facets. It is hard to determine all the important attributes of a given genre and therefore developing a complete set of facets will be a real research challenge. Therefore, due to these difficulties, this approach has been never put to practice and no annotation experiment has been conducted based on this proposal.

Reviewing all these genre classification schemes shows that there is no universal and standard genre classification scheme. The literature review illustrates a lack of consensus among the researchers about genre classification. Although genres are recognized and used, there is disagreement in definitions, boundaries, granularities and level of specificity of genre labels [82]. Genre classification is highly subjective and people define their own genres differently. However, based on the definition of genre that we adopted in this thesis (Section 2.2), genre classes must be socially recognizable. In other words, existence of a genre class depends on whether people can distinguish this category from other genre classes reliably. Therefore, as long as a genre class fulfils this criterion (i.e. being socially recognizable) and also documents belong to this class share same communicative purpose and form, it can be added to the genre classification scheme.

2.5 Existing Publicly Available Web Genre Corpora

Several efforts have been made to build genre annotated web corpora and to employ them for research in the field of automatic genre identification. These collections can be divided into two main groups: page-level and site-level. In the page-level corpora, genre labels are assigned to individual web pages, whereas, in the site-level collections, the website is the unit of the process. Although the majority of the web genre collections are page-level, there exist a few site-level genre annotated corpora such as the one employed in the study by Lindemann and Littig [88] and the corpus used in [102].

However, only a few of web genre collections are publicly available. Also, it must be noted that each of these collections is different in terms of the size of the corpus; how the web pages were collected; how the web pages were preserved and the set of genre labels used (see Appendix A for the complete lists of genre labels for each of these collections). Table 2.2 compares some properties of these corpora. The following is a short description of publicly available web genre collections, after which we will summarize some of their characteristics.

Hierarchical Genre Collection (HGC) [151] was annotated based on a set of hierarchical genre labels with seven main categories and thirty two sub categories, e.g., literature as a main category with the subcategories poem, prose and drama. This collection consists of 1280 web pages preserved in HTML format. For each genre category, forty example pages were manually collected.

I-EN-Sample [140] consists of 250 web pages randomly selected from the I-EN corpus of web pages representing a snapshot of the English web texts from 2005 [137]. This collection was annotated using the Functional Genre Classification (FGC) scheme which consists of seven macro-genres aimed at describing the genre of any text. The genre palette in FGC is based on the function or purpose of the document e.g., discussion which includes academic papers, forums, emails or political debates, or instruction which covers FAQs, manuals and tutorials. Therefore, this annotation scheme differs from all others by sacrificing depth and specificity of the annotation scheme for coverage.

KI-04 [104] is another genre-annotated web corpus consisting of 1205 HTML documents. This collection has been annotated using eight genres, e.g., link collection, shop and articles. The genre list in this collection focuses on including the genre classes that are most useful for web search —it was developed by asking a group of students to fill in a questionnaire about typical topics for queries and favourite genre classes.

The KRYS I [16] collection consists of 6200 PDF documents. This corpus has been annotated using seventy genres which are grouped into ten sets, e.g. commentary and review in the journalism group. Although this selection is meant to be a web genre-annotated corpus, it includes only web pages in PDF format. Therefore, genres that do not normally use this format, such as homepage and shop, are not included in this corpus.

MGC (Multi-labelled Genre Collection) [159] is the only genre-annotated corpus which allowed multi-labelling. This means that each web page can be categorized as belonging to several genre classes. It consists of 1536 web pages classified into twenty genres. They were collected by targeting web pages in these genres, as well as using random web pages and popular web pages coming from Google Zeitgeist.

SANTINIS [131] corpus which consists of 1400 web pages was annotated based on seven genres. This collection focused on genres which are exclusive to the web, e.g. blog and frequently asked questions (FAQs). In the compilation of this corpus only web pages which clearly belong to these genres were manually collected.

The Syracuse [32] collection consists of 3027 web pages annotated based on 292 genres. The genre palette in this collection was developed by asking three groups of people (teachers, journalists, engineers) to produce web genre terms themselves.

The corpus constructed in [43] which has 1000 random web pages categorised into eight very broad main genres (e.g. description, discussion and opinion) and 56 sub-genres. This corpus was annotated via Amazon Mechanical Turk which is a crowd-sourcing website. The corpus presented in this paper [43] is the work most similar to ours in that they use crowd-sourcing as annotation methodology. However, they fail to achieve reliability (see Table 2.2 and further discussion below on reliability).

As can be seen, these corpora differ in genre definitions and categories. However, they share the following problems or shortcomings.

Reliability. One problem with all the existing genre-labelled collections is the issue of reliability of the annotations. Corpora such as SANTINIS, KI-O4 and Syracuse have been annotated by a single person and as a result, their inter-annotator agreement measures cannot be computed. The MGC, I-EN-Sample and KRYS I corpora have been double-annotated. However, agreement measures were low (α=0.56 for MGC and α=0.55 for I-EN-Sample) as discussed in detail in [139].[1] As an example, Table 2.3 shows the low percentage agreement for the KRYS I corpus in percentage agreement (chance-corrected agreement tends to be even lower).

The corpus constructed in [43] is annotated via crowd-sourcing. In this collection, four annotations were assigned to each web page by the annotators in Amazon Mechanical Turk which is a crowd-sourcing website. However, the result of this annotation study shows that they fail to achieve good reliability in annotation, even for the broader genres: on the main genres, on only 63% of the web pages at least 3 out of four annotators are in agreement; for the fine-grained genres, on only 43% of the web pages at least 3 out of 4 annotators are in agree-

[1] We refer the reader to [4] for a comprehensive survey of inter-coder agreement measures such as percentage agreement as well as chance-corrected agreements α and κ.

Corpus	# of		# of pages per genre			Format	Collection method	Reliability
	pages	genres	min	max	med			
KRYS I [16]	6200	70	6	117	97	PDF	focused search	a.p.a.= 50.38% (Table 2.3)
MGC [159]	1536	20	55	227	77	HTML with images	both random selection and focused search	Low α=0.56 [139]
HGC [151]	1280	32	40	40	40	HTML only	focused search	not measured
KI-04 [104]	1205	8	126	205	145	HTML only	focused search	not measured
SANTINIS [131]	1400	7	200	200	200	HTML only	focused search	not measured
I-EN-Sample [140]	250	7	10	99	30	TXT from HTML	random selection	Low α=0.55 [139]
Syracuse [32]	3027	292	1	174	3	HTML only	focused search	not measured
The corpus in [43]	1000	56	0	99	1	HTML	random selection	43% of the time three out of four annotations agreed

Table 2.2: This Table summarizes some characteristics of genre-annotated corpora. a.p.a. stands for average percentage agreement. Max, min and med are abbreviations of minimum, maximum and median respectively. See [4] for a comprehensive survey on inter-coder agreement measures such as percentage agreement, α and κ.

ment. Only percentage agreement is measured — chance-corrected agreement is likely to be even lower as their genre distribution is skewed (see also Section 3.2.4 for a discussion of agreement measures).

In addition, another issue regarding annotation in these corpora is that expert annotation can mislead with regards to the general applicability of the annotation scheme, especially if these are the same experts as the ones who developed the theoretical terms and concepts underlying the annotation scheme [124]. This was the case for example in SANTINI and KI-04.

Size. Some of the prior collections are not large enough to ensure representativeness of genre classes. Table 2.2 compares these collections in terms of maximum, minimum and median number of web pages per genre category. As can be seen, they often have few annotated web pages *per category*, especially for the KRYS-I, Syracuse and the corpus created by [43] via crowd-sourcing, while machine learning algorithms often require a reasonable number of training examples in order to

Annotators	Agreement
Student and Secretary I	51.74%
Student and Secretary II	53.76%
Secretary I and II	45.65%
All three	37.85%

Table 2.3: Human agreement for the KRYS I corpus [16] which has *seventy* genre classes. Results illustrate a low percentage agreement.

produce satisfactory results.

Format. Another major drawback of the existing corpora is that they have been preserved in different formats such as PDF or plain text which results in losing HTML tags. For instance, each web page in KRYS I corpus is saved in PDF format and as a result automated tools are needed to convert PDF to plain text or HTML format. However, these tools are error prone and as a result some information may be lost or wrongly converted. Also previous studies in AGI show that HTML tags can improve the accuracy of genre classification [63] and should therefore be kept when collecting web genre corpora.

Topic Diversity. There are genres which have a natural, strong correlation with certain topics, for example the genre label recipe has a profound connection to the topic label food. These types of correlations between genres and topics are true and explicit connections and will always exist. However, in some existing (designed) genre-annotated corpora, there are a number of correlations between genres and topics which are spurious in that they are due to the way the focused search for genre texts was conducted. For example, a large sample of the frequently asked questions texts in Santinis corpus [131] come from web sites about hurricanes. Such spurious correlations will mislead any investigation into typical genre properties (for example [112] show that the often best-performing word unigrams features in AGI perform considerably worse when topic is varied so that potentially AGI based on these features learns topics rather than genres). Therefore, a corpus without any unwanted correlation between genres and topics is needed (see Section 2.8 for a discussion of the influence of topic on genre classification).

In this section, we introduced existing publicly available web genre corpora and pointed out their limitations. Next section describes various features which are extracted from these corpora and employed in automatic genre classification using standard supervised machine learning algorithms such as SVM and Decision tree. In contrast, Section 2.7 focuses on algorithms used in AGI which go beyond the standard supervised algorithms. In order to be able to easily compare the performance of different approaches, we summarise the results of different AGI systems on the most frequently used genre corpora in Table 2.4 at the end of Section 2.7.

2.6 Genre Classification: Features

There has been a considerable body of research in AGI. In previous studies on automatic genre classification, various combinations of features have been employed for genre classification. However, because these experiments have been conducted on different corpora with different sets of genre labels, it is quite difficult to compare these results. Moreover, some studies focus only on one specific genre. For instance, the experiment presented in [142] explores only home pages and its sub-genres personal, corporate and organization.

In this thesis, we group the features used in genre classification into four main categories: structural, lexical, text statistic and other features. The lexical group comprises features such as word *n*-grams, character *n*-grams and genre-specific words that can be directly derived from the text. In contrast, extracting structural features which refer to features such as part-of-speech tags and noun or verb phrases, need more sophisticated natural language processing techniques. The extraction of these features is dependent on automated tools such as part-of-speech taggers and parsers.

The third group comprises a wide range of text statistics such as type/token ratio, number of sentences, average length of sentences, average length of words, frequency of punctuation marks and HTML marks. The last group (i.e. other features) includes features such as the ones which are extracted from the URLs of web pages, or the ones which are extracted from the images of the documents. While many features in these four groups are common between the documents in the print world and the documents on the web, there are features such as HTML tags and URLs which are specific to web documents. The following subsections examine the related work on automatic genre classification with respect to the use of these groups of features in more detail.

2.6.1 Structural Features

Structural features refer to features which capture syntactic characteristics of the text. One attractive property of structural features is that they are topic independent. Therefore, they potentially capture the differences between genre classes without being influenced by the topic of the documents. As a result, they are known as stable features in genre classification [112].

However, one disadvantage of these features is that their extraction can be computationally expensive and may not be practical for being employed by search engines. Another drawback of the structural features is that their extraction depends on automated tools such as part-of-speech taggers and parsers whose performance varies for different genres [52]. Even the best part-of-speech taggers and parsers are error prone and cannot be trusted on new unseen genres. Therefore, as pointed out by other researchers such as McEnery [100], extracting structural features is a time-consuming and unreliable procedure. The following subsections examine the related work which used structural features in genre classification.

2.6.1.1 Part-of-speech Tags

Genre classification can be traced back to studies by Karlgren and Cutting in 1994 [67] on the Brown corpus. They extracted twenty features from the corpus which included frequency of some part-of-speech tags such as adverbs, first person pronouns, prepositions, nouns and present-tense verbs. Then they employed statistical data analysis in order to classify the genre categories. With this set of features, they achieved about 65% accuracy (compared to the baseline accuracy of 19.4%) on the 10-genre Brown corpus (they grouped the sub-genres under*fiction* together, leading to 10 genres to classify). Since then, many other researchers such as Eissen and Stein [104], Vidulin et al. [159] and Feldman et al. [46] used frequency of the part of speech tags in genre classification as structural cues in combination with other features.

The motivation behind the use of POS tags in genre classification could be rooted in Biber's [17] study of genre classes. The objective of his research was to explore similarities and differences among genres in spoken and written texts with respect to text type. In order to reach this aim, two text corpora (namely the LOB corpus [59] which comprises written texts in 15 genres and London-Lund corpus [153] of spoken English in six genres) were selected for the experiment. In the next step, the frequencies of 67 linguistic features which include part-of-speech tags as well as features such as type/token ratio and word length were obtained.

Next, factor analysis was employed to identify the most significant underlying orthogonal dimensions based on the group of features that co-occur in texts with high frequencies. Then each group of features also known as factor is interpreted as a linguistic dimension based on the communicative functions that are shared by the features in each group. Based on the recognized co-occurring features, Biber identified six text type dimensions such as "Involved versus Informational production" and "Narrative versus Non-Narrative". For each factor, the average factor scores for each genre class was computed and the results were compared. For example, in the dimension "Involved versus Informational Production", the POS features with high positive weights are present tense verbs and second person pronouns. On the other hand, the POS features with negative weights are nouns and prepositions. In this dimension, example of genres with high positive and negative scores are conversations and academic prose respectively. However, in the second dimension "Narrative versus non-narrative", the most co-occurring feature is past tense verbs. Genres which have high scores in this dimension are romantic fiction, mystery fiction and science fiction (narrative) whereas genres such as official documents and academic articles obtain a low value in this dimension (non-narrative). Although Biber did not perform automatic genre classification, he showed that some particular POS tags are more frequent in some genre classes.

2.6.1.2 Part-of-speech *n*-grams

Argamon et al. [2] were the first to use part of speech trigrams (sequences of three consecutive part of speech (POS) tags) to capture the syntax of the text. Argamon et al. [2] provide two reasons to justify the use of POS trigrams: first, they are large enough to grasp useful syntactic information, and second they are small enough to be computationally manageable. Argamon et al. [2] employed POS trigrams in automatic genre classification and reported an accuracy of 78% for differentiating news articles from editorials (compared to the baseline accuracy of 50%) . This result shows that these two genre classes are different in terms of syntactic structure and that POS trigrams can capture these differences to a reasonable degree of accuracy.

Santini [131] adopted this approach and examined the discriminative power of POS trigrams on the BNC. She selected a subset of documents belonging to 10 different genre classes of the BNC. POS trigrams achieved 82.6% accuracy on this subset (the baseline accuracy for this subset of BNC is 10%). Following this good result, she was encouraged to investigate whether POS trigrams could also be used as genre-revealing features on the web. In order to answer this question, she tested this approach on Santinis [131] web genre corpus which consists of seven genre classes. She reported an accuracy of 86.50% (compared to the baseline accuracy of 14.28%) for this collection. This result illustrates the usefulness of POS trigrams as genre-revealing features on the web.

2.6.1.3 Features based on Grammatical Analysis

POS trigrams as introduced in the previous section are capable of capturing the syntactic structure of the texts to some extent without using a parser. However, this technique is a shallow approach to extract the syntactic properties of a text. Therefore, Santini [131] examined a set of features which are based on more sophisticated analyses of linguistic structure.

The assumption behind this work is that certain syntactic constructions are more likely to appear in particular genre classes. This assumption was inspired by studies such as Biber [17], Quirk et al. [120] and Biber et al. [19]. The grammatical features explored by Santini [131] include syntactic patterns (e.g. adverbial clauses, complement clauses and main clauses) and functional cues (e.g. first, second and third person; imperatives; active and passive voice). These features which are referred to as facets are macro-features containing several micro-features. For example, “third person facet” comprises third person singular and plural pronouns, including possessives and reflexives.

Santini [131] compared the performance of these features with POS trigrams features on Santinis corpus and reported that POS trigrams returned an average accuracy of 86.50%, while grammatical features returned an average accuracy of 84.28%. These results show that POS trigrams can more accurately distinguish the genre classes. One reason for the lack of success of the grammatical features could be that their extraction relies on the output of automatic parsers whose performance may not be very accurate. On the other hand, the extraction of POS trigrams is much easier and do not require a parser.

2.6.2 Lexical Features

2.6.2.1 Common Words

Stamatatos et al. [146] were the first to use common words (words such as *the, of, and, a*) as features to distinguish four genre categories namely: editorials, letters to the editor, reportage, and spot news from the Wall Street Journal corpus. The result of their experiment showed that common words can be very helpful in determining these genres. They reported an impressive accuracy of more than 90% when they used the fifty most common words as features (baseline accuracy for this corpus is 25%).

The most important characteristic of this set of features is that it is topic independent. Therefore, it can capture the differences between the genre classes without being influenced by the documents' topics. This property inspired other researchers such as Petrenz and Webber [113] and Santini [131] to include common words as features in their automatic genre classification experiments.

2.6.2.2 Function Words

Another set of features used in genre classification is function words. In linguistics, words are classified into two distinct sets : function words and content words. As Tausczik and Pennebaker [156, p. 29] state:

> "Intertwined through these content words are style words, often referred to as function words. Style or function words are made up of pronouns, prepositions, articles, conjunctions, auxiliary verbs, and a few other esoteric categories." ([156, p. 29])

One property of function words is that they are more frequent than content words. Although there are almost 100,000 word types in the English vocabulary, only about 500 (or 0.05%) are function words [156]. However, 55% of all the word tokens we speak, hear, and read are function words [156]. This distribution of words in English follows Zipf's law [168] which states there are few words which are used frequently and many words which are used rarely.

Another property of function words is that they often lack lexical meaning [19]. In contrast to content words, many function words do not have any meaning themselves but rather they are defined in terms of their use, or function. We explain this property of function words with an example. In the sentence *"he has left the building."* the auxiliary verb *"has"* has little or no lexical meaning , but we can say that the function of the auxiliary verb in this sentence is to express present perfect tense. Therefore, function words mainly link the content words together by providing the grammatical relationships between them. Function words are also known as closed class words, whereas content words are referred to as open class words, because we can freely add new members to content words such as new nouns, but it is very uncommon that we come up with a new function word [36].

Argamon et al. [2] were the first to use a list of 500 function words in genre classification. Like common words, the rational behind the use of function words in genre classification is that they are topic independent and as a result they are expected to remain invariant for a given genre category across different topics. Argamon et al. [2] tested the discriminative power of function words by pairwise classification of four text collections: news stories from NY Times, editorials from NY Times, news stories from NY Daily News and magazine articles from Newsweek. They reported accuracy between 61% and 82% for different pairwise genre classifications. They also compared the performance of function words and POS trigrams in pairwise genre classification using these datasets. The output of these experiments illustrates that in the majority of cases, function words outperformed POS trigrams. Following this study, other researchers such as Vidulin et al. [159] included function words in the list of features they used for genre classification.

2.6.2.3 Word *n*-grams

Word *n*-grams, especially word unigrams, are popular feature sets in topic classification. However, it is also used in genre classification. Word unigram features are used in studies such as Freund et al [50] and Finn and Kushmerick [48]. Sharoff et al. [139] compared the performance of word unigrams, bigrams and trigrams on HGC, I-EN-Sample, KI-04, KRYS-I, MGC, SANTINIS, BNC and the Brown corpus and concluded that word unigrams perform better compared to other word *n*-grams. The lack of success of longer word *n*-grams could be due to feature vector sparsity, because most of the word *n*-grams are not encountered in a given text and as a result the feature vectors have less discriminative power.

Moreover, Sharoff et al. [139] who also compared a wide range of word, character and POS based features in AGI reported that word unigrams was one of the best performing feature sets. However, they concluded that these results cannot be trusted because of two main reasons. First, some of these results are not trustworthy because of the low inter-coder agreement of some of these collections. Inter-coder agreement is the measure of reliability of the annotation and any results based on unreliable data could be misleading. Second, the spurious correlation between topic and genre classes in some of these corpora was one of the reasons for some of the very impressive results (e.g. FAQs in SANTINIS corpus are mainly about tax and hurricane). Therefore, this feature set must be used cautiously in genre classification as it can be easily influenced by the topic of documents (see Section 2.8 which focuses on the impact of topic on genre classification).

In order to avoid using topic-dependent words in genre classification, a number of researchers used only function words in AGI, as noted in the previous section. However, the fact that some content words which are not necessarily topical are more likely to appear in some genre classes, motivated some researchers to identify and select these words and use them as genre-specific features. For example, Santini [131] produced a set of manually selected genre-specific-word facets as features which together with other features such as HTML tags and punctuation symbols led to a better result (90.6% vs. 86.50% using POS trigrams) on SANTINIS corpus using SVM classifier with 10-fold cross validation. Stubbe and Ringlstetter [152] also built a set of hand-crafted classifiers based on a set of manually collected features which have been derived based on linguistic knowledge and not by statistics. In order to determine whether these features have discriminative power, they calculated the mean occurrence of these features within each genre class of the training corpus and discarded the ones which showed no effective discrimination between the genre classes. This approach achieved 72.2% precision and 54.0% recall on the HGC corpus [151]. However, the main disadvantage of these two approaches is that these features are not appropriate for a fast changing environment such as the web, because they need to be manually reproduced or updated in order to be able to maintain their performance as genre revealing features.

The study by Kim and Ross [72] is also based on genre-specific words. They refer to these words as prolific words list. However, they construct this list automatically by compiling all the words within a genre class and then counting the number of files in which each word is found. In the next step, the prolific word list is built by taking the union of all the words found in at least 75% of the files in each genre. Each web page in the dataset is represented as a vector, where each entry is the frequency of a word in the prolific word list. These models were examined on the six genre classes (academic monograph, business report, book of fiction, minutes, periodicals, and thesis) of the KRYS I corpus. The reported results for individual categories show that this set of features achieves precision ranging from 0.74 to 0.91 and recall ranging from 0.71 to 0.94 on this dataset.

Another study which is also based on genre-specific vocabularies is presented by Eissen and Stein [148]. They proposed another technique for automatic extraction of genre-specific words which is more than simple count statistics. This approach assigns a score $f_c(w)$ to each word w in each genre class C based on $P(w)$ which represents the overall probability of w in the data set, and $P(w|C)$ which denotes the probability that w is drawn from documents which belong to C. The following formula defines this score:

$$f_c(w) = P(w|C).\frac{P(w|C)}{P(w)}.\frac{df_c(w)}{|C|} \tag{2.1}$$

where $df_c(w)$ is the number of documents from C that have w. The words with high scores are considered as genre-specific words (T_c). In addition to this score system, this study also takes into account the distribution of genre-specific vocabularies in a document. They justify this approach by showing that some genre-specific terms tend to appear concentrated in certain places on a web page and therefore, the concentration of these words in each web page plays an important role in genre classification. They used two different statistics in order to measure the concentration of the vocabulary in a document. While the first one measures the term concentration strength within a text window, the second one calculates how unequally genre-specific words are distributed over a document. This method achieved 80% average F-Measure on the KI-04 corpus. Additionally, they implemented a software called WEGA for web-based genre analysis based on this approach [149].

The experiments by Kim and Ross [74] also concentrate on distributive statistics of words as features in genre classification. This study shows that word distribution patterns are better features in AGI than the word frequency. In order to capture the word distribution patterns within a document, they measured three different statistics which measure, first, the time duration before the first occurrence of the word within the document, second, the average period ratio which calculates how evenly the word is distributed within the document and third, the time duration after the last occurrence of the word to the end of the document. They tested this approach on Santini corpus and also 31 genre classes of the KRIS I corpus. The outcome of these experiments indicates that the features based on word distribution patterns outperformed word frequency features on both datasets. This method yielded 80% accuracy on the subset of KRIS I corpus compared to 73% accuracy achieved by term frequency. Similar results are reported for the Santini corpus (96% vs 92%). However, it must be noted that this is not the best result reported for the Santini corpus as the character 4-grams binary representation [139] outperforms all other features on this corpus.

Moreover, we must emphasise that the words selected by the automatic genre-specific word selection methods such as the ones used by Eissen and Stein [148] and Kim and Ross [72] may not be only genre-revealing words, because topic-revealing words could be chosen by these methods as genre-revealing words, if there is a false correlation between the topic labels and the genre classes in the corpus. Therefore, when we employ these techniques, we still need to control factors such as topic which can affect genre classification. In the next chapter, we construct a web genre corpus from a diverse range of topics and sources in order to minimize the influence of topic on genre classification as much as possible (see Section 2.8 for a discussion of the influence of topic on genre classification).

2.6.2.4 Character *n*-grams

In all the techniques reviewed so far in this report, the smallest unit for linguistic feature extraction is the "word" whereas the methodology presented by Kanaris and Stamatatos [64] is based on character *n*-grams of variable length as features. In order to keep the dimensionality of the features to a practical level, only character 3-grams, 4-grams and 5-grams were considered in their experiment. The initial set of features consisted of the 1,000 most frequent fixed-length character *n*-grams. In the next step, each *n*-gram is compared with either immediately longer or shorter *n*-grams and only the dominant *n*-grams based on their frequencies are kept in the list.

This feature selection technique was tested on two corpora: SANTINIS and KI-04. The resulting feature sets of variable-length character *n*-grams were used for both binary and term frequency representation of web pages of each corpus. They used SVM as classification algorithm. The result shows that binary features improved the best reported results for both corpora at the time (accuracy of 96.2% for SANTINIS and 82.8% for KI-04).

In a much less complex approach, Mason et al. [95] used feature sets of fixed length character *n*-grams with normalized frequency representation. They examined the performance of different *n*-grams sizes (ranging from 2 to 7) and different number of the most frequent *n*-grams selected from each web page (ranging from 500 to 5000 in increments of 500). As classification algorithm, they used simple distance measure in two different ways. In the first approach, they measured the distance between a test web page and each training web page and assigned the genre of the web page in the training set to which it is closest. In the second approach, they construct a centroid feature set for each genre class using training data and measured the distance between each test set and these centroids and assigned the closest label to the test web pages. They reported that the second approach with using the 1000 most frequent 7-grams yielded the better result on SANTINIS corpus (accuracy of 94.6%). Although the result of this approach is not as high as the technique used by Kanaris and Stamatatos [64], it involves a much simpler procedure to produce the feature set.

In a more recent study, Sharoff et al. [139] investigated the performance of a wide range of textual features (word-based, POS-based and character-based features) on a wide range of genre collections (HGC, I-EN-Sample, KI-04, KRYS-I, MGC, SANTINIS, BNC, Brown corpus). In this experiment, each web page in each corpus was converted into plain text. Then from each document, the 1000 most frequent features were collected and combined together to create feature vectors. In the next phase, two feature representations normalized frequencies and binary have been used for word and character features. In the classification phase, a speed-up variant of linear SVM called Liblinear [44] was employed with ten-fold cross-validation. The outcomes of these experiments show that the binary version of character 4-grams and word unigrams as features yield the best results. They reported an accuracy of 97.14% for SANTINIS and 85.81% for KI-04 yielded by binary representation of character 4-grams which outperformed the results produced by the character *n*-grams of variable length reported by Kanaris and Stamatatos [64].

However, similar to word *n*-grams, character *n*-grams can also be easily influenced by the topic of web pages. Therefore, if we want to use this set of features in genre classification, the corpus must be constructed in a special way which is discussed in Section 2.8.

2.6.3 Text Statistic Features

Another widely used feature set in AGI are text statistics (e.g. type/token ratio, average word length, average sentence length and frequency of punctuation marks) which are usually used in addition to structural and lexical features. One property of these type of features is that they are topic-independent. As we observe in the literature, this characteristic encouraged researchers from early on to use these features in AGI without being worried about the impact of topic on genre classification. For example, in one of the earliest papers on automatic genre detection of web pages which dates back to 1998, Karlgren et al. [66] built a fully functional prototype genre-enabled search engine called DropJaw. This system was developed and tested based on a corpus of 1358 web pages with 11 genres. They extracted features such as part-of-speech tags as well as text statistic features such as number of digits, average word length and number of images from each web page. Simple if-then categorization rules using C4.5 [119] were employed for classification. This genre classifier yielded 66% accuracy compared to a 16.56% baseline.

An important set of experiments regarding the text statistic features is presented in Kessler et al. [70]. They conducted experiments on the six genre classes of the Brown corpus with the aim of comparing the performance of structural features with more easily extractable features (which they refer to as surface features). They built two classifiers: the first one based on 55 surface features which are mainly text statistic features (e.g. punctuation marks, average sentence length, type/token ratio and average word length) and the second one based on structural features (features used in Karlgren and Cutting [67] such as adverb count, first person pronoun count and preposition count). The result shows that structural features outperformed surface features. However, the difference between the results achieved by these two sets of features is only marginal. Therefore, they concluded that although structural features produce better results, this advantage does not justify the extra computational cost in most cases.

In addition to these text statistical features which are common in both the documents in non-web corpora and the documents on the web, frequency of HTML tags have been used as features in AGI on the web (e.g. [21, 104, 131]). HTML tags, which are specific to the web documents, provide a wide range of information about the web pages such as information about their layout, style, formatting and visual features (e.g. the number of images, tables, videos and buttons). HTML features are usually used in combination with other features in AGI. For instance, Kanaris and Stamatatos [64] showed that adding the frequency of HTML tags to character *n*-grams yielded better results (i.e. 84.1% versus 82.8% for KI-04 and 96.5% versus 96.2% for SANTINIS corpus).

2.6.4 Other Features

2.6.4.1 URL Features

Uniform Resource Locators (URL) is a string composed of characters which specifies the location of a document on the Internet [15]. Like HTML tags, URLs are also exclusive to the web documents. There are many different schemes for describing the URLs on the web. For instance, an HTTP URL has the syntax:

$$http : // < host >:< port > / < path >? < searchpart >$$

where $< port >$ can be omitted [15]. An example of a URL is:

$$http : //www.engineering.leeds.ac.uk/computing/$$

which is a unique address for the homepage of the School of Computing of the University of Leeds. Many researchers (e.g. [21, 26, 86, 160]) extracted features from URLs and used them for genre classification of web pages. The most common features extracted from the URLs are the existence of particular words such as genre names (e.g. *news, blog, faq*) in the URLs and also the type of the domain area of the URLs (e.g. *'ac.uk'* in the above example).

For instance, the URL features extracted by Lim et al. [86] include the domain area (e.g. com, edu, org), existence of particular words or characters in the URLs and the depth of the URL (i.e. number of directories included in the path). They compared the performance of the URL features with various other features such as 50 most frequent function words, HTML tags and text statistics on a corpus of 1,224 web pages categorized into 16 genre classes. They reported that HTML tags outperformed other features when compared with other individual feature sets (accuracy of 55.1% for HTML tags versus 43.5% for URL features). However, the best result was yielded by combining all the features (accuracy of 75.7% versus 55.1%).

Vidulin et al. [160] also extracted similar features from the URLs using the Multi-labelled Genre corpus. In addition to the URL features used by Lim et al. [86], they used new features such as appearance of year or existence of a query (i.e. $< searchpart >$) in the URL. However, they used these features in combination with other features and as a result the comparison between these two studies is not feasible.

2.6.4.2 Visual Features

Research in automatic genre classification has mainly focused on extracting features from the textual content of web pages. However, web documents in some genre classes can differ in structure and layout. These structural differences between genre categories could be exploited to improve genre classification of web pages. Thus, employing algorithms which capture visual similarity between different documents could assist us in obtaining better genre classification results. One of the robust advantages of visual features is that they are language independent and can be directly applied to web pages in other languages as long as web pages which belong to the same genre class do not vary visually across different cultures.

There are a few studies that investigated whether visual features of web pages can improve genre classification. Kim and Ross [72] present classification models mainly based on two groups of features which are referred to as image and style. These models were examined on six genre classes (academic monograph, business report, book of fiction, minutes, periodicals, and thesis) of the KRYS I corpus. The image features are low resolution bit maps of the first page of the web documents, while the style features are based on genre-specific words (prolific words explained in Section 2.6.2.3). The best reported result in this experiment was produced by style features. Although image features did not perform well over all, they showed the best recall rate for identifying theses and periodicals genre classes. Therefore, the image classification is more appropriate for genres which are different in terms of layout and similar in terms of style and content.

Levering et al. [85] also investigated whether visual features of web pages can improve genre classification. They argue that while using HTML tags to capture the number of visual elements such as tables, figures and videos might achieve some improvement over text alone in classification, including features that capture the layout characteristics of the genres should be more beneficial. In their research, they conducted experiments on a corpus that consists of web pages of e-commerce stores classified into three classes: store home pages, product lists and product descriptions. They showed that adding visual features to lexical and HTML tags resulted in improving the genre classification accuracy.

Therefore, it seems that extracting visual features that capture the structural differences between genres should be helpful in genre classification. However, this approach is only appropriate for genres which are dissimilar in terms of layout.

2.7 Genre Classification: Algorithms

The majority of the work in AGI is based on classification with a standard supervised algorithm. As shown in the previous section, various combinations of features and supervised learning algorithms (e.g. decision trees, naive bayes and SVM) have been employed for genre classification. However, this section focuses on the machine learning techniques employed in AGI which go beyond the standard supervised algorithms.

2.7.1 Structural Learning Approach to Genre Classification

There are two general approaches to genre classification, namely the flat approach and the hierarchy classification approach. In the flat approach, one tries to directly categorize documents into genre classes without making use of the category hierarchy. In contrast, hierarchy classification methods try to exploit the hierarchy structure of the genre taxonomy in order to improve genre classification accuracy.

Wu et al. [163] were the first to explore a way of improving fine-grained genre classification using a hierarchy of genres. This study is based on the hypothesis that when the genre labels are organized in a hierarchy structure, deciding whether a document belongs to a genre in higher levels might be easier than at the lower levels. When a document is securely classified in a higher level, it can be further classified into finer grained categories e.g. the distinction between science fiction and mystery becomes easier when a document is first securely classified as fiction. They present a novel use of structural Support Vector Machine (SVM) which works based on measuring the distance between different genres in the hierarchy.

Two approaches are used to measure the distance between two genre classes: distance measures based on Path Length and distance measures based on Information Content. While measuring the distance between two nodes in a hierarchy using the path length approach is well-defined, the concept of information content of a genre is not straightforward. They used the distance measure based on information content suggested by Resnik [123] which requires measuring the frequency of a genre.

Wu et al. [163] proposed two ways of measuring the frequency of a genre. The first one is measuring genre frequency based on document occurrence. In other words, the frequency of a genre node in a hierarchy equals the number of all documents belonging to that genre including any of its sub nodes. The second approach is measuring the genre frequency based on the genre names. This method estimates the frequency of a genre as occurrence frequency of its name or label in a corpus as well as the occurrence frequencies of the labels of all its sub nodes in the hierarchy. Genre-annotated collections which are used in this experiment are the Brown Corpus, BNC, HGC and Syracuse. Moreover, character 4-grams are extracted as features because they out-performed any other features in the previous studies.

Their experiment on the 15-genre Brown Corpus using the structural SVM with path length measure achieves 55.40% which is significantly better than the result produced by a flat SVM classifier (52.40%). While the results on the Brown corpus were encouraging, using structural SVM for genre classification did not result in significant improvement on BNC, HGC and Syracuse corpora. Two main reasons are specified for lack of success in this experiment. First, it seems that if a genre hierarchy has sufficient depth (greater than 2), is visually balanced and the number of examples at each leaf node has the same distribution, the better results would be produced by structural SVM. Second, information content based measures do not perform very well because, measuring genre frequency based on genre label frequency could lead to low results as genre label frequency and class frequency of documents are not necessarily linked.

In order to prove the first reason, Wu et al. [163] deformed the Brown genre hierarchy in two ways. First, they flattened it by removing its second layer. Second, they skewed its visual and distributional balance. These changes on the Brown genre tree worsened the results as expected. The outcome of this study suggests that this proposed approach leads to a better result if we have a balanced hierarchical genre taxonomy with sufficient depth. However, based on the web genre taxonomies developed in the previous studies, this prerequisite seems unlikely to be achieved on the web.

2.7.2 Multi-label Genre Classification

Most research on the web genre classification has focused on labelling each web page with a single genre class. In other words, they considered genres as mutually exclusive categories. However, there are web pages which belong to more than one genre class [34, 126, 128, 132]. It is also possible that a web page does not belong to any particular genre class as we observed in the experiment by Crowston and Williams [33] described in Section 2.3.

In order to address these problems, Santini [130] has proposed a zero-to-multi genre classification scheme. In this method, a web page can belong to zero or one or several genres. She implemented a classification model based on this scheme which had two steps. The first step of this algorithm uses features based on grammatical analysis (described in Section 2.6.1.3) to infer the text types of the web pages by employing a modified form of Bayes' theorem, called the odds-likelihood or subjective Bayesian method. In the second step, a set of if-then rules which uses information from the combination of text types produced in step one and other features such as linguistic and HTML features, was created in order to identify the genre classes of the web pages. However, she could not test this algorithm for zero-to-multi genre classification due to the absence of a genre benchmark with zero-to-multi genre annotation.

Inspired by Santini's [130] proposal, Vidulin et al. [160] built the Multi-labelled Genre corpus (MGC). This collection is the only web genre corpus which allows multi-labelling. They built two multi-label models based on problem transformation [158]. In the first model (referred to as multi-class transformation), they transformed a multi-label model into a multi-class single-label problem by treating different labels assigned to a web page as a single label. For example, web pages with two labels: *children's* and *informative* transformed into web pages with one label *children's-informative*. In the second model (referred to as binary transformation), they transformed the multi-label classification problem into several single-label binary classification problems (e.g. blog or non-blog, informative or non-informative). They reported that the first model (i.e. multi-class transformation) correctly classified a higher number of examples than the binary transformation. The exact match ratio for the first model was 38%, while it was only 29% for the second model built on the MGC corpus.

Although the study by Vidulin et al. [160] is the first step towards multi-label genre classification, it does not exactly comply with the zero-to-multi genre classification scheme proposed by Santini [130], because none of the web pages in the MGC corpus has zero label. In other words, the ability of these classifiers to assign no genre label to a web page could not be tested due to the lack of such data.

2.7.3 Open-set Genre Classification

Machine learning classification models can be divided into two categories: closed-set and open-set. In a closed-set model, an instance in a test set is classified into one or more predefined classes. However, an open-set classification model has the capability to detect instances which do not belong to any of the predefined classes. In other words, a closed set classifier which distinguishes between two or more classes assumes that it has seen all the possible classes in the training set, whereas an open-set classifier's assumption is that there are other classes in the test set that we might have not encountered in the training set.

In order to apply closed-set algorithms on the real web, we need to create a dataset which contains all the possible genre classes. However, considering the number of genre categories listed in the web genre classification schemes (e.g. 292 genre classes presented in [32]) creating such a dataset is very expensive or even completely impractical. In addition, the web is an evolving phenomenon and new genres are emerging all the time. Therefore, some researchers proposed open-set classification algorithms for AGI which do not require a complete set of genre labels and, as a result, are more suitable for genre classification on the web. It must be noted that open-set genre classification algorithms could be considered a sub-set of zero-to-multi-genre classification scheme proposed by Santini [130] (Section 2.7.2). Because the main property of these open-set classification algorithms is that they are capable of rejecting to classify an instance to the predefined categories. In other words, they can assign zero label to the instances which are detected as outliers.

Pritsos and Stamatatos [116] investigated open-set genre classification on two datasets: Santinis and KI04. They examined one-class SVM and random feature sub-spacing ensemble (RFSE) algorithms as open-set classification techniques for AGI. One-class SVM differs from binary or multi-class SVM because it learns from only the positive examples. One-class classification or outlier-detection techniques are used for two reasons: first, only positive examples are available and second, having a complete set of negative classes is not feasible due to the diversity of such classes. The second reason is the main rationale for using one-class SVM in AGI.

In the first open-set genre classification algorithm (i.e. one-class SVM) proposed by Pritsos and Stamatatos [116], one one-class classifier is built for each genre class available in the training corpus. Then, in the test phase, the final label that this algorithm assigns to each test instance is the one whose classifier has the highest positive distance from the hyperplane. In cases where all the classifiers return a negative distance which indicates that the web-pages do not belong to any of the predefined genre classes, the algorithm refuses to assign any pre-defined genre label to these web pages.

The second open-set genre classification algorithm proposed by Pritsos and Stamatatos [116] is the random feature sub-spacing ensemble (RFSE). In the first phase of the RFSE algorithm, a centroid vector for each genre is constructed by averaging all the feature frequency vectors of the training instances for each genre. Then, a number of classifiers are learned based on random feature sub-spacing (i.e. a number of random subsets of the full feature set). Each classifier uses the cosine distance to each centroid vector in order to predict the most likely genre. For each test instance, a score for each genre class is computed based on the number of times it was selected as the genre of the web page in the sub-spacing procedure. The final label is the one which has the highest score as long as it is bigger than a certain threshold. If the score of the final label is smaller than the threshold, the web page is detected as an outlier or “Don’t know”. For these experiments, Pritsos and Stamatatos [116] showed the precision values for 11 standard recall levels with respect to different values of parameters in these algorithms. The results indicates that RFSE significantly outperforms the one-class SVM algorithm.

In another study, Stubbe et al. [152] proposed a cascading classifier which can be used as an open-set classification algorithm. A cascading classifier is a sequential ensemble of binary classifiers ordered based on a particular selection scheme. The first binary classifier has the advantage to classify all the test instances and assign the final label to the web pages which it classifies as positive. Then all the web pages classified as negative are passed to the next binary classifier. This process continues to the last binary classifier. A test instance can remain unclassified if it is not positively identified by any of the binary classifiers.

In Stubbe et al. [152] the binary classifiers are ordered based on their performance in the training set. They chose two different selection scheme to order the binary classifiers. In the first method, they used F-measure values and the genre class with highest F-measure in the training set is the first classifier to label the test web pages. In the second method, they used ordering by dependencies and recall which is a much more sophisticated approach. In this method, a preliminary sequence of classifiers is ordered by the recall value of these classifiers on the training data set. In the second step, a dependency graph is generated using the confusion matrices of these classifiers. Then this dependency graph is used to rearrange the ordering of the binary classifiers.

The comparison between these two approaches (i.e. ordering by the F-measure and ordering by recall and dependencies) in this study shows that the second approach yields better results. Stubbe et al. [152] tested this approach on the HGC corpus and compared it with other classification algorithms such as SVM and Decision tree. They reported that their cascading classifier produced the best result with an average precision of 72.2% and average recall of 54.0%.

Although the algorithms proposed by Pritsos and Stamatatos [116] and Stubbe et al. [152] are open-set, their capability to detect outliers is not tested, because none of these datasets have web pages with zero label and as a result, the outlier detection of these algorithms can not be investigated. Therefore, in order to remedy this gap, in Chapter 6, we will test the open-set classification algorithm proposed by Stubbe et al. [152] on a random web genre dataset which has a considerable number of web pages with zero label.

2.8 Investigating the Impact of Topic and Authorship on Genre Classification

Genre and topic are often regarded as orthogonal and independent in the field of text classification [48]. Documents which belong to the same genre class can be about different topics and vise versa. As noted in section 2.6.2.3, Sharoff et al. [139] showed that in some web genre corpora, the features learned by the genre classifiers are mainly topical because in these collections, there is a false correlation between the genre and topic classes. Since topic is a factor which could influence the performance of genre classifiers, it must be considered when we evaluate genre classification algorithms. Based on this rationale, Finn and

Approach	Features	Corpus	Performance
Sharoff et al. [139]	POS trigrams	KI-04	63.40% acc
Santini [131]	genre-specific-word , 50 most common words, POS tags, punctuation, HTML tags	KI-04	68.9% acc
Eissen and Stein [104]	POS tags ,HTML tags, punctuation marks, text statistics, hand-selected genre words	KI-04	70.0% acc
Eissen and Stein [148]	word unigrams with feature selection	KI-04	80% average F-Measure
Kanaris and Stamatatos [63]	character n-grams of variable length	KI-04	82.8% acc
Kanaris and Stamatatos [63]	character n-grams of variable length, HTML tags	KI-04	84.1% acc
Sharoff et al. [139]	character 4-grams	KI-04	85.8% acc
Santini [131]	syntactic patterns and functional cues	SANTINIS	84.28% acc
Santini [131]	POS trigrams	SANTINIS	86.50% acc
Santini [131]	genre-specific-word , 50 most common words, POS tags, punctuation, HTML tags	SANTINIS	90.6% acc
Mason et al. [95]	character n-grams	SANTINIS	94.6% acc
Kanaris and Stamatatos [63]	character n-grams of variable length	SANTINIS	96.2% acc
Kanaris and Stamatatos [63]	character n-grams of variable length, HTML tags	SANTINIS	96.5% acc
Sharoff et al. [139]	character 4-grams	SANTINIS	97.1% acc
Kim and Ross [74]	word distribution patterns	SANTINIS	96% acc
Kim and Ross [74]	word distribution patterns	subset of KRIS I	80% acc
Stubbe and Ringlstetter [151]	hand-selected features	HGC	72.2% precision and 54.0% recall
Sharoff et al. [139]	character 4-grams	HGC	65.51% acc
Sharoff et al. [139]	POS trigrams	BNC	65.66% acc
Sharoff et al. [139]	word unigrams	BNC	75.03% acc
Sharoff et al. [139]	character 4-grams	BNC	74.54% acc
Sharoff et al. [139]	POS trigrams	10-genre Brown	59.40% acc
Karlgren and Cutting [67]	POS frequencies and text statistics (the whole corpus was used for training and testing)	10-genre Brown	65% acc
Sharoff et al. [139]	word unigrams	10-genre Brown	64.00% acc
Sharoff et al. [139]	character 4-grams	10-genre Brown	65.80% acc
Wu et al. [163]	character 4-grams and structural learning approach	10-genre Brown	68.80% acc

Table 2.4: Comparing genre classification results on the most frequently used corpora reported in prior work(acc is abbreviation for accuracy).

Training set			Test 1			Test 2		
	Fact	Opinion		Fact	Opinion		Fact	Opinion
Football	X	X	Football	X	X	Football		
Politics			Politics			Politics	X	X
Finance			Finance			Finance	X	X

Table 2.5: The experimental set up of single-domain (Test 1) and domain transfer (Test 2) for genre classification in Finn and Kushmerick [48].

Kushmerick [48] suggested the notion of domain transfer for the first time for the evaluation of genre classifiers. The aim of domain transfer is to assess whether genre classifiers trained on documents about one topic can successfully be applied to documents about other topics.

To evaluate the genre classifiers with different features in terms of domain transfer, Finn and Kushmerick [48] conducted an experiment to distinguish newspaper articles into fact and opinion categories(i.e. objective or subjective). First, they trained and tested on the same topic (Table 2.5 Test 1) with three different features: word unigrams (words were stemmed and function words were removed); part of speech tags; and text statistics which comprise features such as function words, sentence length and number of words. In this experiment, word unigrams performed best in all the domains. In the second experiment, they performed a domain transfer (i.e. trained on one topic and tested on another topic which is illustrated in Table 2.5 Test 2). The results of this experiment show a decrease in the performance of all the features (average accuracy for word unigrams fell from 87.2% to 67.3% , for POS tags from 84.7% to 78.5% and for text statistics from 83.2% to 67.8%). However, on average the performance of word unigrams experienced the largest decrease in accuracy.

Inspired by the work of Finn and Kushmerick [48], Petrenz and Webber [112] examined the performance of AGI systems when there is a complete shift in topics. Their study was conducted on the three genre classes news reports (News), editorials (Edit) and letters to the editor (LttE) from the New York Times Annotated Corpus (NYTAC) [129], on three different topics: education (Edu), defence (Def) and medicine (Med). In their two set of experiments, first they examined the performance of different feature sets on a dataset where each genre class is on a different topic (Table 2.6: training set and test set 1). The result of the first ex-

	Training set				Test set1				Test set2		
	News	Edit	LttE		News	Edit	LttE		News	Edit	LttE
Edu	X			Edu	X			Edu		X	X
Def		X		Def		X		Def	X		X
Med			X	Med			X	Med	X	X	

Table 2.6: The experimental set up in Petrenz and Webber [112]

	Training set				Test set		
	News	Edit	LttE		News	Edit	LttE
Edu	X	X	X	Edu			
Def				Def	X	X	X
Med				Med	X	X	X

Table 2.7: The correct way of domain or topic transfer experimental set up for the study by Petrenz and Webber [112] which they did not implement.

periment confirms that binary character 4-grams and word unigrams are the best features in AGI compared to other sets of features.

In the second part of their experiment, the genre-topic distributions of test set completely differ from the training set (Table 2.6: training set and test set 2). It must be noted that this experiment is not a domain transfer as proposed by Finn and Kushmerick [48] because in the domain transfer the training data should be all on the same topic and the test data on another topic (Table 2.7). The results of the second experiment show that the performance of both character 4-grams and word unigrams is significantly worse (about 50% drop) compared to the first experiment, whereas this change has less impact on POS feature. The reason behind such a dramatic drop in the genre classification accuracy based on features such as word unigrams and character 4-grams is that the documents in different genre classes in the training set also differ in terms of topic (e.g. genre class g_1 on topic t_1 and genre class g_2 on topic t_2). Therefore, the model based on this training data learns both genre revealing features and topic revealing features. However, if the experiment was conducted based on topic transfer (Table 2.7), the decrease in the performance of lexical features should be less dramatic because the influence of topic is kept constant across the different genre classes in the training set.

However, topic is not the only factor that can influence the performance of genre classification algorithms. Other factors such as authorship can also affect the results of genre classifiers. Features such as function words or common words which are popular in genre classification because they are topic independent, are also used in authorship classification (e.g. [3]) because they can capture the style of the authors.

Petrenz [111] investigated the performance of lexical and POS-based features when the style of writing changes. In a three tiers experiment, Petrenz [111] used three genres: news, editorial and letters to the editors from the New York Times corpus for the period between 1987 and 1998 (NYT 87-98) for training and testing (test 1) and tested the same model on the New York Times corpus for the period between 2003 and 2007 NYT 03-07 (test 2) and Wall Street Journal [93] corpus (test 3) in order to compare the performance of word unigrams and POS-based features. The results of these experiments show that word unigrams outperformed POS-based features in all three tests with quite a big margin. Although both word unigrams and POS-based models experienced a decrease in accuracy in test 2 and 3 compared to test 1, this drop was much larger for POS-based feature sets. Since these three datasets have different styles of writing (NYT 87-98 and NYT 03-07 used two different versions of the New York Times manual of style), the results show that word unigrams can cope much better with changing style. Therefore, in contrast to the results of studies by Finn and Kushmerick [48] and Petrenz and Webber [112] which show large decreases of accuracy of word unigrams models when tested on a new topic, the result of the study by Petrenz [111] shows a stable performance of this feature set when it is tested on a completely new datasets with different writing styles. On the other hand, the performance of POS-based features dramatically decreased.

Although exploiting the natural correlation between topic and genre classes (e.g. news articles are more likely to be about politics than science) could help to improve the result of automatic genre classification, the studies by Petrenz and Webber [112], Finn and Kushmerick [48] and Petrenz [111] emphasise that the performance of genre classifiers should be evaluated in a topic and style transfer scenario. Therefore, in order to truly evaluate the performance of a genre classification technique, we must train on a data set which all its documents are about the same topic and written by the same person. And test on a dataset which is on a completely different topic and written by another person (i.e. topic and style

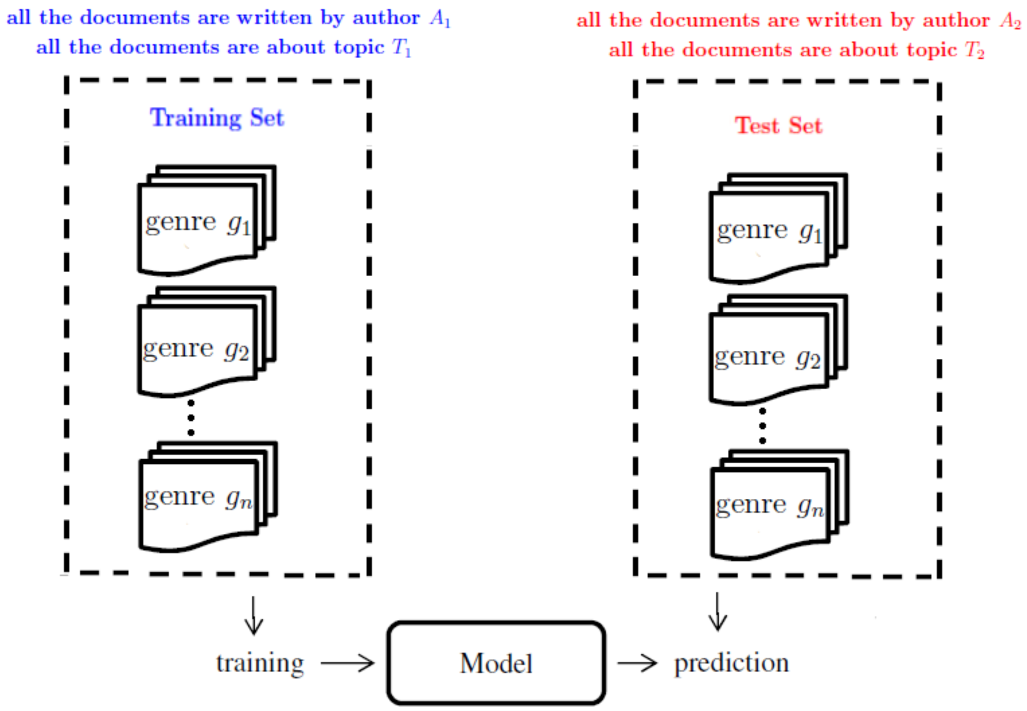

all the documents are written by author A_1
all the documents are about topic T_1
Training Set
genre g_1
genre g_2
genre g_n
all the documents are written by author A_2
all the documents are about topic T_2
Test Set
genre g_1
genre g_2
genre g_n
training
Model
prediction

Figure 2.1: Topic and style transfer: Authorship and topic are kept constant across the genre classes in the training data. The evaluation is performed on a dataset which is on a completely different topic and written by another person.

transfer: Figure 2.1). This way, we reduce the influence of these factors on genre classification by keeping them constant across the genre classes in the training data and as a result it is more likely that genre classification models can capture the genre of documents without being influenced by their topics or the style of their authors. However, constructing such a dataset is practically very hard or even impossible for genre classes on the web. The other more practical solution to this problem would be to collect data from various topics and sources in order to minimize the impact of these factors on genre classification.

2.9 Conclusions

In this chapter, we investigated the previous approaches to automatic genre identification. Various types of features such as common words [146], function words [2], word unigrams [50], character n-grams [64,95], part-of-speech tags [46,67] , part-of-speech trigrams [2, 131], document statistics (e.g. average sentence length, average word length and type/token ratio) [48, 70], HTML tags (e.g. [104, 131]) have been explored in AGI. However, researchers conducted genre classification experiments with different features on different corpora with different sets of genre labels. As a result, it is difficult to compare them. Although Sharoff et al. [139] examined a wide range of word-based, character-based and POS-based features on the existing genre-annotated corpora and showed that lexical features such as word unigrams and character 4-grams are the best performing features in AGI, they concluded that the results cannot be trusted because of two main reasons. First, some of these results are not trustworthy because of the low inter-coder agreement of some of these collections. Second, the spurious correlation between topic and genre classes in some of these corpora affected the results. Therefore, the question which set of features produces the best result in automatic genre classification on the web is still an open question. Therefore, one of the aims of this thesis is to answer this question. However, in order to fulfil this aim, we need a web genre corpus which overcomes the shortcomings of existing genre-annotated web corpora (e.g. having low reliability and having false correlation between topic and genre classes). The next chapter describes designing and constructing such a corpus.

Based on the definition of genre that we adopted in this thesis (Section 2.2), genre classes must be socially recognizable. Therefore, people must be able to reliably distinguish a genre class from other genre classes. This is the rationale behind the annotation study described in the next chapter. Also, based on the lessons that we learned from studies by Petrenz and Webber [112], Finn and Kushmerick [48] and Petrenz [111] which demonstrate the influence of topic and authorship on genre classification (Section 2.8), we reduced the impact of these factors on genre classification to the minimum by collecting data from a diverse range of sources and topics. Then, in Chapter 4, we employ this corpus to compare the performance of various features in AGI.

Another lesson which we learned from the literature review is that, as noted in Section 2.7.3, multi-class closed-set classification algorithms are not adequate for genre classification on the web, because they require a corpus which contains all the possible genre classes on the web and constructing such a corpus is very expensive or even completely impractical. Therefore, open-set classification algorithms which are capable of refusing to classify a web page to the predefined categories are needed to be employed on the web. Although open-set classification algorithms have been proposed for genre classification (e.g. [116] and [152]), they are not tested on a data set with web pages which do not belong to any of predefined genre categories. Therefore, Chapter 6 fills this research gap by constructing such a corpus and testing an open-set classification algorithm on it.

Chapter 3

Corpus Construction: Designing a Reliable Genre-annotated Corpus

3.1 Motivation

In Chapter 2, we noted the main shortcomings of the existing publicly available genre-annotated web corpora which are:

- having low inter-coder agreement in KRYS I [16], MGC [159], I-EN-Sample [140] and the corpus in [43] which shows that these collections are not reliable. Unreliability of the data significantly undermines the experimental results and the conclusions drawn from it.
- compiled from topically similar sources which is the problem especially in SANTINIS [131] corpus. The study by Petrenz and Webber [112] and Sharoff et al. [139] emphasize that the impact of topic on genre classification must be somehow eliminated or controlled. As we explained in Section 2.8, a practical solution to this problem would be to collect data from various topics and sources in order to minimize the impact of topic on genre classification. This solution also controls the impact of other factors such as the style of the authors and the design of web pages on genre classification.
- having a small number of web pages per genre category (Table 2.2). Ma-

chine learning algorithms usually need a large amount of data in order to capture similarities and differences between the classes.

- preserved in formats such as plain text or PDF (I-EN-Sample [140] and KRYS I [16] corpora) which resulted in losing HTML tags.
- annotated by experts which can mislead with regards to the general applicability of the annotation scheme, especially if these are the same experts the ones who developed the theoretical terms and concepts underlying the annotation scheme [124]. This was the case for example in SANTINIS, MGC and KI-04.

These drawbacks highlight the necessity of developing a web genre-annotated corpus which eradicates these defects. The need for such a corpus is also acknowledged by other researchers in this field (e.g. [73] [122]). Therefore, we have built Leeds Web Genre Corpus Balanced-design (LWGC-B) which fulfils the following criteria:

- It is reliably annotated for genre as measured by chance-corrected agreement, focusing on 15 genre classes that are common to the web.
- It avoids circularity (i.e. annotating the corpus using the same experts who developed the annotation guidelines) by crowd-sourcing naive annotators that were not involved in annotation scheme development [124].
- It is collected from a diverse range of sources and topics in order to minimize the impact of factors such as topic and authorship on genre classification.
- It has large number of web pages per genre category.
- Web pages have been saved in HTML format. Also the appearance of each web page has been preserved by taking a screen shot of its whole content. The latter can facilitate using visual features as well as textual and HTML features in AGI.

The next section describes different steps of building this corpus.

3.2 LWGC-B: A Web Genre Corpus Designed via Focused Search

3.2.1 Genre Inventory

The quality of manual annotations depends on the use of precise and consistent guidelines which include the definitions of the categories. Therefore, the development of the annotation guidelines must be seen as one of the crucial tasks in annotation projects. The vagueness and ambiguity in the annotation guidelines, especially the definition of the categories which increases the subjectivity of the annotation task, could be the reason for low inter-coder agreement in existing web genre corpora. For example, annotators may have different interpretation of broad and vague categories such as article in KI-04 [104], informative and entertainment in MGC corpus [159] and information and reporting in I-EN-Sample [140]. Defining broad genre categories not only could cause disagreement between annotators, but it could also have a negative impact on automatic genre classification. Kessler et al. [70] emphasise this point by saying that a genre should not be so broad that the texts belonging to it do not share any distinguishing properties.

> “we would probably not use the term genre to describe merely the class of texts that have the objective of persuading someone to do something, since that class which would include editorials, sermons, prayers, advertisements, and so forth has no distinguishing formal properties.” ([70, p. 33])

We therefore chose quite specific genre names which are in common use (such as blog) with short 1-2 sentence descriptions. With regards as to which genres to include in our inventory, we started with the genre classification scheme used in KI-04 corpus with eight genre classes which was developed by asking a group of students to fill in a questionnaire about typical topics for queries and favourite genre classes. However, some of the genre classes in this classification scheme are very broad and vague such as article. As noted above, we need to avoid such categories in our annotation guidelines. Therefore, we divided the category article into more well-defined categories such as news, editorial, biography, interview, story, personal blog and review. We also broke down the genre category help in KI04 into frequently asked questions, instruction or how-to and recipe. Moreover, we split the genre category non-personal home pages into company/business homepage and educational organization homepage. Since we are interested in textual properties of genres, we excluded download and link collection genre classes which do not contain much text. Therefore, we concentrated on 15 genre classes, which of course is not exhaustive. We explore coverage of our scheme on the web in Chapter 6.

Table 3.1 shows the set of 15 genre labels and their definitions, while Table 3.2 shows how these 15 selected genre classes correspond to those used in other genre-annotated corpora. However, since different genre-annotated corpora used different genre classes with different level of granularity, any one-to-one comparison between our genre labels and their genre classes can only be approximate. For example, the genre label journalistic in MGC can include several genres in our corpus such as news, editorial, interviews and reviews. Another example is the periodicals (newspaper, magazine) category from the KRYS I corpus which is very broad and can include many genre classes such as recipe, interview and reviews.

3.2.2 Corpus Compilation

The next step after defining the categories is corpus compilation. Web corpora are categorized into two subtypes, i.e. designed and random, which are different in terms of how they are collected [71]. The content of a designed corpus is selected based on its design specification whereas there is much less control on the content of a corpus constructed by random selection. HGC [151] and UKWac [9] are examples of designed and random corpora respectively. we use a designed corpus as the first step for testing our annotation scheme and crowd-sourcing effort, for two reasons. First, we can provide a corpus with a large number of web pages for each category via this method. While collecting random web pages is fast and cheap, there is no guarantee that it fulfils this criterion. Second, random web pages could be noisy whereas manually collected clear and prototypical examples provide a good test bed for our annotation scheme for naive annotators in the first instance. It is also possible that these clear examples are better for training machine learners. The use of a designed corpus was also suggested by Rehm et al. [122] as an initial step when building a reference corpus of web genres. Unfortunately, their proposal stayed just that, a proposal and the authors did not actually follow up with their own web genre corpus following these guidelines.

On the flip side, a designed corpus where we look for pages of specified genres will not give us an accurate representation of the genre distribution on the

Genre	Definition
Personal Homepage (php):	created by an individual to contain content of a personal nature rather than on behalf of a company, organization or institution.
Company/ Business homepage (com):	the main web page of a company or an enterprise website which promote a product or a service. These web pages often contain a description of the purpose or objectives of the company.
Educational Organization Homepage (edu):	the main web page of an educational institution website. Examples are universities and schools home pages.
Personal blog /Diary (blog):	where people write about their day-to-day experiences (please only choose this option if the blog is personal and it is about personal experiences)
Online shops (shop):	Web pages created with intention to sell
Instruction/ How to (instruction):	contains instructions and teaches you how to do something (not recipes)
Recipe:	a set of instructions that describe how to prepare or make food
News Article (news):	a report of recent events
Editorial:	an opinion piece written by the editorial staff or publisher of a newspaper or magazine
Conversational Forum (forum):	where people have a conversation about a certain topic
Biography (bio):	a detailed description of someone's life
Frequently Asked Questions (faq):	listed questions commonly asked about a particular topic
Review:	an evaluation of a publication, a product or a service, such as a movie,a video game, a musical composition or a book
Interview	a conversation in which one or more persons question another person
Story	a narrative, either true or fictitious, with the aim to entertain the reader

Table 3.1: Definition of Genre labels. To save space, we use the abbreviation of genre labels which are specified in front of the genre names.

Genre	KRYS I	MGC	HGC	KI-04	SANTINIS	Syracuse
php		X	X	X	X	X
com		X				X
edu				X		
blog		X	X		X	X
shop		X	X	X	X	X
instruction				X		X
recipe						X
news	X		X			X
editorial			X			X
forum	X	X	X	X		X
bio	X		X			X
faq	X	X	X		X	X
review	X		X			X
interview	X		X			X
story	X	X	X			X

Table 3.2: This table illustrates which genre classes in LWGC-B are also included in previous genre-annotated corpora.

web nor an idea of the coverage of our annotation scheme. Annotation results on clear and prototypical web pages are also likely to overestimate inter-annotator agreement [140]. In other words, reaching a high inter-coder agreement for genre annotation could be more difficult on random web pages. We will investigate both issues in Chapter 6 where we collate and annotate a smaller, random corpus, the LWGC-R.

In order to obtain the wished-for balanced collection, we hand-selected web pages mainly from existing web directories, particularly the Yahoo Directory[1] and Open Directory Project[2] websites. We selected 3964 web pages from a diverse range of sources to avoid creating false correlation between topic and genre labels. We will discuss the source and topic diversity of the corpus further in Section 3.4. Since most researchers in AGI have used individual web pages as the unit for genre classification as noted in Section 2.5, we have also chosen to take individual web page as the unit of process in this thesis.

In the next phase, we used a tool namely KrdWrd [147] to download the web pages in HTML format. However, only saving a web page in HTML format does not guarantee to preserve the appearance of a web page. To achieve this aim, we could, for each web page, save all its graphics and style files, or take a screen shot of its whole content. We chose the second option and used KrdWrd [147] to

[1] http://dir.yahoo.com/

[2] http://www.dmoz.org/

preserve each web page as an image.

3.2.3 Annotation Procedure

After the compilation of the web pages, the corpus needs to be annotated with the set of chosen genre labels which can be a very time consuming and expensive task. However, in recent years, the advent of crowd-sourcing (e.g. via Amazon Mechanical Turk[3]) has facilitated annotation tasks and this phase can be done cheaper and faster than ever before. Amazon Mechanical Turk (MTurk) has been used for a variety of labelling and annotation tasks e.g. word sense disambiguation, word similarity, text alignment, temporal ordering [144]; machine translation [23]; building question answering dataset [62]. It has also been used for genre annotation by [43] but without reliable results (see Section2.5). In addition to saving expense and time, using naive annotators that are independent of the annotation scheme developers ensures easy re-use of the annotation scheme by different annotation teams and avoids circularity in annotation [124].

3.2.3.1 Amazon's Mechanical Turk

The Mechanical Turk web site provides a service which enables requesters such as researchers or companies to create and publish jobs also known as Human Intelligence Tasks (HITs). These HITs can be carried out by untrained MTurk workers (turkers) all around the world for a small amount of money. The main advantages of Mturk are low cost and efficiency in terms of the speed of task completion as well as its infrastructure, which allows the requesters to develop their HITs using standard HTML and Javascript.

With turkers, quality control is crucial in order to detect poor quality or randomly selected answers. Moreover, Mturk HITs like any other web-based interface are vulnerable to automated scripts also known as bots which are used by some turkers in order to maximize their income [96]. To improve our resulting annotation, we used two types of qualification criteria added to our HIT design, as provided by MTurk. The first type is known as "system qualifications", and is independent of the specific task one creates. They include HIT submission rate (the percentage of submitted HITs by the turker), HIT approval rate (ratio of accepted

[3] https://www.mturk.com/mturk/welcome

HITs compared to the total number of HITs submitted by the turker), HIT rejection rate (ratio of rejected HITs compared to the total number of HITs submitted by the turker) and location (the worker's country of residence).

The second type of quality control measures is task-specific. It includes the possibility of a pre-task qualification test designed by the requesters. Up to five qualification criteria can be assigned to a HIT by the requester and only turkers who pass these qualification measures are permitted to complete the HITs. With regards to after-task quality control, Mturk enables the requesters to download and (automatically or manually) review the submitted works, then reject poor quality data and only pay for the HITs which they approve. In the next section which describes our HIT design, we use both system qualifications and task-specific pre- and after-task quality controls in order to ensure the quality of the annotations.

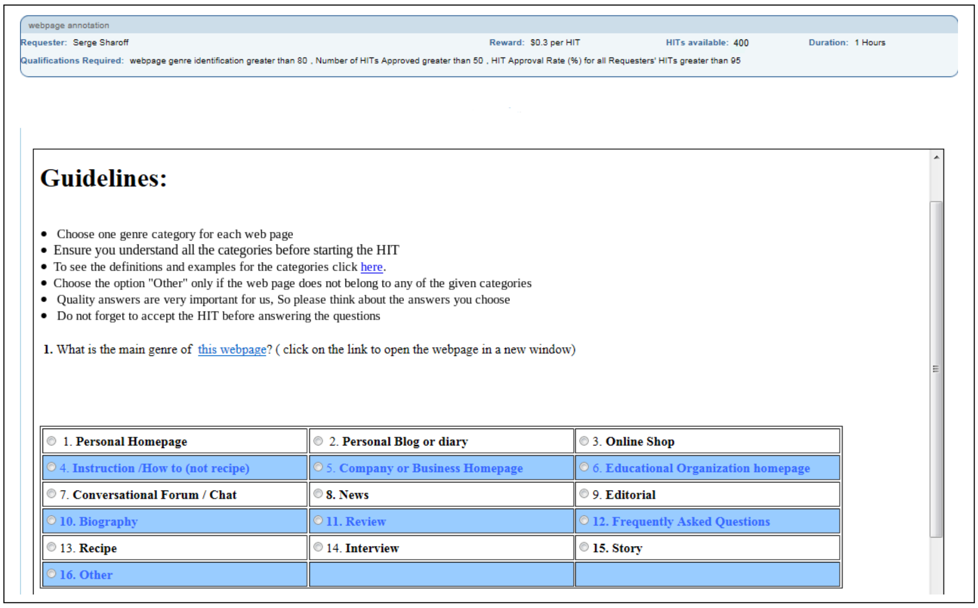
3.2.3.2 HIT Design and Quality Control

This section describes the details of HIT design and quality control measures which were developed to ensure obtaining good quality data.

HIT Design. Turkers were presented with a list of our 15 genre categories (see Table 3.1) as well as very short guidelines that allowed them to view category definitions (again as in Table 3.1) as well as example pages for the categories. As our genre inventory is not exhaustive, annotators were also allowed to choose the option *other* for web pages that do not fit any of the 15 classes. In order to keep the annotation task simple, we decided to choose the single-labelling method, i.e. each web page could only receive a single genre label, despite the fact that there are some web pages that might belong to more than one genre class [34] [70] [130]. Annotators needed to click on a link to open the cached web page that needs annotation in a separate window. Figure 3.1. shows a screenshot of the annotation task.

In our case, we include 10 web pages to be annotated in a single HIT, both as this is more time and cost-effective, and because we are going to use this feature in quality control as described below.

Quality Control. With regard to system qualifications, we restricted the range of workers who can complete our task. We only allowed workers who had completed at least fifty previously accepted HITs and have an approval rate higher



webpage annotation

Requester: Serge Sharoff Reward: $0.3 per HIT HITs available: 400 Duration: 1 Hours

Qualifications Required: webpage genre identification greater than 80 , Number of HITs Approved greater than 50 , HIT Approval Rate (%) for all Requesters' HITs greater than 95

Guidelines:

- Choose one genre category for each web page
- Ensure you understand all the categories before starting the HIT
- To see the definitions and examples for the categories click here.
- Choose the option "Other" only if the web page does not belong to any of the given categories
- Quality answers are very important for us, So please think about the answers you choose
- Do not forget to accept the HIT before answering the questions

1. What is the main genre of this webpage? (click on the link to open the webpage in a new window)

○ 1. Personal Homepage	○ 2. Personal Blog or diary	○ 3. Online Shop
○ 4. Instruction /How to (not recipe)	○ 5. Company or Business Homepage	○ 6. Educational Organization homepage
○ 7. Conversational Forum / Chat	○ 8. News	○ 9. Editorial
○ 10. Biography	○ 11. Review	○ 12. Frequently Asked Questions
○ 13. Recipe	○ 14. Interview	○ 15. Story
○ 16. Other		

Figure 3.1: Screen-shot of our genre annotation task on the Mturk website

than 95%, thus reasonably experienced and diligent workers.

As a task specific pre-task qualification test, we let turkers read the definitions and examples of genre classes and then complete a trial HIT of ten genre annotations on pages that we deemed highly prototypical and therefore should be annotated without much scope for error. Only turkers which completed this qualification test with the score of equal or higher than 80% were allowed to take part. This was supposed to weed out bots and random clickers.

To allow after-task quality control without excessive manual work and without introducing substantial expert bias, we used one of the ten web pages to be annotated per HIT as a "trap" question. A set of twenty web pages from different genre classes in our dataset which the author of this thesis judged as unambiguous and clear example of one of our predefined genre categories, were selected as gold standard and used as trap questions. We performed semi-automated monitoring of the annotations by checking the answers to the trap questions and rejected the work from workers who gave wrong answers to the trap questions more than 80% of the time.

Because adding more annotators can help to reduce annotation bias, it is encouraged in human annotation projects to have as many annotators as possible [11]. We chose to have five annotations per web page because Snow et al. [144] compared the quality of annotation done by experts and Mturk workers and concluded that an average of 4 non-expert workers in Mturk often provides expert-level label quality.

3.2.4 Inter-coder Agreement Measures

In Natural Language Processing and machine learning, a reliably annotated dataset plays a crucial role. Reliability of annotated data is an essential factor for reliability of the research result. In other words, the results of research based on unreliable annotation can be considered as untrustworthy, doubtful and even meaningless. In order to be able to measure the reliability of annotation, different annotators judge the same data and the inter-coder agreement is calculated for their judgements. The most commonly used inter-coder agreement measure which employed to measure the extent of consensus in judgements among annotators are: percentage agreement, S [12], Scott's π [135], Cohen's κ [27] and Krippendorff's α [81] (see [4] for a comprehensive survey on inter-coder agreement measures).

Percentage or observed agreement which is the most commonly used measure of agreement among coders can be computed by measuring pairwise agreement based on Fleiss [49] definition. If we define n_{ik} as the number of times an instance i is annotated as category k, then the sum of $\binom{n_{ik}}{2}$ for all categories is all the pair agreement for item i. Then if we divide this value by all the pairwise combination of all coders$\binom{c}{2}$, the final value is the amount of agreement for instance i.

$$agr_i = \frac{1}{\binom{c}{2}} \sum_k \binom{n_{ik}}{2} = \frac{1}{c(c-1)} \sum_k n_{ik}(n_{ik}-1) \tag{3.1}$$

Then the mean of agr_i for all items i is the overall observed agreement (note that i denotes the total number of items).

$$A_o = \frac{1}{\mathrm{i}} \sum_i agr_i \tag{3.2}$$

Although the computation of observed agreement is not complicated, this measure cannot be trusted because it does not take into account the agreement which is expected to happen by chance and as a result it can overestimate the true agreement. Therefore, in order to overcome the shortcoming of percentage agreement, other inter-coder agreement measures such as Scott's π or Cohen's κ which correct for chance agreement must be computed. Originally these coefficients were proposed for calculating inter-coder agreement between two annotators. Then Fleiss [49] proposed a generalization for Scott's π and Davies and Fleiss [35] gave generalization for Cohen's κ.

In order to calculate π and κ, first we need to compute expected agreement (A_e). For calculating expected agreement for π we only take into account the combined judgements of all coders and not the number of items assigned to each category by each individual coder. Therefore, if we define n_k as the number of times an instance has been assigned to category k regardless of who assigned it, the probability that an arbitrary coder assign an item to category k (Equation 3.3) is the overall proportion of items assigned to this category (n_k) divided by the overall number of assignments (number of instances i multiplied by the number of coders c).

$$P(k) = \frac{1}{ic} n_k \tag{3.3}$$

Because we assumed the independence of coders, the probability that any two coders assign an instance to the same category is the joint probability of each coder making this assignment alone. Therefore, we can compute the expected agreement(A_e^π) as the sum of this joint probability over all the categories.

$$A_e^\pi = \sum_{k \in K} (P(k))^2 = \frac{1}{(ic)^2} \sum_{k \in K} n_k^2 \tag{3.4}$$

Unlike π , for calculating expected agreement for κ, we take into account the number of times coder c assigns an item to category k which is denoted by n_{ck}. Therefore, the probability that an arbitrary coder c assign an item to category k is:

$$P(k|c) = \frac{1}{i} n_{ck} \tag{3.5}$$

Since all coders annotate all instances, we need to calculate the mean of the joint probability over all pairwise combination of coders. Thus the expected agreement for κ is defined as follows:

$$A_e^\kappa = \sum_{k \in K} \frac{1}{\binom{c}{2}} \sum_{m=1}^{c-1} \sum_{n=m+1}^{c} P(k|c_m) P(k|c_n) \tag{3.6}$$

Then the coefficients π and κ can be computed using the following formula:

$$\pi, \kappa = \frac{A_o - A_e}{1 - A_e} \tag{3.7}$$

Although these two measures are very similar and often have very close values, there is one difference between them. For calculating expected agreement for π we only take into account the combined judgements of all coders and not the number of items assigned to each category by each individual coder. Unlike π, for calculating expected agreement for κ, we take into account the number of times each coder assigns an item to a category.

Since in MTurk the annotations have been done by various workers, κ is not a good measure as it takes into account the proportion of items assigned by each annotator to each category. Therefore, like other annotation studies using crowdsourcing ([107], [99], [13]) we calculated the generalization of π also known as Fleiss's kappa [35] for the annotation. The next section presents the result of inter-coder agreement results.

3.2.5 Results of Annotation Study

The annotation task was completed within seven days with the total cost of $820. Overall 42 annotators participated in annotating the corpus in MTurk. The annotation study shows high agreement in the annotation results.The percentage agreement is 88.2% and π is 0.874. Based on the interpretation of the inter-coder agreement value by Landis and Koch [83] (Table 3.3), the π value for our annotation task shows perfect agreement between the annotators and therefore we can consider the annotation reliable.

We also computed π for each single category in order to identify the most and the least agreed on categories. Single category π measures the agreement for one target category and treats all other categories as one non-target category and measures agreement between the two resulting categories. Table 3.4 shows the results of inter-coder agreement measures for individual genre classes. Results show that recipe was the easiest category for the annotators whereas company/ business home pages caused the most disagreement between the annotators. However, π values for the individual categories illustrate substantial agreement among the coders and, as a result, annotations for all the genre classes are highly reliable. Overall we show that genre identification for the listed genre classes can be reliably anno-

generalization of π proposed by Fleiss [49]	Level of agreement
< 0	Poor
0.0 – 0.2	Slight
0.2 – 0.4	Fair
0.4 – 0.6	Moderate
0.6 – 0.8	Substantial
0.8 – 1.0	Perfect

Table 3.3: Landis and Koch interpretations [83] of generalization of π proposed by Fleiss [49]

tated and therefore is a well-defined task for automatic classification.

.

3.2.6 LWGC-B Gold Standard Corpus

The next phase of building a reliable genre annotated dataset for developing supervised machine learning classifiers is to convert the annotated dataset into a gold standard. There are a number of different methods to derive a gold standard from an annotated dataset [11]. For instance, the annotators can discuss together [89] to reach an agreement on the disagreed items or if more than two annotators are employed in the annotation task, a majority vote approach [161] can be employed on the disagreed items. Also, a domain expert can be used to decide the final label for the disagreed instances [54,145] or simply the instances which cause disagreement can be excluded from the dataset [11].

Since, we employed MTurk for annotation, reaching agreement through discussion between annotators is not possible. Therefore, as we have five annotations per web page, the majority vote strategy was employed to assign the final label to the disagreed web pages. There are seven possible types of inter-annotator agreement when there are five annotators (Table 3.5).

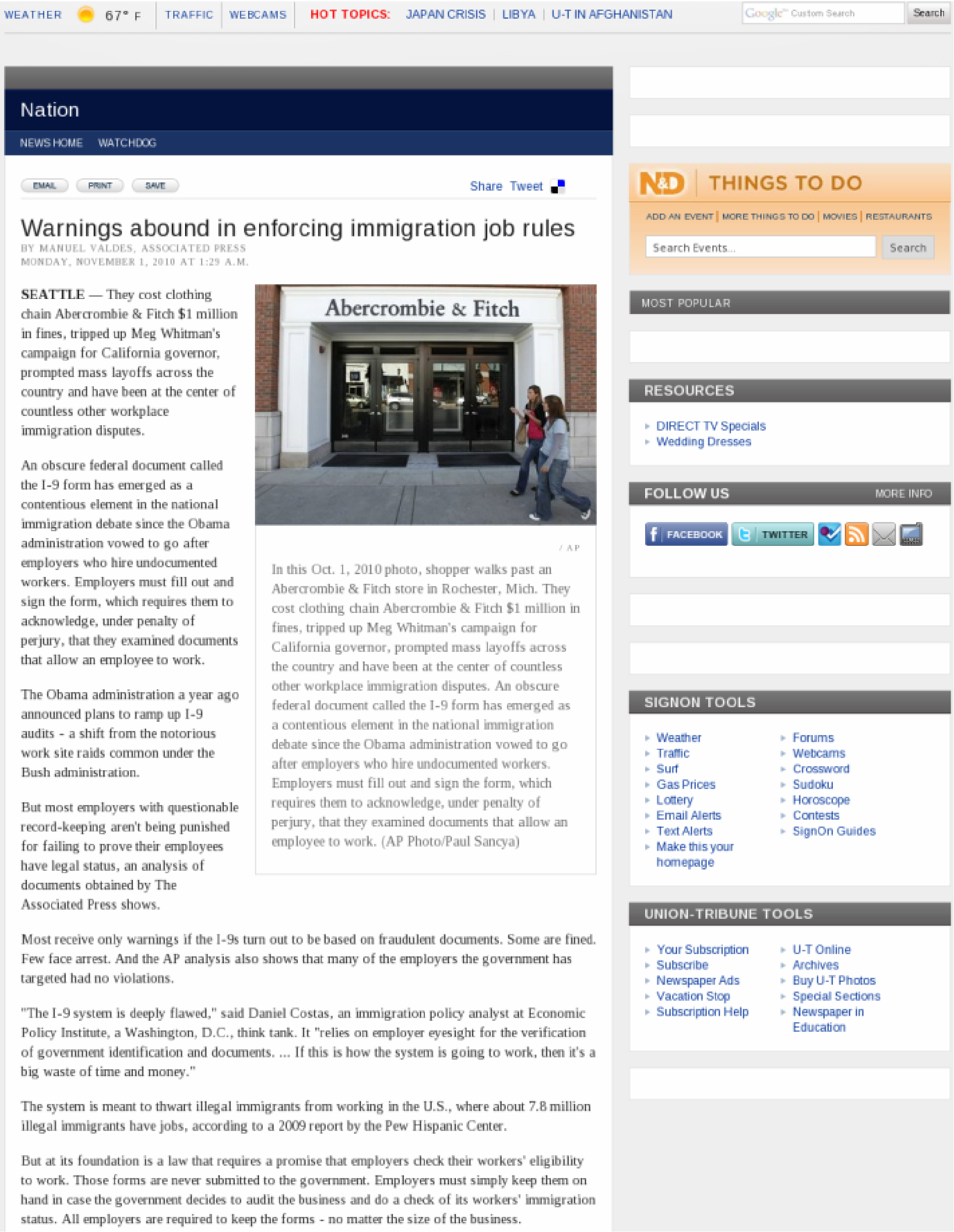
In order to analyse how often the annotators agreed with each other, we calculated the percentage of each type of inter-annotator agreement (Table 3.6). For more than 74% of the web pages all the five annotators agreed and for 95% of the data at least four annotators agreed which indicates high level of agreement

Genre Labels	Percentage agreement	π
Personal homepage	0.979	0.858
Company/ Business homepage	0.962	0.713
Educational organization homepage	0.993	0.953
Personal blog /Diary	0.977	0.812
Online shops	0.976	0.830
Instruction/ How to	0.985	0.871
Recipe	0.995	0.971
News article	0.970	0.801
Editorial	0.981	0.877
Conversational forum	0.994	0.951
Biography	0.988	0.905
Frequently asked questions	0.992	0.915
Review	0.984	0.880
Story	0.996	0.953
Interview	0.992	0.905

Table 3.4: Inter-coder agreements for individual categories in LWGC-B show substantial agreement among the coders. Therefore annotations for all the genre classes are highly reliable.

between the coders. Low percentage of the other five types of inter-coder agreement confirms the high value of π for the annotation task. Since we did not have majority vote for eight web pages, the final label for these instances were assigned by the author of this thesis.

Disagreements in cases where only three annotators agreed with each other while the other two disagreed with the majority but agreed with each other, are mainly caused by confusion between news and editorial and between shop and company home page. Figures 3.2 and 3.3 are examples of web pages which caused such disagreement. The web page in Figure 3.2 shows a news article which to some extent is argumentative. Two out of five annotators chose editorial label whereas the other three annotators chose the category news for this web page. Also, Figure 3.3 shows a web page which two out of five annotators categorized it as a shop whereas the other three annotators labelled it as a company homepage. This kind of disagreement is caused by two kinds of web pages: web pages that can belong to more than one genre category and border-line web pages which are ambiguous.



WEATHER 67° F | TRAFFIC | WEBCAMS | HOT TOPICS: JAPAN CRISIS | LIBYA | U-T IN AFGHANISTAN
Google™ Custom Search
Search
Nation
NEWS HOME WATCHDOG
EMAIL PRINT SAVE
Share Tweet

Warnings abound in enforcing immigration job rules

BY MANUEL VALDES, ASSOCIATED PRESS
MONDAY, NOVEMBER 1, 2010 AT 1:29 A.M.

SEATTLE — They cost clothing chain Abercrombie & Fitch $1 million in fines, tripped up Meg Whitman's campaign for California governor, prompted mass layoffs across the country and have been at the center of countless other workplace immigration disputes.

An obscure federal document called the I-9 form has emerged as a contentious element in the national immigration debate since the Obama administration vowed to go after employers who hire undocumented workers. Employers must fill out and sign the form, which requires them to acknowledge, under penalty of perjury, that they examined documents that allow an employee to work.

The Obama administration a year ago announced plans to ramp up I-9 audits - a shift from the notorious work site raids common under the Bush administration.

But most employers with questionable record-keeping aren't being punished for failing to prove their employees have legal status, an analysis of documents obtained by The Associated Press shows.

/ AP

In this Oct. 1, 2010 photo, shopper walks past an Abercrombie & Fitch store in Rochester, Mich. They cost clothing chain Abercrombie & Fitch $1 million in fines, tripped up Meg Whitman's campaign for California governor, prompted mass layoffs across the country and have been at the center of countless other workplace immigration disputes. An obscure federal document called the I-9 form has emerged as a contentious element in the national immigration debate since the Obama administration vowed to go after employers who hire undocumented workers. Employers must fill out and sign the form, which requires them to acknowledge, under penalty of perjury, that they examined documents that allow an employee to work. (AP Photo/Paul Sancya)

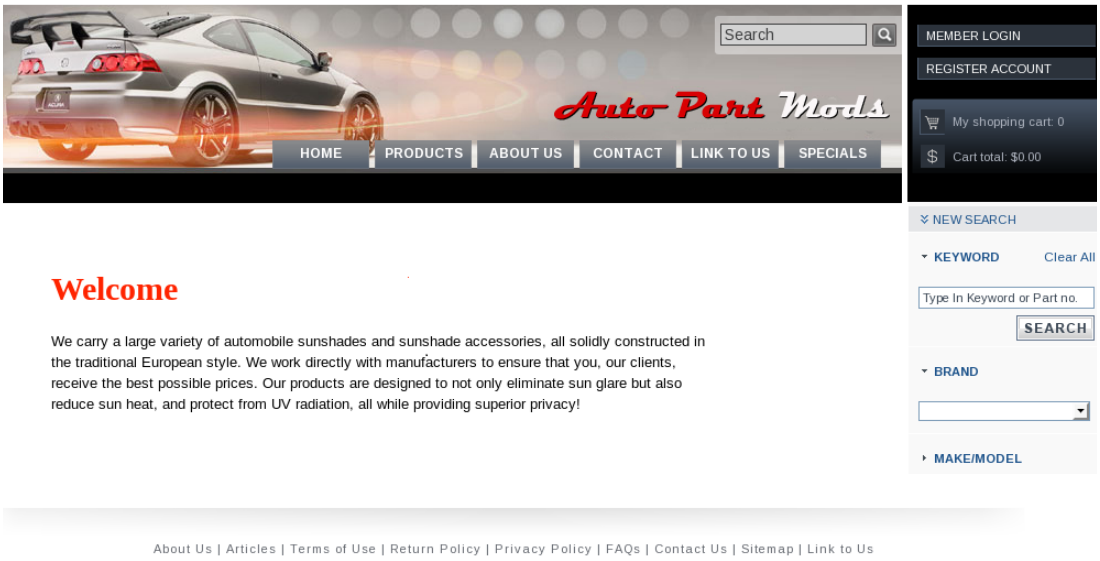
Most receive only warnings if the I-9s turn out to be based on fraudulent documents. Some are fined. Few face arrest. And the AP analysis also shows that many of the employers the government has targeted had no violations.

"The I-9 system is deeply flawed," said Daniel Costas, an immigration policy analyst at Economic Policy Institute, a Washington, D.C., think tank. It "relies on employer eyesight for the verification of government identification and documents. ... If this is how the system is going to work, then it's a big waste of time and money."

The system is meant to thwart illegal immigrants from working in the U.S., where about 7.8 million illegal immigrants have jobs, according to a 2009 report by the Pew Hispanic Center.

But at its foundation is a law that requires a promise that employers check their workers' eligibility to work. Those forms are never submitted to the government. Employers must simply keep them on hand in case the government decides to audit the business and do a check of its workers' immigration status. All employers are required to keep the forms - no matter the size of the business.

Figure 3.2: An example of web pages which cause confusion between the categories news and editorial. URL: http://www.signonsandiego.com/news/ 2010/nov/01/warnings-abound-in-enforcing-immigration-job-rules/



Welcome

We carry a large variety of automobile sunshades and sunshade accessories, all solidly constructed in the traditional European style. We work directly with manufacturers to ensure that you, our clients, receive the best possible prices. Our products are designed to not only eliminate sun glare but also reduce sun heat, and protect from UV radiation, all while providing superior privacy!

Figure 3.3: An example of web pages which cause confusion between the categories company home page and shop. URL: http://www.autopartmods.com

Types of inter-annotator agreement	Represented as
all agreed on a choice of category	5,0
four annotators agreed and the fifth disagreed	4,1
only three annotators agreed with each other while the other two disagreed with the majority as well as each other	3,1,1
only three annotators agreed with each other while the other two disagreed with the majority but agreed with each other	3,2
only two annotators agreed with each other	2,1,1,1
two annotators chose the same category and the other two annotators also chose the same category but different from the first two annotators	2,2,1
all five annotations differed	1,1,1,1,1,1

Table 3.5: All possible combinations of five annotations

Types of inter-coder agreement	# of web pages	% of web pages
5,0	2945	74.29%
4,1	791	19.95%
3,1,1	104	2.62%
3,2	116	2.92%
2,1,1,1	4	0.10%
2,2,1	4	0.10%
1,1,1,1,1	0	0%

Table 3.6: Distribution of different types of inter-annotator agreement for LWGC-B corpus

Number of genres	15
Number of web pages	3964
Number of web pages for the smallest category	184
Number of web pages for the largest category	332
Median number of web pages for the categories	266
Number of tokens	7,205,820
Number of types	130,254
Number of sentences	329,861

Table 3.7: The corpus statistics for LWGC-B

3.3 LWGC-B Corpus Statistics

In order to provide further insight into the constructed corpus, we computed some corpus statistics such as number of tokens, number of types and number of sentences. Table 3.7 gives an overview of the corpus statistics. The corpus consists of 3964 web pages, distributed across 15 genres. [4] It contains more than 7 million words which makes it approximately seven times bigger than the Brown corpus in terms of the number of tokens.

We also calculated these statistics for each genre category which are presented in Table 3.8. A number of interesting observations can be made from the individual categories' text statistics. First, the number of sentences in the home pages is much lower than for other genre categories in this corpus. On the other hand, personal blogs and interviews seem to contain the highest number of sentences. A

[4] Although individual annotators used the label other, it was never the majority annotation due to the focused search collection.

Genre	Number of											
	sentences			tokens			types			types/token ratio		
	max	min	med	max	min	med	max	min	med	max	min	med
php	326	0	11	4,165	21	241	1,232	17	142	1	0.22	0.57
com	195	0	11	4,906	32	330	1390	29	172	0.90	0.22	0.53
edu	179	0	10	5,501	12	396	1960	11	209	0.93	0.13	0.54
blog	1,041	14	139	19,488	214	2,905	2926	141	882	0.69	0.09	0.30
shop	600	0	33	25,651	71	1337	7,459	45	456	0.69	0.05	0.33
instruction	595	11	99	12,767	199	1,219	1,988	102	447	0.57	0.14	0.36
recipe	584	2	20	11,445	123	428	2,218	68	221	0.74	0.15	0.50
news	702	7	41	16,642	271	1312	3,052	140	603	0.64	0.15	0.45
editorial	511	9	45	10,537	311	1367	2,309	187	661	0.63	0.19	0.47
forum	619	2	60	13,010	269	1454	1,932	144	473	0.60	0.12	0.32
bio	2,465	4	67	23,838	198	1616	4,603	103	625	0.67	0.15	0.39
faq	613	5	54	14,312	119	971	2,220	68	355	0.73	0.13	0.36
review	1,107	12	96	19,261	174	2,094	2,979	118	634	0.73	0.15	0.31
story	1,012	10	98	10,521	239	1777	2,043	98	586	0.56	0.16	0.33
interview	1,243	29	150	19,687	380	2,487	3,601	153	733	0.50	0.13	0.30

Table 3.8: Corpus text statistics for individual categories in LWGC-B corpus. Max, min and med are abbreviations of minimum, maximum and median respectively.

closer look at the corpus statistics also reveals that home pages tend to have high type/token ratio compared to other categories. Based on these observations, it seems that automatic genre classification algorithms could benefit from the discriminative power of these statistics as features.

3.4 Investigating Topic and Source Diversity

Collecting data for a genre category from topically similar sources was one of the drawbacks of some of the existing genre-annotated corpora which we mentioned in Section 3.1. In the construction of LWGC-B corpus, we therefore tried to compile web pages from a diverse range of sources. We calculated source-diversity statistics for each genre in Table 3.9. We can see that our focused search avoided collecting too many web pages from single sites with most genre categories having a median of one web page collected per site. This is positive as it avoids associating genres with specific websites and layouts which are subject to fast change (although, of course, genres also change over time). However, note that there are still some web sites that might be over-represented such as the maximum of 23 pages from a single shopping web site.

However, even different sources could be on the same topic so we conducted an additional investigation into the topic diversity of LWGC-B corpus by extract-

Genre	Number of		Number of pages from the same website		
	web pages	websites	max	min	med
php	304	288	9	1	1
com	264	264	1	1	1
edu	299	299	1	1	1
blog	244	215	9	1	1
shop	292	209	23	1	1
instruction	231	142	15	1	1
recipe	332	116	8	1	1
news	330	127	12	1	1
editorial	310	69	11	1	3
forum	280	106	11	1	1
bio	242	190	15	1	1
faq	201	140	8	1	1
review	266	179	15	1	1
story	184	24	38	1	7
interview	185	154	11	1	1

Table 3.9: Statistics for individual categories which illustrate source diversity of LWGC-B corpus. Max, min and med are abbreviations of minimum, maximum and median respectively.

ing and comparing keywords of web pages in each genre category. The underlying assumption of this approach is that if web pages in a genre category of a corpus have topically similar keywords, that corpus is not topically diverse. For keyword extraction, we used log-likelihood statistic [42]. We want to extract words of a web page which have a significantly higher frequency compared to their frequency in the whole corpus. The keyword extraction procedure [121] for each web page consists of the following steps:

1. We produced a word frequency list for each web page as well as the whole corpus.

2. For each word in the word frequency list for each web page, we calculated the log-likelihood statistic by constructing the contingency table shown in Table 3.10 where *a* and *b* are the frequency of the word in the web page and the whole corpus respectively; *c* corresponds to the number of the words in the web page and *d* is the number of the words in the whole corpus. We can compute the log-likelihood value based on this formula:

	The web page	Whole Corpus
Freq of word	a	b
Freq of other words	c-a	d-b

Table 3.10: Contingency table for calculating log-likelihood. *a* and *b* are the frequency of the word in the web page and the whole corpus respectively. *c* corresponds to the number of words in the web page and *d* is the number of words in the whole corpus.

$$LL = 2((a\log(\frac{a}{E1})) + (b\log(\frac{b}{E2}))) \tag{3.8}$$

where $E1 = c\frac{(a+b)}{(c+d)}$ and $E2 = d\frac{(a+b)}{(c+d)}$.

3. Then we sort the word frequency list of each web page according to their LL values. The word with the biggest LL value in a web page has the most significant relative frequency difference between that web page and the whole corpus or to put it differently, the words with high LL value in a web page are its keywords.

We only considered keywords which are significant at the level of $p < 0.0001$ and also we removed some common words such as pronouns and determiners. Next, we needed to generalise from individual web pages to genre classes and counted the number of web pages in each genre class that a keyword appears in. Table 3.11 shows the keywords which appear in the highest number of web pages in each genre category of LWGC-B corpus. Each number shows the number of documents that the corresponding word has been selected as a keyword for.

A qualitative analysis of the results presented in Table 3.11 shows that there are very few topic-specific words in the table and that the majority of the words are genre-specific. For example, frequently asked questions are not distinguished by keywords that indicate FAQs on a specific topic but instead by general question words (such as *how* or *what*) and parts of the genre name itself. An exception is the keyword *program* which might indicate several FAQs on programming languages. Similarly, blogs and forums are not distinguished by specific topics but by, for example, posting dates for blogs, and forum-specific words such as *member, join, thread.* An exception is the genre category *recipe* where an unavoidable correlation to the topic *food* holds but even there our corpus did not contain only recipes of a specific type say, such as mostly vegetable recipes, as indicated by prevalence of flexible widely used ingredients with the possible exception of *chicken*. Some potentially topic-dependent keywords such as *cars, autos* for editorials turned out not to be because of the corpus containing many editorials about cars but because of frequent advertising links in the boiler plate. In addition, it is also important to note that some topic-like keywords probably mirror the actual state of web genres currently, for example the fact that many personal home pages are of scientists. Although to a certain degree subjective, we indicated potentially spurious "topic invasion" in our corpus with italics in the Table.

In order to compare our corpus with the existing web genre corpora in terms of topic diversity, we also extracted keywords from comparable genre classes in these corpora. Table 3.12 depicts some of the results. The qualitative analysis of the results show that the faq category in SANTINIS [131] is the least topically diverse category. Almost all the web pages in this genre class are about *hurricane* and *tax*. Also, results in Table 3.12 show that although keywords from categories such as blog and *forum* are mainly genre-specific, personal home pages in KI-04 [104] and SANTINIS seem to be mainly about Artificial Intelligence researchers and mathematicians respectively.

3.5 Conclusion

The interest in the web and its genres [103] resulted in a proliferation of genre-annotated web corpora, each of which was built according to its specific principles, using its own classification scheme and annotation guidelines. Problematically, these also tend to be unreliably annotated or the annotation reliability was just not tested. Therefore, [122] already call for a reliably annotated web genre corpus, but do not present a follow-up actual corpus. This chapter takes steps to remedy this research gap. Moreover, as noted in Section 2.5, other shortcomings of these corpora include provision of few pages for many genre classes as well as the occasional lack in source and topic diversity and appropriate storage formats.

In this chapter, we tackle these problems by using crowd-sourcing for genre annotation, leading to the Leeds Web Genre Corpus Balanced-design (LWGC-B) which is the first reliably annotated web genre corpus. We suggest that crowd-sourcing is the appropriate method to remedy the most important problem of the lack of reliable annotation, arguing that this allows speedy, accurate and inexpensive genre annotation that detaches the annotation proper from the potential bias

faq (201)		blog (244)		com (264)		editorial (310)		edu (299)	
58	can	70	posted	54	company	93	opinion	130	school
51	questions	50	january	48	services	69	news	124	students
46	information	47	comments	33	products	59	editorial	99	university
45	do	46	blog	25	service	50	blogs	71	campus
44	are	43	was	23	ltd	44	state	69	research
33	does	34	december	20	systems	41	columns	66	student
32	how	31	labels	20	corporation	39	autos	43	programs
29	frequently	31	day	19	clients	38	editorials	43	college
28	services	28	but	18	solutions	33	obituaries	43	academic
26	is	27	august	16	website	31	local	40	undergraduate
26	if	25	share	16	management	31	business	40	events
25	may	25	had	16	contact	29	columnists	37	faculty
24	what	24	july	16	construction	29	city	35	international
23	was	24	christmas	15	design	29	cars	35	graduate
22	will	23	twitter	15	business	29	ads	35	alumni
22	program	23	just	13	provide	28	jobs	33	admissions
22	available	23	april	13	group	27	reprints	32	high
21	top	23	am	13	corporate	27	government	31	learning
21	site	23	about	12	support	26	*obama*	31	information
20	page	22	october	12	industry	26	editor	31	education
bio(242)		forum (280)		instruction (231)		interview (185)		news (330)	
62	biography	201	posts	102	how	81	do	135	news
39	became	164	forum	46	step	77	was	104	said
27	had	143	join	43	or	74	did	30	police
26	*music*	137	thread	43	if	57	what	30	latest
25	later	135	date	41	do	41	think	29	photos
24	will	105	location	37	will	38	were	28	headlines
24	father	102	member	35	make	35	they	26	tuesday
24	as	93	pm	29	use	35	me	26	government
23	have	92	reply	28	was	34	people	25	sport
23	died	82	quote	26	can	34	had	25	minister
21	published	68	am	25	tips	32	because	25	health
21	born	67	post	24	are	30	really	25	blogs
20	married	66	profile	23	get	29	interview	24	sports
20	life	60	view	21	yourself	28	music	23	watch
20	film	60	forums	20	paper	28	lot	23	sun
20	during	57	re	20	job	27	like	23	national
20	award	53	thanks	19	water	26	there	23	former
20	*album*	46	replies	19	need	26	know	22	search
19	children	43	hi	19	instructions	24	would	22	*president*
18	were	38	linkback	19	be	23	just	22	lifestyle
php (304)		recipe (332)		review (266)		shop (292)		story (184)	
38	*research*	145	recipes	126	review	89	price	72	said
30	university	139	recipe	91	reviews	79	accessories	37	then
23	website	90	cup	39	product	65	shop	34	could
18	cv	75	sauce	38	very	65	shipping	33	old
16	site	69	cooking	37	rating	64	product	33	little
16	guestbook	67	garlic	35	recommend	61	free	31	shall
14	welcome	64	butter	34	service	57	more	31	came
14	page	63	pepper	33	comment	54	items	30	eyes
12	economics	62	sugar	31	overall	49	*amazon*	29	door
12	blog	60	the	27	helpful	47	reviews	28	words
11	teaching	60	ingredients	27	*book*	47	*clothing*	28	*king*
11	publications	60	cheese	26	great	46	delivery	27	thought
11	*professor*	58	add	25	excellent	46	buy	25	stood
11	pdf	56	teaspoon	25	but	45	customer	25	went
10	social	56	cook	24	*video*	43	gift	25	replied
10	engineering	55	onion	24	useful	42	products	25	man
9	web	55	*chicken*	24	reviewer	41	see	25	looked
9	projects	54	minutes	24	good	41	basket	24	cried
9	personal	53	chopped	23	out	40	store	24	woman

Table 3.11: Keywords from genre categories in LWGC-B

SANTINIS [131]							
faq (200)		php (200)		blog (200)		shop (200)	
110	*hurricane*	26	*math*	40	but	32	click
109	*noaa*	16	page	39	march	29	*dvd*
107	center	16	*mathematics*	30	just	28	price
84	aoml	15	university	29	posted	26	more
65	*tropical*	13	unl	28	had	25	basket
57	*tax*	12	*lincoln*	28	comments	22	uk
48	publication	12	guestbook	28	blog	22	info
47	faq	12	dk	27	like	21	delivery
46	form	11	research	25	am	21	add
42	pdf	11	*mathematical*	24	get	17	order
41	references	10	teaching	22	february	17	here
40	*cyclones*	10	*bradley*	20	know	17	details
37	*income*	9	theory	20	got	16	save
36	topic	9	office	20	going	15	summer
33	file	9	*nebraska*	19	really	15	offers
32	back	9	homepage	18	trackback	14	*games*
31	return	8	thesis	18	think	14	free
31	if	8	edu	18	there	14	*flowers*
26	*storm*	8	department	18	as	14	catalogue
25	*weather*	7	mit	17	very	13	product
KI-04 [104]							
php (126)		help (139)		shop (167)		forum (127)	
19	*intelligence*	52	do	42	store	41	post
18	computer	47	how	30	price	41	pm
17	research	40	what	23	cart	39	forum
15	*artificial*	37	can	22	shop	39	am
11	proceedings	36	faq	19	*books*	37	posts
11	conference	33	if	17	shipping	37	message
10	systems	26	there	17	gifts	30	reply
10	science	24	search	16	gift	24	thread
10	reasoning	24	com	16	buy	23	topic
10	homepage	23	be	15	products	23	forums
9	*professor*	23	web	15	click	22	posted
8	computational	22	site	14	more	20	view
7	member	22	help	14	*book*	20	quote
7	engineering	22	file	13	*music*	20	new
7	dr	21	will	12	sellers	19	re
7	*ai*	20	why	12	here	18	to
7	*aaai*	20	this	11	valentine	18	send
6	simulation	20	server	11	top	18	profile
6	publications	18	use	11	sale	17	last
6	language	18	http	11	now	17	edit
MGC (Multi-labelled Genre Collection) [159]							
blog (77)		forum (82)		faq (70)		fiction (67)	
30	posted	29	posts	34	can	36	had
22	pm	20	reply	31	if	35	said
20	blog	20	message	28	was	24	back
18	comments	19	quote	28	what	23	up
15	am	18	thread	24	how	22	looked
13	blogs	18	pm	24	do	19	eyes
11	but	17	am	24	faq	18	down
10	weblog	16	profile	23	are	18	could
10	people	16	post	19	http	18	would
10	comment	15	send	17	version	17	then
9	trackback	12	private	17	use	17	out
9	like	12	posted	17	q	17	into
9	here	11	view	16	html	16	door
9	april	11	topic	16	file	16	but
8	your	11	offline	15	be	15	which
8	think	11	list	15	using	15	room
8	site	11	forum	15	user	15	man
8	october	11	buddy	14	does	15	just
8	march	10	mode	14	com	15	head
7	will	10	may	14	web	14	felt

Table 3.12: keywords from some of the genre categories of the existing web genre corpora

of the expert team who developed the guidelines [124].

We developed precise and consistent annotation guidelines which consist of well-defined and well-recognized categories. The result of inter-coder agreement shows that the corpus has been annotated reliably. We also investigated the diversity of the topics per genre category by extracting the keywords. The qualitative analysis of the keywords confirms that our corpus is topically diverse.

Chapter 4

Supervised Models for Genre Classification

4.1 Introduction

This chapter investigates the performance of various features in supervised genre classification. Researchers conducted genre classification experiments with different features on different corpora with different sets of genre labels. As a result, it is difficult to compare them. Although Sharoff et al. [139] examined a wide range of features on most of the existing genre-annotated corpora, they concluded that the results cannot be trusted because of two main reasons. First, the spurious correlation between topic and genre classes in some of these corpora was one of the reasons for some of the very impressive results reported in Sharoff et al. [139]. These good results were achieved by detecting topics and not genres of individual texts. Second, some of these results are not trustworthy because of the low inter-coder agreement of some of these collections. As explained in the previous chapter, inter-coder agreement is a measure of annotation reliability and any results based on unreliable data could be misleading. Therefore, the question which set of features produces the best result in automatic genre classification on the web is still an open question.

The reliability as well as topic and source diversity of the corpus we constructed in the previous chapter allow us to surmount the shortcomings of existing web genre corpora and to investigate the answer to this open question. Therefore, in this chapter, we compare the performance of a wide range of features in automatic genre classification on the web by reimplementing some of the previous approaches to genre classification by researchers in this field. We also explore the discriminative power of new features which have not been used before in genre classification. Moreover, we examine the effect of feature combination as well as feature selection on the accuracy of the genre classifiers. We also for the first time in genre classification, compare the performance of AGI systems on both the original text and the main text of the web pages. The next section explains different steps of the experimental setup.

4.2 Experimental Setup

For all the experiments we use the Leeds Web Genre Corpus Balanced-design (LWGC-B) [6] via 10-fold cross-validation on the web pages. As explained in the previous chapter, the topic of web pages, the writing style of the authors as well as the design of the websites are three factors that can influence genre classification. In order to minimize the effect of these factors we collected web pages from various web sites. To reduce the impact of these three factors on genre classification even further, we ensured that all the web pages from the same websites are in the same fold. Many, if not all of the previous studies in automatic genre classification on the web ignored this essential step when dividing the data into folds.

We prepared two versions of the corpus: the original text and the main text corpora. First, we converted web pages to plain text by stripping HTML markup using the KrdWrd tool. [1] This resulted in the original text corpus which contains individual web pages with all the textual elements present on them. We also prepared the main text corpus which contains only the main text of the individual web pages, because web pages usually consist of different sections such as headers, footers, lists of links and advertisements which do not necessary belong together. Therefore, for the first time in AGI, we compare the performance of a wide range of classifiers on the original and the main text of the web pages. We removed the

[1] https://krdwrd.org/

boilerplate parts (e.g. headers, footers, template materials, navigation menus, lists of links and advertisements) and extracted the main text of each web page using justext [2] a heuristic based boilerplate removal tool [115]. Justext algorithm is the state-of-the-art method for removing boilerplate parts of web pages [115]. The HTML version of the web page is fed to the justext tool and its output is the main text of the web page in plain text format. Figure 4.1 shows the boilerplate part and the main text part of an example web page.

Since the outputs of the justext tool for 518 of our web pages were empty files, the main text corpus has fewer pages. However, the main text corpus still has a balanced distribution with a relatively large number of web pages per category. Table 4.1 compares the number of web pages in the two versions of the corpus. One interesting observation that can be made from this table is that the majority of empty pages produced by the justext tool belong to home page categories. The reason is that many home pages are in the form of a list of links and incomplete sentences.

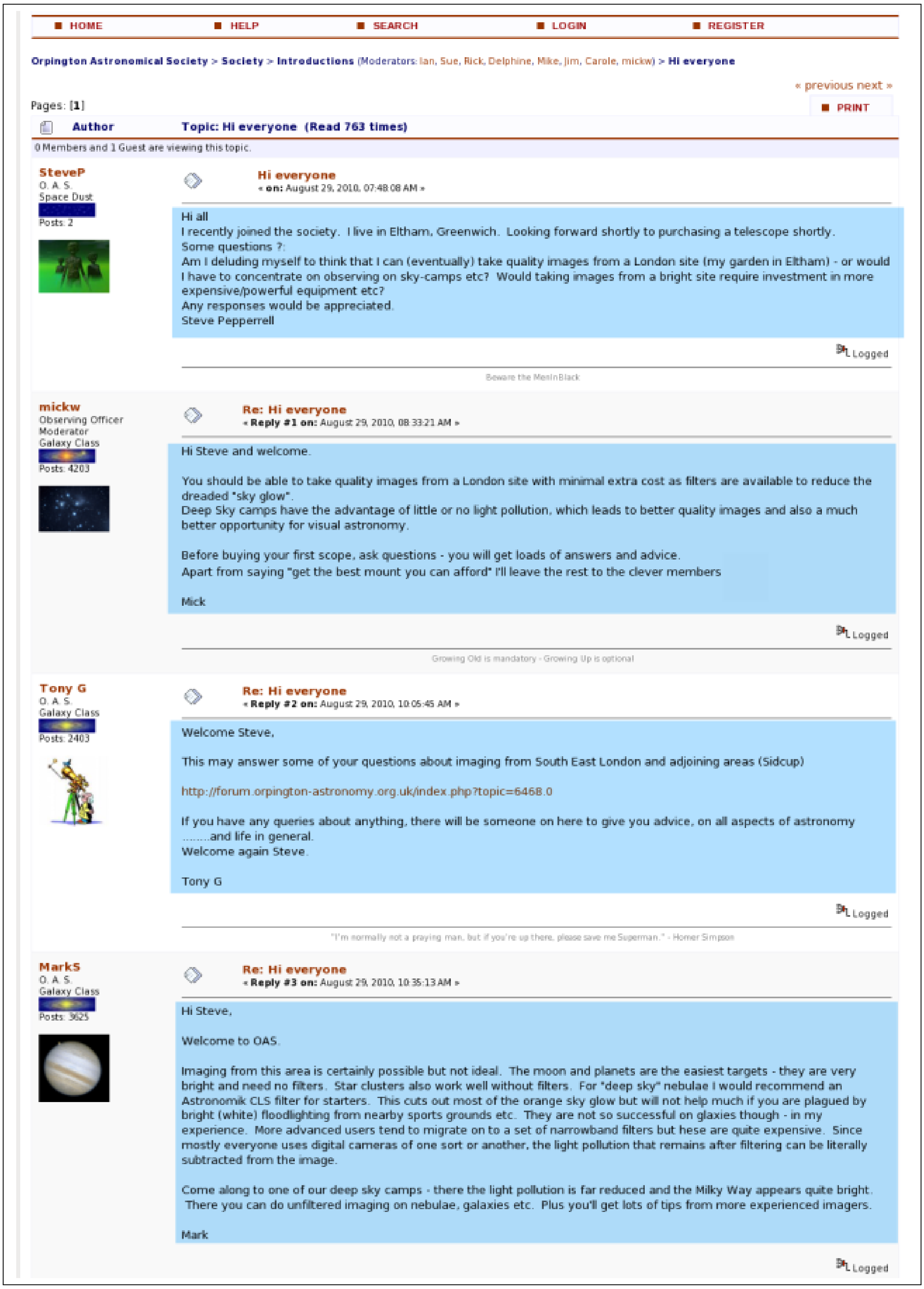
In the experiments conducted in this chapter, we employed two different feature representation techniques: normalized frequency where the frequency of each feature is normalised by the length of the web page; and binary, where the values "one" and "zero" represent the presence and the absence of each feature respectively.

For machine learning we chose Support Vector Machines (SVM) because it has been shown that it can cope well with high dimensional feature space [61]. It has also been shown by other researchers in AGI (e.g. [131] [111]) that SVM produces better or at least similar results compared to other machine learning algorithms. We used one-versus-one multi-class SVM implemented in Weka [57] with the default setting. All the experiments are carried out on both the original text and the main text corpora.

4.3 Evaluation Measures

In this thesis, we compare the performance of different classifiers by employing several standard evaluation measures used in NLP. These measures are computed

[2] http://code.google.com/p/justext/



Orpington Astronomical Society > Society > Introductions (Moderators: Ian, Sue, Rick, Delphine, Mike, Jim, Carole, mickw) > **Hi everyone**

Pages: [**1**]

Author **Topic: Hi everyone (Read 763 times)**

0 Members and 1 Guest are viewing this topic.

SteveP
O. A. S.
Space Dust
Posts: 2

Hi everyone
« **on:** August 29, 2010, 07:48:08 AM »

Hi all
I recently joined the society. I live in Eltham, Greenwich. Looking forward shortly to purchasing a telescope shortly.
Some questions ?:
Am I deluding myself to think that I can (eventually) take quality images from a London site (my garden in Eltham) - or would I have to concentrate on observing on sky-camps etc? Would taking images from a bright site require investment in more expensive/powerful equipment etc?
Any responses would be appreciated.
Steve Pepperrell

Logged

Beware the MenInBlack

mickw
Observing Officer
Moderator
Galaxy Class
Posts: 4203

Re: Hi everyone
« **Reply #1 on:** August 29, 2010, 08:33:21 AM »

Hi Steve and welcome.

You should be able to take quality images from a London site with minimal extra cost as filters are available to reduce the dreaded "sky glow".
Deep Sky camps have the advantage of little or no light pollution, which leads to better quality images and also a much better opportunity for visual astronomy.

Before buying your first scope, ask questions - you will get loads of answers and advice.
Apart from saying "get the best mount you can afford" I'll leave the rest to the clever members

Mick

Logged

Growing Old is mandatory - Growing Up is optional

Tony G
O. A. S.
Galaxy Class
Posts: 2403

Re: Hi everyone
« **Reply #2 on:** August 29, 2010, 10:05:45 AM »

Welcome Steve,

This may answer some of your questions about imaging from South East London and adjoining areas (Sidcup)

http://forum.orpington-astronomy.org.uk/index.php?topic=6468.0

If you have any queries about anything, there will be someone on here to give you advice, on all aspects of astronomy ....and life in general.
Welcome again Steve.

Tony G

Logged

"I'm normally not a praying man, but if you're up there, please save me Superman." - Homer Simpson

MarkS
O. A. S.
Galaxy Class
Posts: 3625

Re: Hi everyone
« **Reply #3 on:** August 29, 2010, 10:35:13 AM »

Hi Steve,

Welcome to OAS.

Imaging from this area is certainly possible but not ideal. The moon and planets are the easiest targets - they are very bright and need no filters. Star clusters also work well without filters. For "deep sky" nebulae I would recommend an Astronomik CLS filter for starters. This cuts out most of the orange sky glow but will not help much if you are plagued by bright (white) floodlighting from nearby sports grounds etc. They are not so successful on glaxies though - in my experience. More advanced users tend to migrate on to a set of narrowband filters but hese are quite expensive. Since mostly everyone uses digital cameras of one sort or another, the light pollution that remains after filtering can be literally subtracted from the image.

Come along to one of our deep sky camps - there the light pollution is far reduced and the Milky Way appears quite bright. There you can do unfiltered imaging on nebulae, galaxies etc. Plus you'll get lots of tips from more experienced imagers.

Mark

Logged

Figure 4.1: Extracting the main text of each web page by removing the boilerplate parts. Blue overlays indicate main text parts of the web page. URL: http://forum.orpington-astronomy.org.uk/index.php?topic=6466.0

	Number of web pages in corpora	
Genre	original text	main text
php	304	221
com	264	190
edu	299	191
blog	244	242
shop	292	221
instruction	231	229
recipe	332	243
news	330	320
editorial	310	307
forum	280	251
bio	242	242
faq	201	160
review	266	262
story	184	184
interview	185	183

Table 4.1: Number of web pages in individual genre classes in both original text and main text corpora.

by comparing the output of the classifiers against the gold standard annotations created in Chapter 3. The first evaluation measure that we employ is accuracy which can be computed by the following formula:

$$Accuracy = \frac{\text{number of correctly classified web pages}}{\text{total number of web pages}} \tag{4.1}$$

While accuracy is a perfect measure to see what fraction of the data set was classified correctly, it is not very informative on its own and other evaluation measures such as recall, precision and F-measure should also be computed to assess the performance of the classifiers for the individual categories. Therefore, we also employ precision, recall and F-measure as evaluation metrics. These measures can be computed as follows:

$$Precision = \frac{TP}{TP+FP} \tag{4.2}$$

$$Recall = \frac{TP}{TP+FN} \tag{4.3}$$

$$F-measure = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{4.4}$$

where TP, FP and FN are the numbers of true positives, false positives and false negatives for the target class respectively. Precision shows what proportion of instances classified as positive are actually positive in the gold standard. On the other hand, recall indicates what proportion of instances in the target class are correctly predicted by the classifier to be positive. F-measure is the harmonic mean of these two measures which provides an overall overview of the performance of the classifier in terms of precision and recall.

It also must be noted that we use the McNemar test [101] at the significance level of 5% to test for significant differences between the performance of the algorithms throughout this thesis. The use of the McNemar test for comparing classification learning algorithms was strongly recommended by Dietterich [38].

4.4 Lexical Features

In this section, we explore various lexical features for automatic genre classification. For all the experiments in this section we converted web pages to plain text using the KrdWrd tool. Then, we tokenized each document using the Stanford tokenizer [3] (included as part of the Stanford part of speech tagger [157]) and converted all the tokens to lower case.

Genre names and genre keywords. The first set of features which comes to mind in genre classification is genre names (e.g. news, editorial, interview, FAQs). The question is how accurately this set of features can classify genre categories. In order to answer this question, in the first experiment we used a list of genre names and genre keywords (presented in Appendix B.1) as features. The difference between genre name and keywords is that a genre name could consist of several genre keywords. For example, while the genre name *"frequently asked questions"* is a single feature in the genre name feature set, it is treated as three different features in genre keywords feature set (i.e. *"frequently"*, *"asked"* and *"questions"*). The motivation behind this experiment is based on the intuition that it is likely that the name of the genre or genre keywords are mentioned in the web pages which instantiate that genre. For instance, we expect to see the genre name "interview" in web pages which represent this genre. Although other studies in AGI (e.g. [131]) have used these features as part of genre-specific words, this is the first time that the performance of genre names are being explored on their own.

[3] http://nlp.stanford.edu/software/tagger.shtml

Common words. As noted in Section 2.6.2.1, Stamatatos et al. [146] were the first to use the 50 most common words (listed in Appendix B.2) in the British National Corpus (BNC) as features to distinguish four genre categories namely: editorials, letters to the editor, reportage, and spot news from the Wall Street Journal corpus. They reported an impressive accuracy of more than 90%.

Since our collection also contains editorial and news classes, it would be interesting to see how well this set of features can distinguish these two classes as well as other genre classes in our collection. Therefore, in order to investigate the discriminative power of these features, we reimplemented this approach on our corpus.

Function words. Another feature set frequently used in AGI is function words as explained in Section 2.6.2.2. However, there is no standard set of function words. Researchers used different sets of function words in their experiments but limited or no information was provided about the way that they were selected. For instance, in genre classification, Argamon et al. [2] used a list of 500 function words. However, they did not provide the list nor the method that they used to select these words. Researchers in authorship classification also used different sets of function words. While Zhao and Zobel [166] used a set of 365 function words and Koppel and Schler [80] proposed a list of 480 function words, Argamon et al. [3] reported a set of 675 words.

The list of function words that we used are extracted from the British National corpus (BNC). Appendix B.3 describes the function words extraction procedure from the BNC. It is interesting to note that the set of the 50 most common words in the BNC except the word *said* is a subset of the function words. This observation confirms the first property of the function words (i.e. they are more frequent than the content words) which is noted in Section 2.6.2.2.

Word unigrams. In the word unigrams model, all the individual words from the web pages in the training set are used as features. The features in a word unigrams model are combinations of function words and content words. In genre classification, word unigrams were used by Freund et al [50] and Sharoff et al. [139]. One disadvantage of this model is that it disregards word order.

To overcome this problem and to take advantage of word order, word *n*-grams (sequence of *n* consecutive words) have been proposed as textual features. Sharoff et al. [139] compared a wide range of features including word unigrams, bigrams and trigrams in genre classification. However, the classification accuracies achieved by word bigrams and trigrams were worse than the ones achieved by individual word features. In fact, in their experiments the word unigrams binary representation was one of the best performing feature sets. One possible reason for the lack of success of word *n*-grams features could be feature vector sparsity. Since most of the word *n*-grams are not encountered in a given text, the feature vectors are very sparse and therefore it is difficult for the classification algorithm to handle them effectively.

Character *n*-grams. Character *n*-grams were first used by Kanaris and Stamatatos [64] in genre classification. In their study, they used character *n*-grams with variable length and showed that this approach outperformed the best existing results for the Santini (96.2% accuracy versus 90.6%) and the KI04 (82.8% accuracy versus 74.8%) corpora. Sharoff et al. [139] improved these results by using fixed length character *n*-grams. They reported that a binary version of character tetra-grams (e.i. four consecutive characters) yields 97.14% on the Santini corpus and 85.81% on the KI04 corpus. Therefore, we re-implemented this approach and built a model based on character tetra-grams binary representation of the text. In this approach, we collected up to the 1,000 most frequent character tetra-grams of each text in the training set and combined them together to form the feature set. Since the number of tetra-grams are usually bigger than the number of words in a text, the character tetra-grams feature vector has higher dimensions than the word unigrams feature space.

The character tetra-grams model can capture some morphosyntactic properties of the texts such as ending of verbs (e.g. -ed, -ing) , ending of adjectives (e.g. -est) and adverbs (e.g. -ly). This model can also learn from different constituents of words such as prefixes and suffixes. Moreover, many of very common words in English such as pronouns (e.g. *they, she*), modal verbs (e.g. *may, can*) and coordinating conjunctions (e.g.*and, but, so*) have four or fewer characters.

The next section discusses and compares the empirical performance of these lexical features.

Features	original text	main text
Baseline	8.38	9.29
Genre Names binary	57.39	29.02
Genre Name normalized frequency	38.29	14.16
Genre keywords binary	64.17	37.05
Genre keywords normalized frequency	52.92	19.70
Common words binary	39.00	37.69
Common words normalized frequency	63.09	59.20
Function words binary	65.71	55.57
Function words normalized frequency	74.95	66.86
Word unigrams binary	**89.32**	76.61
Word unigrams normalized frequency	85.21	74.91
Character 4-grams binary	87.96	**78.88**
Character 4-grams normalized frequency	84.13	76.20

Table 4.2: Classification accuracy of different lexical features in genre classification on the LWGC-B. The best results are presented in bold.

4.4.1 Results and Discussion

Table 4.2 shows the result of the lexical features listed in the previous section on both the original text and the main text corpora. At first glance, we see that all the features outperformed the baselines (the baselines are the accuracies of the majority classifiers that categorize every document as the most frequent class). We can also observe that the results of genre classification on the original text corpus are higher than on the main text corpus. This shows that boiler plates contain valuable information which helps genre classification. The results also show that binary representation of genre names performs better than its normalized frequency representation. This means that the existence of genre names is more informative than their frequency. However, it only yields 57.39% for the original text corpus and 29.02% for the main text corpus which shows that removing boilerplates resulted in removing many of the genre names.

Surprisingly, the model based on the normalized frequency of the fifty most common words in English yields 63.09% accuracy on the original corpus and

59.20% (less than a 4% drop) on the main text corpus which shows the discrimination power of such a small set of features in genre classification. However, the binary representation of common words yields lower accuracy compared to their normalized frequency representation which shows that the number of occurrence of these features is more informative than simply the existence or non-existence of these features. This is because these common words usually appear in many different genres but with different frequency. These results show that there is a correlation between these genre categories and the fifty most common words. The keywords for different genre classes presented in Table 3.11 depict some examples of this correlation. For instance, the keywords *how, if* and *can* are common in the genre class instruction, while the word *said* is a keyword in the genre classes news and story.

In order to identify the easiest and the most difficult categories for the classifier based on normalized frequency of the fifty most common words, we computed the F-measures for individual genre classes obtained by this model on both the original text and the main text corpora (Figure 4.2). It is interesting to see that the F-measure for four genre categories: story, biography, news and editorial are higher than 0.7 on the original text. The same picture can also be observed for these categories in the main text corpus with the exception of the editorial class which experiences a slight drop in F-measure. These results confirm the results reported by Stamatatos et al. [146] which showed that this model can differentiate news from editorial with high accuracy. However, Figure 4.2 illustrates that this model struggles to distinguish genre categories such as company homepages, on-line shops and reviews. Therefore, we can conclude that the discriminative power of this set of features varies across different genre classes.

Moreover, the results (Table 4.2) show that the binary representation of the word unigrams is the best performing feature set when we use the whole text of the web pages. However, on the main text corpus, character tetra-grams outperform other features. Tables 4.3 and 4.4 depict confusion matrices for word unigrams binary representation for the original corpus and the main text corpus, respectively. The left most column shows the true genre labels of the web pages while the top row of the table illustrates the genre labels assigned by the classifier. We can observe in Table 4.3 that the two most confused categories are news and editorial. However, Table 4.4 shows a higher degree of confusion between

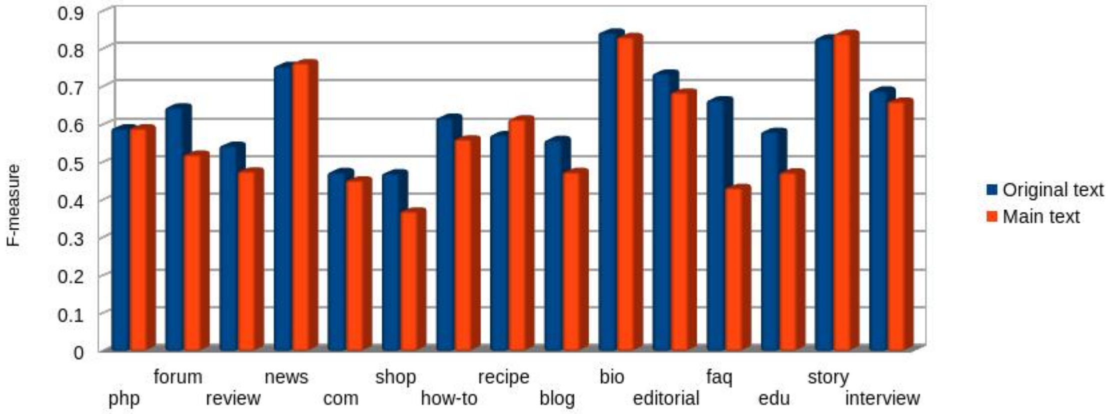


Figure 4.2: Comparing the F-measure for individual genre classes, using only the 50 most frequent words in BNC as features (normalized frequency representation).

	php	forum	review	news	com	shop	howto	recipe	blog	bio	editorial	faq	edu	story	interview
php	285	0	1	1	3	0	3	0	2	5	0	0	2	0	2
forum	1	264	1	4	0	1	1	0	4	0	0	1	0	0	3
review	10	0	229	4	2	2	2	0	9	1	0	0	0	1	6
news	0	0	0	294	4	0	0	0	1	2	20	5	1	0	3
com	8	0	2	0	243	3	1	1	0	0	0	3	3	0	0
shop	1	0	27	0	10	247	4	0	0	0	0	2	0	0	1
howto	3	4	3	0	2	0	202	2	3	0	0	7	1	0	4
recipe	0	0	0	0	0	0	2	330	0	0	0	0	0	0	0
blog	8	0	2	1	1	0	2	0	214	1	0	0	0	8	7
bio	11	0	0	3	0	1	0	0	1	214	2	0	2	0	8
editorial	0	0	1	67	1	0	0	0	1	0	235	2	0	0	3
faq	3	2	0	1	3	2	8	1	0	1	1	179	0	0	0
edu	7	0	0	1	2	1	2	0	0	0	0	0	286	0	0
story	6	0	0	0	0	0	2	0	9	2	0	0	0	160	5
interview	6	0	3	5	0	0	2	0	8	1	0	1	0	0	159

Table 4.3: Confusion matrix for genre classification of original web pages with word unigrams binary representation.

genre categories. It means that boiler plates contain many discriminative features (e.g. dates and times in forums) and removing them reduces the accuracy of all the classifiers. However, comparing the two confusion matrices in Tables 4.3 and 4.4 shows that removing the boiler-plates of web pages resulted in a decrease of confusion between the categories news and editorial. In Table 4.3, 67 editorial web pages were classified wrongly as news, whereas this was reduced to 42 in Table 4.4. The reason for this improvement is that news and editorial web pages often have similar design and consequently similar boilerplates which could confuse the genre classifiers.

In order to identify the easiest and the most difficult categories for the classifiers, we also computed recall, precision and F-measure for each individual category. Tables 4.5 and 4.6 illustrate these standard classification evaluation metrics for word unigrams binary representation of the original text and the main text corpora, respectively. While the category recipe can be identified by the classifier very easily with an F-measure equal to 0.99, other categories such as interview, news and editorial seem to be more challenging for the classifier in the original text corpus. However, a different picture emerges from the main text corpus. Table 4.6 shows that by removing the boiler plates, the F-measures for all the categories fall. Nevertheless, this decrease in F-measure is much more significant for some genre classes such as faqs, forum and blog. The reason could be that the classifier mainly relied on features in boilerplates such as genre names to identify these categories.

It has to be noted that the list of lexical features used in this chapter is not complete. Researchers in the field of automatic genre classification have used various techniques to select lexical features. For instance, as noted in Section 2.6.2.3, Kim and Ross [72] presented a classification model based on a genre-specific word list. They construct this list by compiling all the words within a genre class and then counting the number of files in which each word is found. In the next step, genre-specific word list is built by taking the union of all the words found in at least 75% of the files in each genre. Eissen and Stein [148] proposed another technique for automatic extraction of genre-specific words which is more than simple count statistics. This approach was described in details in Section 2.6.2.3.

We did not investigate the performance of such techniques which require thresholding because we can use feature selection methods to obtain the optimum fea-

	php	forum	review	news	com	shop	howto	recipe	blog	bio	editorial	faq	edu	story	interview
php	172	9	1	3	6	0	2	0	9	7	0	2	9	0	1
forum	13	187	8	5	3	4	4	0	15	0	3	1	5	0	3
review	1	27	198	4	1	2	7	1	7	0	8	0	1	0	5
news	3	8	1	249	4	1	0	0	2	6	29	6	4	0	7
com	8	4	2	0	142	16	2	0	0	1	1	7	7	0	0
shop	5	2	23	1	23	127	11	1	2	2	0	12	9	0	3
how-to	1	14	2	0	1	4	185	4	5	0	3	7	0	0	3
recipe	2	7	2	0	4	4	4	215	2	0	0	3	0	0	0
blog	8	24	5	3	1	2	9	0	148	2	5	0	1	17	17
bio	7	0	0	5	1	1	0	0	2	212	1	0	3	1	9
editorial	0	9	1	42	0	5	0	0	8	2	231	1	3	0	5
faq	2	8	0	3	7	13	16	2	1	1	5	96	5	0	1
edu	5	1	0	2	4	4	1	0	0	1	2	1	170	0	0
story	0	3	1	6	0	0	0	0	14	1	0	0	0	158	1
interview	2	7	2	2	0	0	1	0	12	1	5	0	0	1	150

Table 4.4: Confusion matrix for genre classification of main text of web pages with word unigrams binary representation.

ture set in the word unigrams model. Many feature selection methods are available such as information gain, χ^2 and mutual information. In a recent study, Sarkar et al. [134] compared these feature selection techniques and concluded that the use of information gain has the most positive impact over other feature selection methods. Therefore, we employed information gain algorithm to select the informative features in the word unigrams model and also to reduce the dimension of the feature space. Information gain reduced the number of features in the word unigrams model dramatically (from 67,208 words to 7,353 words in the original text corpus[4]) which is very beneficial with respect to memory usage and computational cost. It also improved the genre classification accuracy from 89.32% to 89.56% on the original text corpus. However, this improvement is not statically significant (see Section 4.7 for more comprehensive experimental results for feature selection).

4.5 POS-based Features

Argamon et al. [2] and Santini [131] used part of speech trigrams (sequences of three consecutive part of speech (POS) tags) to capture the syntactic properties of the text. Examples of syntactic features could be passive voice (e.g. *he was born*), exclamation (e.g. *oh my god*) and question (e.g. *what is that*) sentences. Frequency of parts of speech (e.g. verb, noun, adjective) are also used in genre classification as structural cues by many researchers such as Karlgren and Cut-

[4] These numbers are the average number of features extracted from the training data in the cross validation after removing singletons.

class	Recall	Precision	F-measure
php	0.937	0.816	0.872
forum	0.942	0.977	0.96
review	0.860	0.851	0.856
news	0.890	0.771	0.827
com	0.920	0.896	0.908
shop	0.845	0.961	0.899
how-to	0.874	0.874	0.874
recipe	0.993	0.988	0.990
blog	0.877	0.849	0.862
bio	0.884	0.942	0.912
editorial	0.758	0.910	0.827
faq	0.890	0.895	0.892
edu	0.956	0.969	0.962
story	0.869	0.946	0.906
interview	0.859	0.791	0.823

Table 4.5: Precision, Recall and F-measure for genre classification of original web pages with word unigrams binary representation

class	Recall	Precision	F-measure
php	0.778	0.751	0.764
forum	0.745	0.603	0.666
review	0.755	0.804	0.779
news	0.778	0.766	0.772
com	0.747	0.720	0.733
shop	0.574	0.693	0.628
how-to	0.807	0.764	0.785
recipe	0.884	0.964	0.922
blog	0.611	0.651	0.631
bio	0.876	0.898	0.887
editorial	0.752	0.788	0.77
faq	0.6	0.705	0.648
edu	0.890	0.783	0.833
story	0.858	0.892	0.875
interview	0.819	0.731	0.773

Table 4.6: Precision, Recall and F-measure for genre classification of main text of web pages with word unigrams binary representation

ting [67] and Feldman et al. [46]. In order to explore the discriminative power of structural features, we reimplemented these approaches using our corpus. Therefore, the goal of the set of experiments presented in this section is to investigate whether parts of speech and part of speech *n*-grams can be employed as genre-revealing features.

For POS tagging, we used the Stanford maximum entropy tagger [157] which annotates text with 36 POS tags. Since Santini [131] showed that the set of POS trigrams without punctuation tends to perform better than the set with punctuation, we removed the punctuation marks. Table 4.7 shows a sample text and its POS tags conversion.

4.5.1 Results and Discussion

The results of structural features (see Table 4.8) show that binary representation of POS trigrams outperforms other structural features on both versions of the corpus. Overall, accuracy results reported in Table 4.8 show that the classification accuracies obtained by POS bigrams and unigrams are lower than the accuracy yielded by POS trigrams. The results also highlight that the performance of structural features (i.e. POS and POS *n*-grams) are much less accurate than that of lexical features such as word unigrams and character *n*-grams reported in the previous section. The reason is that the actual words of a web page have more discriminative power than their POS tags. For example two phrases *it was established* from

text	its POS tags representation
I recently beaded a necklace for my daughter for her birthday. It was really playful and I even put a magnetic clasp on it so she could easily take it on and off. I made it using the tubular peyote stitch. Unfortunately, it didn't last long. Just the other day I found her in her car seat with the necklace in her mouth and beads everywhere. Oh well, we'll try again in a few years.	PRP RB VBD DT NN IN PRP$ NN IN PRP$ NN . PRP VBD RB JJ CC PRP RB VBD DT JJ NN IN PRP IN PRP MD RB VB PRP IN CC IN . PRP VBD PRP VBG DT JJ NN NN . RB , PRP VBD RB JJ JJ . RB DT JJ NN PRP VBD PRP$ IN PRP$ NN NN IN DT NN IN PRP$ NN CC NNS RB . UH UH , PRP MD VB RB IN DT JJ NNS . PRP RB VBD JJ IN PRP VBD RB .

Table 4.7: A sample text from a personal blog and its POS tags conversion. URL: http://3sistersbeads.blogspot.com/

Structural Features	original text	main text
Baseline	8.38	9.29
POS 3-grams binary	73.18	61.23
POS 3-grams normalized frequency	70.28	57.83
POS 2-grams binary	64.10	54.91
POS 2-grams normalized frequency	68.94	60.76
POS normalized frequency	60.14	54.64

Table 4.8: Classification accuracy of POS-based features in genre classification

a company home page and *he was born* from a biography are treated the same as a POS trigrams features (i.e. PRP-VBD-VBN) whereas they are different features in the word unigrams representation of the text.

4.6 Text Statistics Features

Other researchers (e.g. [48, 70]) discovered the benefit of document statistics such as average sentence length, average word length and type/token ratio in automatic genre classification. These studies motivated us to explore the discriminative power of text statistics features in genre classification. Therefore, the question that we seek to answer in this section is whether text statistics features can help us to improve automatic genre classification. We divided the text statistics features that we explore in genre classification into four groups: token features, named entity tags, readability features and HTML tags. Table 4.9 summarizes these features (please see Appendix C for the complete lists of HTML features). In the following sections, we will provide technical details about the experiments and feature extraction.

Text statistics category	Features
Token features	number of tokens number of types frequency of punctuation marks frequency of currency characters
Named entity tags	time, location, organization, person, money, date
Readability features	average sentence length standard deviation of sentence length average word length standard deviation of word length average number of syllables per word type/token ratio average parse tree height average number of noun phrases per sentence average number of verb phrases per sentence entity coherence
HTML tags	sections / style formatting visual features such as forms, images, lists,tables programming

Table 4.9: List of text statistics features

4.6.1 Token Features

The first group of text statistics features is token features which includes number of tokens, types, punctuation marks and currency characters. A significant advantage of such features is that they can be easily extracted from the text with no additional requirements except the availability of a tokenizer. Moreover, we must note that frequency of punctuation marks and currency characters were normalized by the document length.

4.6.2 Named Entity Tags

Named-entity recognition provides some semantic information about words in a text by categorizing them into predefined categories such as the names of person, organization, location, date, time and money. The idea of using named entity tags as features in genre classification comes naturally and it is based on the assumption that the likelihood of appearances of these named entity categories in different genres vary. For instance, we expect to see organization names in company homepages and list of prices in on-line shops. To test this hypothesis, we used normalized frequency of named entity tags of time, location, organization, person, money and date as features. Previously, named entity tags are used in genre classification by Vidulin et al. [159]. However, they used only three named entity tags namely: date, location and person.

For extracting named entity tags, we used the Stanford Named Entity Recognizer [47]. Each text from original and main text corpora was run through this tool, generating an output file with the relevant named entity tags attached to each word. Table 4.10 shows the output of the Named Entity Recognizer for a company homepage. The tokens and the named entity tags are separated by "/". The tag "O" indicates that the token does not belong to any of the predefined named entity tags.

4.6.3 Readability Features

Klare [75] defines readability as "the ease of understanding or comprehension due to the style of writing.". The readability of documents varies across different genres. This is the assumption and rationale behind the features proposed for automatic genre classification in this section. This assumption is based on the findings reported by Kate et al. [69] where it was shown that using features that are indicative of the genre classes significantly improve the readability prediction accuracy. This result indicates that there is a correlation between genre classes and the level of readability. Therefore, the aim of this section is to exploit this correlation in order to improve automatic genre classification accuracy.

One criterion for determining the level of readability of a text is sentence complexity. In general, documents with complex sentences are harder to comprehend. Different authors may construct sentences in different genres with different de-

Abbott/ORGANIZATION and/O Livery/ORGANIZATION Service/ORGANIZATION Inc./ORGANIZATION ,/O your/O complete/O transportation/O service/O located/O in/O the/O heart/O of/O the/O Berkshire/LOCATION Mountains/LOCATION ,/O in/O western/O Massachusetts/LOCATION ./O

Family-owned/O and/O operated/O ,/O Abbott/ORGANIZATION 's/O quality/O service/O and/O solid/O reputation/O for/O punctuality/O ,/O regardless/O of/O time/O or/O weather/O conditions/O ,/O has/O made/O our/O company/O grow/O steadily/O since/O its/O inception/O in/O 1965/DATE ./O

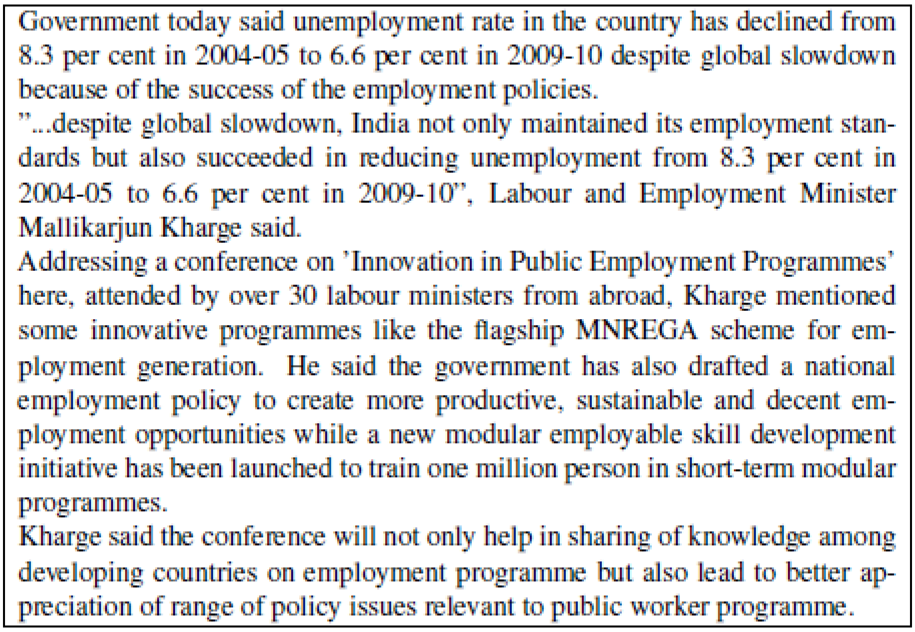
Don/PERSON Abbott/PERSON founded/O Abbott/ORGANIZATION and/O Livery/ORGANIZATION Service/ORGANIZATION in/O 1965/DATE ,/O now/O seven/O members/O of/O the/O Abbott/PERSON family/O are/O actively/O engaged/O in/O the/O business/O year/O round/O ,/O 24/O hours/O a/O day/O ,/O 7/O days/O a/O week/O ./O

Table 4.10: Output of Named Entity Recognizer for a company homepage. URL: http://abbottslimo.com/

grees of complexity. To demonstrate this point, we give an example which supports this hypothesis. Figures 4.3 and 4.4 show part of a news article and an interview respectively. By comparing the sentences in these two examples, we observe that the news article is more complex in terms of the sentence structure than the interview. Therefore, measuring the sentence complexity of texts could be beneficial in automatically differentiating the genre classes. One simple approach to measure the sentence complexity is to compute the average sentence length as well as standard deviation of the sentence length. Another more sophisticated technique to measure the magnitude of sentence complexity is by computing the average number of verb phrases and noun phrases in its sentences [110], as including more verb phrases and noun phrases in a sentence increases the sentence complexity [114].

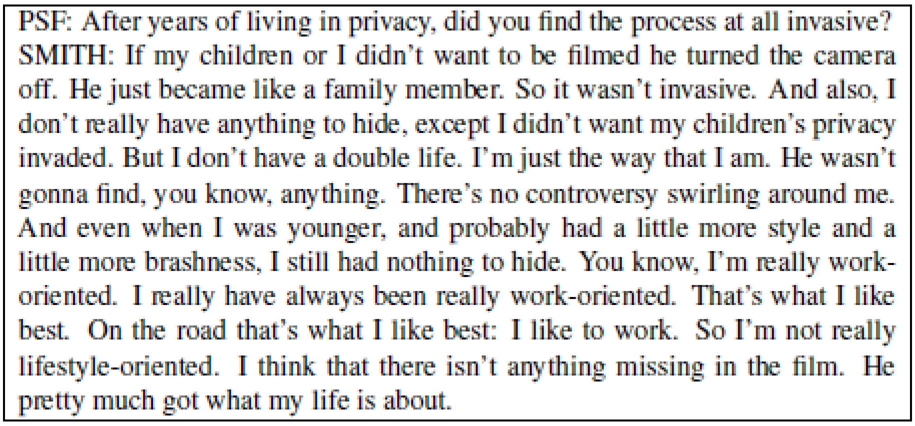
In addition to these features, features such as average word length, standard deviation of word length, average number of syllables per word and type/token ratio are usually used to determine the level of readability of a document. One advantage of these features is that their extraction is computationally cheap and does not required any specific tool. The only resource that we used a part from a tokenizer, to extract these features was the CELEX English Lexical Database [5]

[5] http://www.lel.ed.ac.uk/facilities/howto/celex/



Government today said unemployment rate in the country has declined from 8.3 per cent in 2004-05 to 6.6 per cent in 2009-10 despite global slowdown because of the success of the employment policies.
"...despite global slowdown, India not only maintained its employment standards but also succeeded in reducing unemployment from 8.3 per cent in 2004-05 to 6.6 per cent in 2009-10", Labour and Employment Minister Mallikarjun Kharge said.
Addressing a conference on 'Innovation in Public Employment Programmes' here, attended by over 30 labour ministers from abroad, Kharge mentioned some innovative programmes like the flagship MNREGA scheme for employment generation. He said the government has also drafted a national employment policy to create more productive, sustainable and decent employment opportunities while a new modular employable skill development initiative has been launched to train one million person in short-term modular programmes.
Kharge said the conference will not only help in sharing of knowledge among developing countries on employment programme but also lead to better appreciation of range of policy issues relevant to public worker programme.

Figure 4.3: Part of a news article URL: http://articles.economictimes.indiatimes.com/2012-03-01/news /31113736_1_unemployment-rate-labour-global-slowdown



PSF: After years of living in privacy, did you find the process at all invasive?
SMITH: If my children or I didn't want to be filmed he turned the camera off. He just became like a family member. So it wasn't invasive. And also, I don't really have anything to hide, except I didn't want my children's privacy invaded. But I don't have a double life. I'm just the way that I am. He wasn't gonna find, you know, anything. There's no controversy swirling around me. And even when I was younger, and probably had a little more style and a little more brashness, I still had nothing to hide. You know, I'm really work-oriented. I really have always been really work-oriented. That's what I like best. On the road that's what I like best: I like to work. So I'm not really lifestyle-oriented. I think that there isn't anything missing in the film. He pretty much got what my life is about.

Figure 4.4: Part of an interview URL: http://www.furious.com/perfect/pattismith.html

which we employed for computing the average number of syllables per word.

Another approach to measure the scale of sentence complexity is to consider the depth of the syntax trees in the documents. A deep parse tree is an indication of a complex sentence. Figure 4.5 and 4.6 illustrate the syntax trees of two sentences from a news article and a blog respectively. Although these two sentences have the same length, the sentence in Figure 4.5 has a much deeper parse tree which indicates its complexity. Therefore, we used the average depth of syntax trees as a novel feature in genre classification.

Another criterion for measuring the readability of a text is its coherence. Barzilay and Lapata [10] introduced a method for coherence assessment of a document that is based on its entity grid representation. In the first step of this approach, entities in each sentence of the document are detected as the subject (S), object (O) or other (X). In the second step, a grid which shows different entity transitions is constructed. For example, an (S O) transition occurs when an entity is the subject in one sentence and an object in the following sentence; (O -) transition happens when an entity appears in object position in one sentence and is not present in the next. In the next step, the probability of each transition type is computed. The entity coherence features which are 16 in total, are the probability of each of these pairs of transitions.

For example, Table 4.12 shows the entity grid for the text in Table 4.11. In this grid, the entity JOURNALIST in the first sentence with the grammatical role object (O) is a subject (S) in the second sentence. Therefore, an (O S) transition occurs for this entity. As far as we are aware, entity coherence features have not been used in genre classification and it would be interesting to know if they can help us in distinguishing genre categories.

For extracting the readability features mentioned above, we used the Stanford Parser [76]. However, web pages must be cleaned before they can be fed to a parser, because parsers cannot handle tables and lists of links. For instance, parsing the text version of the web page which is shown partially in Figure 4.7 due to its length, took 69 minutes and 40 seconds, whereas, parsing this page after cleaning and extracting the main text using the justext tool [115] took only 41 seconds on the same machine. The reason is that the table or the list outlined by

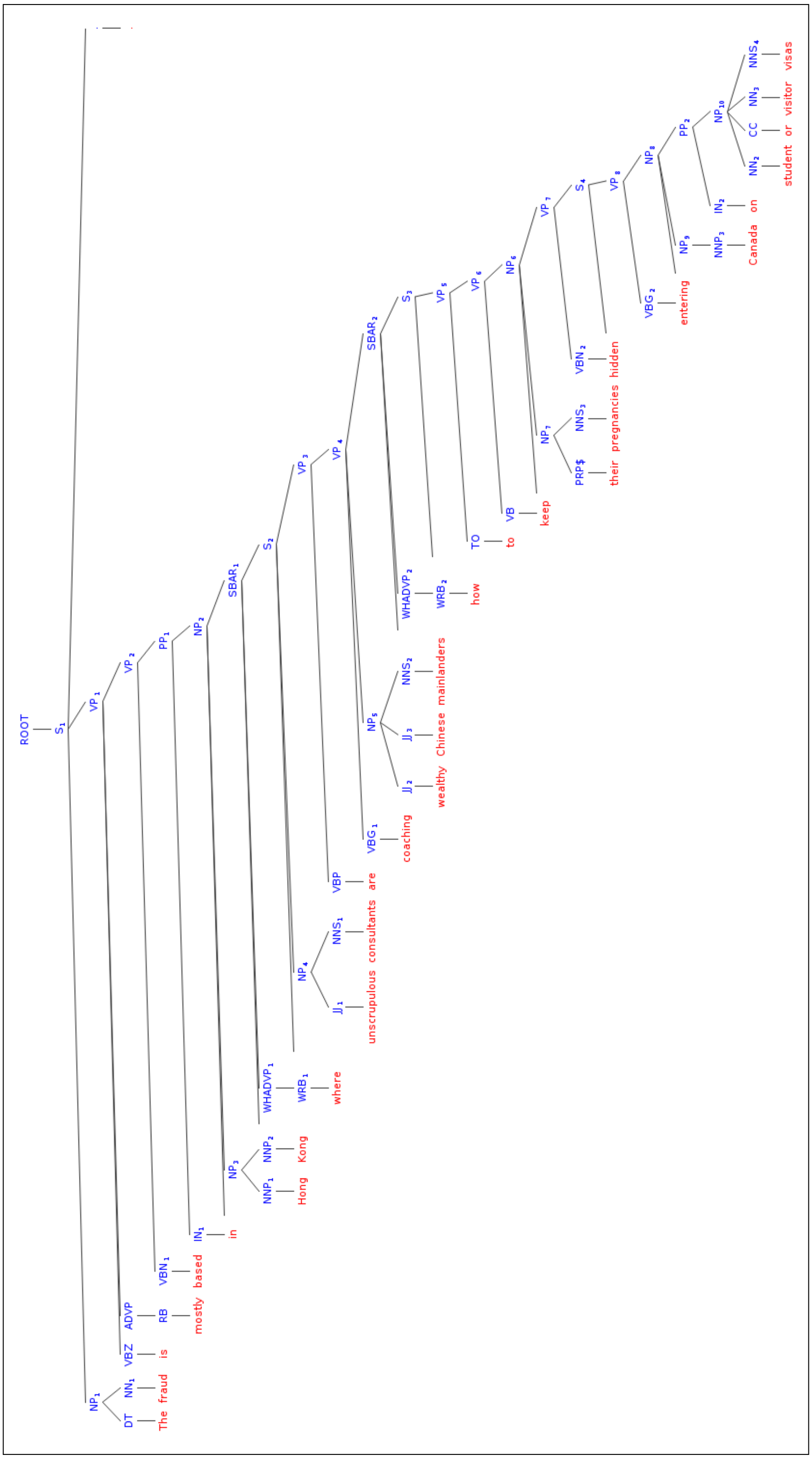


Figure 4.5: Parse tree of a sentence from a news article URL: http://cnews.canoe.ca/CNEWS /Politics/2012/02/21/19406306.html

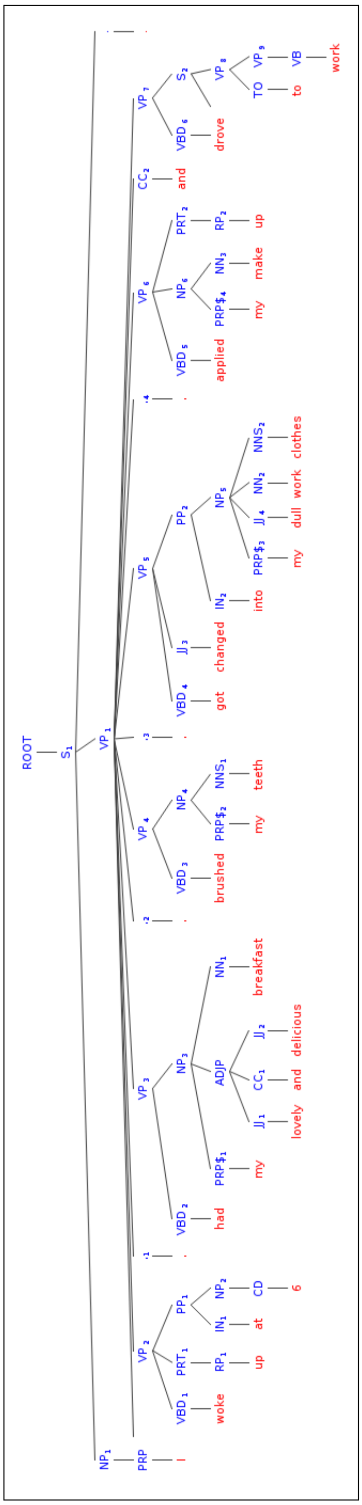


Figure 4.6: Parse tree of a sentence from a personal blog URL: http://ninatoday.com /2008/october/a5.html

1	Unidentified gunmen have on Monday shot and killed a broadcast journalist in the fragile city of Galka'yo of Mudug region in central Somalia, witnesses reported.
2	The dead journalist was identified by the witnesses as Ali Ahmed Abdi, one of the Somali journalists working for a website and an independent local radio station in Garsor Village, northern part Galka'yo town and controlled by Puntland administration of Somalia.
3	Mr. Abdi, who was shot in the head and chest, died at the scene, the witnesses said.
4	The identities and motives of the gunmen who shot dead the journalist were not clear.

Table 4.11: Part of a news web page URL: http://allafrica.com/stories/201203050594.html

	1	2	3	4
GUNMEN	S	-	-	X
MONDAY	X	-	-	-
JOURNALIST	O	S	-	X
CITY	X	-	-	-
REGION	O	-	-	-
SOMALIA	X	X	-	-
WITNESSES	S	X	S	-
ABDI	-	X	S	-
ONE	-	X	-	-
JOURNALISTS	-	X	-	-
WEBSITE	-	X	-	-
STATION	-	X	-	-
TOWN	-	X	-	-
ADMINISTRATION	-	X	-	-
CHEST	-	-	X	-
SCENE	-	-	X	-
MOTIVES	-	-	-	S

Table 4.12: Entity transition grid for the text in Table 4.11

	parse time	# sentences
original web page	69m40.640s	99
cleaned web page	0m41.969s	98

Table 4.13: Comparing the parse time for the original web page shown in Figure 4.7 and its cleaned version produced by the justext tool [115].

the red line in Figure 4.7 poses difficulties for the parser. This table is assumed by the parser to be one very long sentence. However, since it is not a well-formed sentence and it is very long (651 words), it takes the parser a considerable amount of time to process it. Table 4.13 compares the parsing time of the original web page shown in Figure 4.7 and its main text only. Therefore, we only used the main text of each web page as an input to the parser. For web pages for which the justext tool [115] produced empty files, we treated these features as missing values. Moreover, we used the Brown Coherence Toolkit [6] to construct the entity grid for each web page. This tool needs the parsed version of the text as an input. Therefore, for web pages for which the justext tool [115] produced empty files, we also treated these features as missing values.

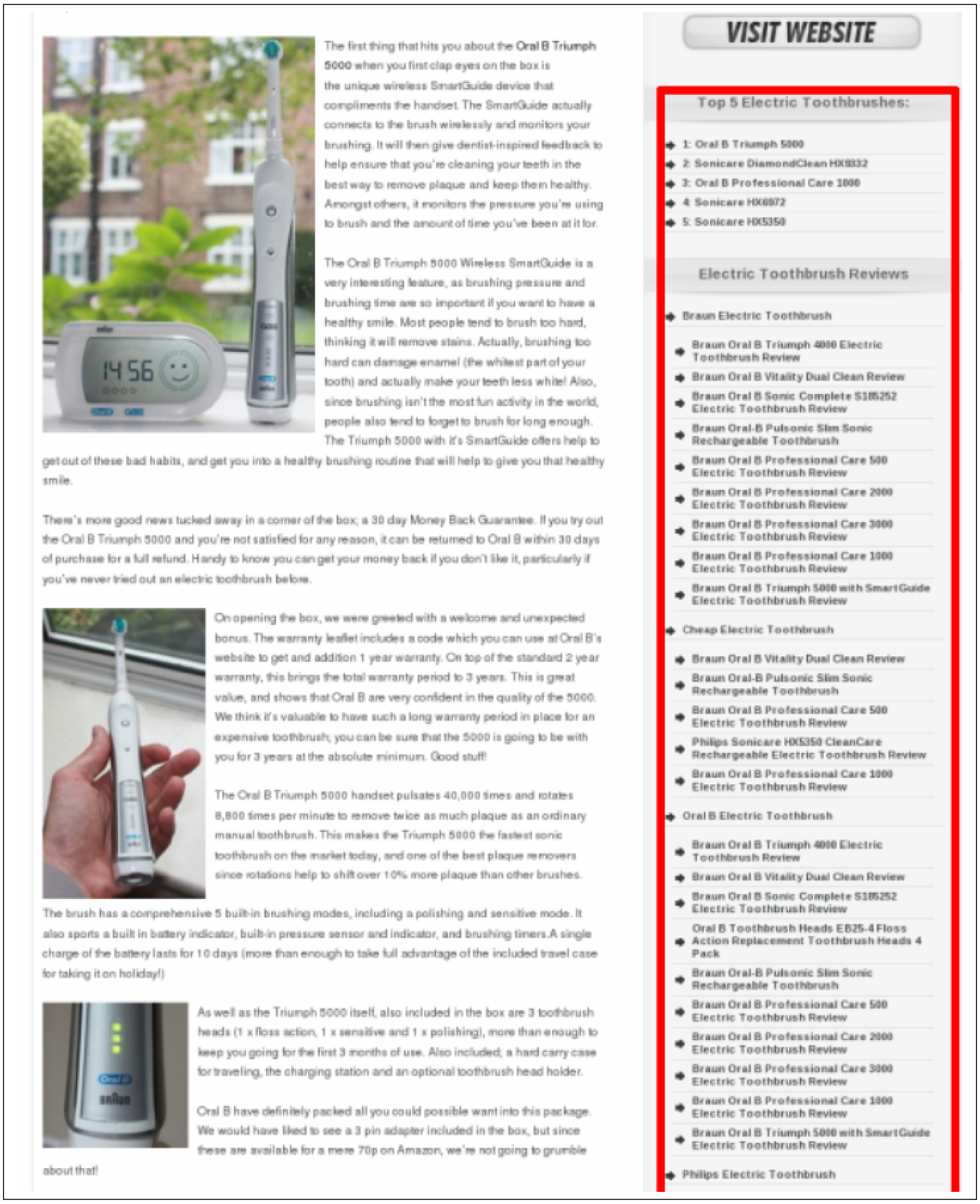
4.6.4 HTML Tags

Many researchers (e.g. [131], [104]) have used HTML tags as features in automatic genre classification. We also used normalized frequency of HTML tags as features. We grouped HTML tags into four categories: sections and style, formatting, visual features and programming. The detailed description of HTML tags that we used as features in automatic genre classification can be found in Appendix C.

4.6.5 Results and Discussion

The results of the text statistics features set (Table 4.14) show that these features yield 55.47% accuracy for the original text corpus and 59.17% for the main text corpus. That means that text statistics features are less accurate than structural and lexical features in automatic genre classification. However, not all these features contributed equally to the genre classification accuracy and it would be interest-

[6] http://www.cs.brown.edu/ melsner/manual.html



The first thing that hits you about the **Oral B Triumph 5000** when you first clap eyes on the box is the unique wireless SmartGuide device that compliments the handset. The SmartGuide actually connects to the brush wirelessly and monitors your brushing. It will then give dentist-inspired feedback to help ensure that you're cleaning your teeth in the best way to remove plaque and keep them healthy. Amongst others, it monitors the pressure you're using to brush and the amount of time you've been at it for.

The Oral B Triumph 5000 Wireless SmartGuide is a very interesting feature, as brushing pressure and brushing time are so important if you want to have a healthy smile. Most people tend to brush too hard, thinking it will remove stains. Actually, brushing too hard can damage enamel (the whitest part of your tooth) and actually make your teeth less white! Also, since brushing isn't the most fun activity in the world, people also tend to forget to brush for long enough. The Triumph 5000 with it's SmartGuide offers help to get out of these bad habits, and get you into a healthy brushing routine that will help to give you that healthy smile.

There's more good news tucked away in a corner of the box; a 30 day Money Back Guarantee. If you try out the Oral B Triumph 5000 and you're not satisfied for any reason, it can be returned to Oral B within 30 days of purchase for a full refund. Handy to know you can get your money back if you don't like it, particularly if you've never tried out an electric toothbrush before.

On opening the box, we were greeted with a welcome and unexpected bonus. The warranty leaflet includes a code which you can use at Oral B's website to get and addition 1 year warranty. On top of the standard 2 year warranty, this brings the total warranty period to 3 years. This is great value, and shows that Oral B are very confident in the quality of the 5000. We think it's valuable to have such a long warranty period in place for an expensive toothbrush; you can be sure that the 5000 is going to be with you for 3 years at the absolute minimum. Good stuff!

The Oral B Triumph 5000 handset pulsates 40,000 times and rotates 8,800 times per minute to remove twice as much plaque as an ordinary manual toothbrush. This makes the Triumph 5000 the fastest sonic toothbrush on the market today, and one of the best plaque removers since rotations help to shift over 10% more plaque than other brushes.

The brush has a comprehensive 5 built-in brushing modes, including a polishing and sensitive mode. It also sports a built in battery indicator, built-in pressure sensor and indicator, and brushing timers.A single charge of the battery lasts for 10 days (more than enough to take full advantage of the included travel case for taking it on holiday!)

As well as the Triumph 5000 itself, also included in the box are 3 toothbrush heads (1 x floss action, 1 x sensitive and 1 x polishing), more than enough to keep you going for the first 3 months of use. Also included; a hard carry case for traveling, the charging station and an optional toothbrush head holder.

Oral B have definitely packed all you could possible want into this package. We would have liked to see a 3 pin adapter included in the box, but since these are available for a mere 70p on Amazon, we're not going to grumble about that!

VISIT WEBSITE

Top 5 Electric Toothbrushes:

- 1: Oral B Triumph 5000
- 2: Sonicare DiamondClean HX9332
- 3: Oral B Professional Care 1000
- 4: Sonicare HX6972
- 5: Sonicare HX5350

Electric Toothbrush Reviews

- Braun Electric Toothbrush
  - Braun Oral B Triumph 4000 Electric Toothbrush Review
  - Braun Oral B Vitality Dual Clean Review
  - Braun Oral B Sonic Complete S185252 Electric Toothbrush Review
  - Braun Oral-B Pulsonic Slim Sonic Rechargeable Toothbrush
  - Braun Oral B Professional Care 500 Electric Toothbrush Review
  - Braun Oral B Professional Care 2000 Electric Toothbrush Review
  - Braun Oral B Professional Care 3000 Electric Toothbrush Review
  - Braun Oral B Professional Care 1000 Electric Toothbrush Review
  - Braun Oral B Triumph 5000 with SmartGuide Electric Toothbrush Review
- Cheap Electric Toothbrush
  - Braun Oral B Vitality Dual Clean Review
  - Braun Oral-B Pulsonic Slim Sonic Rechargeable Toothbrush
  - Braun Oral B Professional Care 500 Electric Toothbrush Review
  - Philips Sonicare HX5350 CleanCare Rechargeable Electric Toothbrush Review
  - Braun Oral B Professional Care 1000 Electric Toothbrush Review
- Oral B Electric Toothbrush
  - Braun Oral B Triumph 4000 Electric Toothbrush Review
  - Braun Oral B Vitality Dual Clean Review
  - Braun Oral B Sonic Complete S185252 Electric Toothbrush Review
  - Oral B Toothbrush Heads EB25-4 Floss Action Replacement Toothbrush Heads 4 Pack
  - Braun Oral-B Pulsonic Slim Sonic Rechargeable Toothbrush
  - Braun Oral B Professional Care 500 Electric Toothbrush Review
  - Braun Oral B Professional Care 2000 Electric Toothbrush Review
  - Braun Oral B Professional Care 3000 Electric Toothbrush Review
  - Braun Oral B Professional Care 1000 Electric Toothbrush Review
  - Braun Oral B Triumph 5000 with SmartGuide Electric Toothbrush Review
- Philips Electric Toothbrush

Figure 4.7: Tabular structures pose difficulty for parsers. URL: http://www.bestelectrictoothbrushreviews.co.uk/braun-oral-b-triumph-5000/

ing to find the best individual text statistics features that maximize classification accuracy. To achieve this aim, we used the information gain algorithm to select the best features from the training data.

Features	original text	main text
Text statistics	55.47	59.17

Table 4.14: Classification accuracy of text statistics features in genre classification

Category	Feature
Readability	average word length
Readability	average number of syllables per word
HTML sections / style	<title>
named entity tags	organization
Token features	number of words
HTML sections / style	<body>
Token features	number of types
Token features	question mark
named entity tags	person
HTML visual features	<a>
Readability	type/token ratio
named entity tags	date
named entity tags	location
Readability	standard deviation of word length
HTML visual features	<input>
HTML visual features	<form>
HTML sections / style	<div>
HTML visual features	<img>
HTML visual features	<li>
HTML sections / style	<h1>

Table 4.15: The top 20 text statistics features selected by information gain.

Table 4.15 shows the top 20 text statistics features sorted by their information gain. These features belong to various text statistics features. The first two top features are average word length and average number of syllables per word which are correlated. This list also shows that text statistics features such as number of tokens, number of types, semantic features such as named entity tags and type/token ratio are very discriminative in genre classification.

Another interesting observation that can be made from this table is that question mark is in the top ten text statistics features which shows that it plays an important role in discriminating text genres. For example, interviews and frequently asked questions web pages are usually characterized by high frequency of question marks.

4.7 Feature Combination and Selection

In the previous sections, we described several different feature sets that we used in automatic genre classification. Each set of features provides us with different kinds of information about the content, the structure and the style of genre classes. In this section, we combine these features in order to put these pieces of information together. Therefore, one aim of this section is to investigate whether we could improve genre classification performance by combining the vectors obtained by different document representations in the previous sections into a single vector simply by concatenating them. Another aim of this section is to examine whether applying feature selection could increase genre classification accuracy. In order to provide answers to these two questions, we tried various feature combinations with and without feature selection. To be more precise, we combined three sets of features. The first set of features is the word unigrams binary representation which was the best performing feature in the lexical category. The second set of features is the combination of the best performing POS-based features and the third set of features is the text statistics features.

As in Section 4.4.1, we employed information gain as the feature selection method. For this experiment we used the corpus via 10-fold cross-validation on the web pages. For each fold, the feature selection was applied on the training set and the model constructed based on the selected features was applied to the test set. Table 4.16 shows the results of various feature combinations with and without feature selection on both original and main text corpora. The results presented in Table 4.16 show that the combination of word unigrams binary features and text statistics features resulted in improving genre classification accuracy for both original and main text corpora. Although this improvement is statistically significant for the main text corpus, there is no significant difference between these two models for the original corpus.

Surprisingly, adding POS-based features to the word unigrams features decreased the genre classification accuracy in both original and main text corpora. When we combine all three feature sets, we also experience decline in accuracy compared to the accuracy yielded by the word unigrams features alone. The reason could be that the models are over-fitted on the training data and as a result, they perform poorly on the test data. Over-fitting which is a common problem in machine learning, is the use of models that include more features than are necessary to represent the data or the use of more sophisticated approaches than are needed [58]. Therefore, combining various features will not always improve the performance of the classification task. In other words, increasing the number of features does not always result in improvement of the classification accuracy.

Moreover, applying feature selection to the models presented in Table 4.16 resulted in improving genre classification performance in all the cases. However, these improvements are not statistically significant. The most important benefit of applying feature selection method as we can observe in Table 4.16 is that it dramatically decreases the dimensions of the feature vectors (average number of features extracted from the training data in the cross validation after removing singletons). As a result the computational cost as well as the memory usage drop sharply.

4.8 Conclusion

In this chapter, we conducted the first comprehensive feature comparison on a reliably annotated and topic diverse web genre corpus. We examined the performance of various lexical, POS-based and text statistics features in genre classifications. We carried out all the experiments on two versions of the corpus: the original text and the main text. The main text of the web pages were extracted by removing the boilerplate parts. This is the first time that the performance of genre classification models is compared on both the original and the main text of the web pages. The classification models tend to perform better on the original text corpus because boilerplate parts of the web pages often contain genre names. Another possible reason could be that the web pages within the same genre category use similar

Feature	Description
1	word unigrams binary
2	combination of POS normalized frequency, POS bigrams normalized frequency and POS trigrams binary
3	text statistics features

Feature	without feature selection				with feature selection			
	original text		main text		original text		main text	
	acc.	# features	acc.	# features	acc.	# features	acc.	# features
1	89.32	67,208	76.61	42,832	89.56	7,353	76.78	4,148
2	73.39	28,883	61.40	21,270	73.54	5,993	61.46	5,461
3	55.47	178	59.17	178	55.39	99	59.40	94
1&2	87.74	96,091	75.13	64,102	88.17	13,346	75.54	9609
1&3	**89.48**	67,386	**78.09** †	43,010	**89.68**	7,452	**78.18** †	4,242
2&3	73.76	29,061	61.72	21,448	73.92	6,092	61.84	5,555
1&2&3	87.99	96,269	75.45	64,280	88.35	13,445	75.59	9,703

Table 4.16: Genre classification accuracy using feature combination with and without feature selection. In this table acc. stands for accuracy. The symbol † indicates the results which are significantly better than the result yielded by word binary unigrams alone.

design or templates.

Moreover, the results show that the best performing models are word unigrams and character 4-grams binary representation of the web pages. However, the weakness of these models is that they can be easily influenced by the topic, the style of authors and the design of the web pages. Therefore, they must be used cautiously in genre classification. In order to minimize the influence of these factors on genre classification, the web pages were collected from various topics and sources. Also, in the experimental setup, we ensured that all the web pages from the same website are either in the training set or the test set. On the other hand, the advantage of these models is that they can be easily extracted from the text. In contrast to POS-based features and some of the text statistics features, they do not require any sophisticated natural language processing tools such as a parser or a part of speech tagger.

The results also show that the performance of the POS-based and the text statistics features is considerably lower than the performance of the lexical features such as word unigrams and character 4-grams. Moreover, one drawback of these features is that their extraction depends on automated tools such as part-of-speech taggers and parsers whose performance varies for different genres. Also, some of these features are computationally very expensive and may not be practical in order to be employed by the search engines.

Chapter 5

A Semi-supervised Graph-based Model for Genre Classification

5.1 Introduction

In the previous chapter, we investigated the performance of various lexical, POS-based and text statistics features in genre classification of web pages using a standard supervised learning framework. Most of the current works in the field of AGI are based on this approach i.e. extracting features from the content of the documents and classify them by employing a standard supervised algorithm. However, on the web there are other sources of information which can be utilized to improve genre classification of web pages. For instance, the web has a graph structure and web pages are connected via hyper-links. These connections and the link patterns between the web pages could be exploited to improve genre classification. The underlying assumption of this approach is that a page is more likely to be connected to pages with the same genre category. For example, if the neighbouring web pages of a particular web page are labelled as shop, it is more likely that this web page is a shop too, whereas, it is highly unlikely that it is a news or an editorial. This property (i.e. entities with similar labels are more likely to be connected) is known as homophily [136]. We hypothesise that homophily exists for genre classes and it can help us to improve genre classification on the web.

While genre categorization on the web is quite mature, the issue of utilizing hyper-links for genre classification has been relatively unexplored. Therefore, in this chapter, our aim is to improve genre classification by learning from neighbouring web pages. Various graph-based classification algorithms have been proposed to improve the classification accuracy in network data sets. These techniques take advantage of the link structure of the data and the homophily property. However, implementing these algorithms in a supervised manner using available genre-annotated data sets is impossible, because for any given web page, we also require genre annotations of its neighbouring web pages. Although labelled documents are costly and very time consuming to obtain, unlabelled web pages are easy to collect. Therefore, we can use semi-supervised graph-based methods which learn from both labelled and unlabelled data.

In this chapter, we first give a brief review of classification algorithms in network data and introduce related work which applies graph-based algorithms to text classification in general. Then we propose a semi-supervised graph-based algorithm namely min-cut which is a novel approach in genre classification (Section 5.2.5 and Section 5.3). The experimental results (see Sections 5.3.3.2 and 5.3.4.2) show the effectiveness of the proposed model, as it outperforms the supervised learning approaches examined in the previous chapter.

5.2 An Overview of Classification in Networked Data

5.2.1 Graph-based Classification Algorithms

There has been a growing interest in learning from networked data. By networked data, we mean data that can be best described by a graph where the nodes in the graph are the entities to be classified and the edges in the graph are the relations or the links between these entities. Examples of network data are web pages which are connected via hyper-links; bibliographic data which are connected through citation and social networks where people are connected by friendship links. Such connected data provides opportunities to develop better-performing machine learning algorithms, because in network data, we have access to nodes' neighbours. One of these opportunities offered by networked data is that it allows collective inference. It means that the classification of linked entities can be conducted simultaneously.

Various graph-based classification algorithms have been developed which take advantage of the rich structure of networked datasets. Although each one of these techniques has different characteristics, they mainly work based on the homophily assumption. Therefore, if in a linked dataset, it is true that connected entities tend to be in the same class, then graph-based methodologies may improve the classification accuracy. Nevertheless, some graph-based learning algorithm can go beyond the homophily property and learn more complicated relationships and patterns between the linked objects in the dataset. The next section introduces some of the graph-based classification algorithms which have been employed for text classification.

5.2.2 Related Work on Graph-based Algorithms for Text Classification

Various graph-based classification algorithms have been proposed to improve topic classification for web pages, such as the relaxation labelling algorithm [25], iterative classification algorithm [90], Markov logic networks [29] and weighted-vote relational neighbour algorithm [92]. These classification algorithms which utilize hyper-link connections between web pages to construct graphs, outperformed the classifiers which are solely based on textual content of the web pages for topic classification. Such connected data presents opportunities for boosting the performance of genre classification as well.

Graph-based web page classification presented in studies such as [29, 90, 92] on the WebKB dataset [30] could be considered as genre classification as opposed to topic classification. The WebKB dataset contains web pages from four computer science departments categorised into seven classes: student, faculty, staff, department, course, project and other. However, this dataset is very specific to the academic domain with low coverage for the web overall, whereas we examine graph-based learning for automatic genre classification of web pages on a much more general dataset with popular genre classes such as news, blog and editorial. Moreover, the graph-based algorithms used on the WebKB dataset are all supervised and were performed on a very clean and noise free dataset which was achieved by removing the class other. Class other contains all the web pages which do not belong to any other predefined classes. However, our experiment is in a semi-supervised manner which is a much more realistic scenario for the web, because it is highly unlikely that for each web page, we have genre labels for all its neighbouring web pages as well. Therefore, we perform our experiment on a very noisy dataset where neighbouring web pages could belong to none of our predefined genre classes. Sections 5.2.4 and 5.2.5 describe our semi-supervised graph-based classification experiment, where we use min-cut algorithm as a novel technique in genre classification.

Previous work on web page categorization utilized the information from the neighbouring web pages mainly in two ways: by merging the text of the neighbouring pages with the text of the page to classify or by employing collective graph-based techniques. The research by Chakrabarti et al. [25] is one of the earliest work which exploit link structure of the web to improve web page classification based on topics. They used two hyper-linked corpora: a sample from IBM's Patent Server database and a sample of 900 web pages from yahoo manually annotated into 13 topic classes such as art, health and business. In their first set of experiments, they naively treated the words in the linked documents as if they were appeared in the target web page. This approach resulted in decreasing the accuracy rate of their system compared to the baseline performance (i.e. using the text in the target document alone). In the second set of experiments, they combined naive Bayes local classifiers with relaxation labelling [125] for collective inference. This approach resulted in a significant error reduction, compared to the baseline performance. Chakrabarti et al. [25] concluded that incorporating words from neighbouring pages hurts the performance, whereas, using only the predicted category of the neighbours improves the classification performance. Similar results were reported by Oh et al. [108] who also attempted to exploit the link structure of the web in order to improve web page classification based on topic. They used a collection of encyclopaedia articles categorised into 76 topic classes.

In the same line of research, Macskassy and Provost [91, 92] proposed the weighted-vote relational neighbour (wvRN) algorithm for learning in connected data. The wvRN classifier performs relational classification in two steps. Firstly, it computes a weighted average of the estimated class membership probabilities of the node's neighbours. Secondly, it performs collective inference via a relaxation labelling method [125] similar to that used by Chakrabarti et al. [25]. They compared the performance of wvRN algorithm to the algorithm proposed by Chakrabarti et al. [25] on the Cora dataset [98] which contains machine learning papers categorised into seven topics and the WebKB dataset [30]. They reported that while wvRN yielded the best results on the Cora dataset, Chakrabarti et al. [25] algorithm outperformed wvRN on the WebKB dataset.

In the spirit of work by Chakrabarti et al. [25] and Macskassy and Provost [91, 92], Lu and Getoor [90] proposed iterative classification algorithm (ICA) for classification in linked data sets. ICA can learn a variety of different patterns among the categories of linked objects by modelling the link distributions around an object. To be more precise, this approach models the neighbourhood of an object by computing three different link feature vectors namely: mode, count and binary which are all built based on statistics computed from categories of the neighbouring pages. Lu and Getoor [90] evaluated this approach on three datasets: Cora [98], CiteSeer [53] which includes papers from six topic categories and WebKB [30]. In all three datasets ICA outperformed the content-only classifier. However, they did not compare ICA with other graph-based methods.

With the aim of comparing graph-based classification algorithms, Sen et al. [136] compared ICA, Gibbs sampling [60] with other graph-based learning techniques such as loopy belief propagation [155], and mean-field relaxation labelling in topic classification using two bibliographic data sets: Cora [98] and CiteSeer [53] described above. In both data sets, all four graph-based algorithms outperformed the content-only classifier. However, they reported that the differences between these graph-based techniques were not significant.

A different line of research on classification on network data focuses on representing the relational structure of the web in first-order logic. Craven et al. [31] used the WebKB corpus to investigate the performance of the First Order Inductive Learner (FOIL) [118]. The main idea of this approach is that, first, it provides some positive and negative examples of classes and a set of relations between these classes to represent the data. Then FOIL generates rules for these classes and based on these rules it classifies the test data. Slattery and Mitchell [143] extended this idea and proposed a new algorithm called FOIL-HUBS which is a combination of FOIL and Hubs and Authorities algorithm [77]. They reported that FOIL-HUBS performs better than using FOIL alone.

Further studies in the same research direction includes the study by Crane and McDowell [29]. They investigated the performance of Markov Logic Networks (MLNs), which are also based on first-order logic, in classification of linked datasets. They compared the performance of MLNs and ICA algorithms on Cora, CiteSeer and WebKB datasets. In all three datasets, MLNs methods outperformed the ICA approach.

All the graph-based text classification approaches listed in this section are based on topic except the studies which are conducted on WebKB dataset. The classes in this dataset could be counted as genres because they are categorised based on their function. However, as we noted at the beginning of this section, this dataset is very specific to the academic domain, whereas we examine graph-based learning for automatic genre classification of web pages on a much more general dataset. Moreover, all the studies on the WebKB dataset are in supervised manner and were performed on a very clean and noise free dataset, whereas, we implement this approach in a semi-supervised manner which is a much more realistic scenario on the web, because it is highly unlikely that for each web page, we have genre labels for all its neighbouring web pages as well. Therefore, first we give a brief introduction to semi-supervised graph-based learning in Section 5.2.3 and then we describe our semi-supervised graph-based classification experiment, where we use the min-cut algorithm as a novel technique in genre classification, in Sections 5.2.4 and 5.2.5.

5.2.3 Semi-Supervised Graph-based Learning

In many classification problems, collecting labelled data is expensive and time consuming, while unlabelled data is widely available and can be collected very cheaply. Therefore, with the aim of learning from both labelled and unlabelled data, semi-supervised learning methods are proposed (for a detailed literature survey see [167]). In graph-based semi-supervised learning, first, a weighted graph is constructed in which both the labelled and unlabelled data are represented as vertices. Therefore, we suppose that there are m labelled instances $(x_1.y_1),...,(x_m,y_m)$ where x and y represent an instance and its label respectively; and n unlabelled points $x_{m+1},...,x_{m+n}$. The objective of a semi-supervised graph-based algorithm is to learn from both labelled and unlabelled data by taking into account the links between the instances. Since we do not have the labels of the neighbouring web pages of the pages in our dataset, we adopt a semi-supervised graph-based technique to learn from both labelled and unlabelled data.

5.2.4 The Binary Min-cut Algorithm

The binary min-cut algorithm deals with two labels only. The main objective of the min-cut algorithm is to cut the graph into two disjoint partitions by removing a set of edges in a way that the sum of their weights is minimal. The min-cut algorithm has two different forms: the common minimum cut and the minimum s-t cut with the aim of finding the minimum cut of the graph such that the source (s) and sink (t) vertices are in two different partitions. In this thesis, by min-cut algorithm, we mean the minimum s-t cut where the source and the sink correspond to the two classification nodes.

The min-cut algorithm is based on the idea that linked entities are highly likely to belong to the same class. The first step of this algorithm is to construct the graph $G = (V,E)$. All the instances in the training and test dataset are represented as vertices (example vertices) in the graph and the links between them are represented as undirected weighted edges in the graph which indicate the similarity of the connected nodes. The graph G also has two classification vertices s and t. Example vertices are connected to classification vertices s and t by weighted edges which specify the likelihood of the example vertices to be classified as the classification vertices.

The second step of the min-cut algorithm is to find the minimum cut of the graph in order to split the graph into two disjoint subsets of all vertices, S and T with $s \in S$ and $t \in T$. In other words, the task is to find a cut of the graph which minimizes the following formula:

$$W(S,T) = \sum_{u \in S, v \in T} W(u,v) \tag{5.1}$$

where $w(u,v)$ denotes the weight of the edge between two vertices u and v. Fortunately, minimum cuts can be found in polynomial time by using the maximum flow algorithm [28, 79].

Blum and Chawla [20] propose a semi-supervised graph min-cut algorithm which incorporate both labelled and unlabelled data. If n is the number of the instances in the whole data set including both labelled and unlabelled data, and l is the number of the labelled data, we can formulate semi-supervised min-cut as an algorithm that solves the following optimization problem:

$$\min_{y\in\{0,1\}} \infty \sum_{i=1}^{l} (y_i - Y_{li})^2 + \sum_{i,j=1}^{n} w_{ij}(y_i - y_j) \tag{5.2}$$

where y is the predicted label; Y is the given label of the labelled data and w_{ij} is the weight of the edge between two vertices i and j. The first term of the function in Equation 5.2 is called the loss function and the second term is called the regularizer. Note that the loss function has infinite weight (∞). That means if we reverse the class of the labelled data, the value of the function in Equation 5.2 will be infinity. On the other hand, if $y_i = Y_{li}$ for labelled data, the value of loss function will be zero (We define that $\infty \cdot 0 = 0$). Since the classes of the labelled instances are known, the minimization of the function in (Equation 5.2), will be dependent on the minimization of the regularizer.

5.2.5 Multi-class Min-cut

Semi-supervised min-cut classification algorithm proposed by Blum and Chawla [20] is a binary classification algorithm, whereas, we have a multi-class problem. Unfortunately, multi-class min-cut is NP-hard and there is no exact solution for it. Nevertheless, Ganchev and Pereira [51] proposed a multi-class extension to Blum and Chawla's [20] min-cut algorithm by encoding a multi-class min-cut problem as an instance of metric labeling. Kleinberg and Tardos [78] introduced the metric labeling problem for the first time for classification problems where the data involves pairwise relationships among the objects to be classified. The main idea of metric labelling for web page classification can be described as follows:

Assume we have a weighted and undirected graph $G = (V, E)$ where each vertex $v \in V$ is a web page and the edges represent the hyper-links between the web pages. The task is to classify these web pages into a set of labels L which is a function $f : V \rightarrow L$. In order to do this labelling task in an optimal way, we need to minimize two different types of costs. First, there is a non-negative cost $c(v,l)$ for assigning label l to web page v. Second, if two web pages v_1 and v_2 are connected together with an edge e with weight w_e, we need to pay a cost of $w_e \cdot d(f(v_1), f(v_2))$ where $d(.,.)$ denotes distance between the two labels. A big distance value between labels indicates less similarity between them. Therefore, the total cost of labelling task f is:

$$E(f) = \sum_{v \in V} c(v, f(v)) + \sum_{e=(v_1,v_2) \in E} w_e \cdot d(f(v_1), f(v_2)) \tag{5.3}$$

Kleinberg and Tardos [78] developed an algorithm for minimizing $E(f)$. However, their algorithm uses linear programming which is impractical for large data [22]. In a separate study for metric labelling problems , Boykov,et al. [22] have developed a multi-way min-cut algorithm to minimize $E(f)$. This algorithm is very fast and can be applied to large-scale problems with good performance [22, 65].

5.3 Genre Classification via Semi-supervised Min-cut

5.3.1 Why might Semi-supervised Min-cut Work?

Semi-supervised min-cut classification algorithms proposed by Blum and Chawla [20] and Ganchev and Pereira [51] are based on the idea that linked entities are highly likely to belong to the same class. In other words, it is based on the homophily assumption. We also hypothesised that homophily exists for genre classes and can help us to improve genre classification on the web. Therefore, semi-supervised min-cut classification should be able to improve genre classification on the web if our hypothesis holds. Although, as noted in Section 5.2.2, there are other graph-based techniques which we could employ in genre classification, we chose the min-cut algorithm because it is only based on homophily assumption whereas other graph-based techniques such as ICA and relaxation labelling could learn from other links patterns.

Conventional genre classification algorithms consider each web page in isolation. But when it is difficult to accurately classify a particular web page in isolation, its neighbouring web pages could help us to do this task with higher accuracy. In other words, if x_i and x_j are connected via a hyper-link and x_i is classified wrongly by a supervised classifier, knowing that x_i belong to the same genre class as an easily-categorized x_j, makes labelling x_i easier. The min-cut algorithm can take advantage of easily classified neighbours to correctly classify hard instances. For example, if a blog web page is wrongly classified by a supervised classifier, but its neighbouring web pages are correctly classified as blog, the min-cut algorithm uses this information and pulls the wrongly classified web page into the right cut (category).

5.3.2 Selecting Neighbours

A web page w has different kind of neighbours on the web such as parents, children, siblings, grand parents and grand children which are mainly differentiated based on the distance to the target web page as well as the direction of the links [117]. Since the identification of children of a web page (i.e. web pages which have direct links from the target web page) is a straightforward task as their URLs can be extracted from the HTML version of the target web page, in this study, we explore the effect of the target web pages' children on genre classification. Therefore, for every web page in the data set, we extracted all its out-going URLs and downloaded them as unlabelled data. [1] However, using all these neighbouring pages could hurt the genre classification accuracy because web pages are noisy (e.g. links to advertisements) and therefore, some neighbours could have different genres than the target page. In order to control the negative impact of such neighbours, we could preselect a subset of neighbours whose content is close enough to the target page. To implement this idea, we computed the cosine similarity between the web page w and its neighbouring web pages and used different threshold to select the neighbours. If u is a neighbour of w and $\overrightarrow{u}$ and $\overrightarrow{w}$ are the representative feature vectors of these two web pages respectively, we compute the cosine similarity between these two web pages using the following formula:

$$\cos(\overrightarrow{w}, \overrightarrow{u}) = \frac{\overrightarrow{w} \cdot \overrightarrow{u}}{\| \overrightarrow{w} \| \| \overrightarrow{u} \|} = \frac{\sum_{i=1}^{n} w_i \times u_i}{\sqrt{\sum_{i=1}^{n}(w_i)^2} \times \sqrt{\sum_{i=1}^{n}(u_i)^2}} \tag{5.4}$$

where n is the number of the dimensions of the vectors and w_i is the value of the ith dimension of the vector $\overrightarrow{w}$. Table 5.1 shows the number of unlabelled web pages with different cosine similarity threshold. Since word unigrams binary representation model yields the best result for content-based genre classification, we used this representation of web pages to construct their feature vectors. We divided the labelled data into 10 folds while we ensured that all the web pages from the same websites are in the same fold. We used 8 folds for training, one fold for validation and one fold for testing. We learnt the best cosine similarity

[1] We did not make any distinction between in-domain and out-of-domain links in the experiments presented in this chapter.

Cosine similarity	# of unlabelled web pages	Average # of neighbours per labelled page
≥ 0	103,372	40.65
≥ 0.1	98,824	39.08
≥ 0.2	87,834	34.23
≥ 0.3	70,602	26.46
≥ 0.4	50,232	17.52
≥ 0.5	28,437	8.62
≥ 0.6	13,919	3.77
≥ 0.7	7,241	1.86
$\geq$ **0.8**	**3,772**	**0.98**
≥ 0.9	1,732	0.44

Table 5.1: Number of unlabelled web pages with different cosine similarity thresholds

threshold using the validation data and then evaluated on the test data.

5.3.3 Semi-supervised Binary Min-cut

5.3.3.1 Formulation of the Model

The formulation of our semi-supervised min-cut for genre classification involves the following steps:

1. We divide the multi-class genre classification task into fifteen binary classification subtasks e.g. blog and non-blog; news and non-news.

2. For each classification subtask, we build a weighted and undirected graph where web pages are represented as vertices. This graph also has two vertices s (source) and t (sink) as classification nodes which correspond to the two labels e.g. blog and non-blog. Following the definition in Blum and Chawla [20], we call the vertices s and t classification vertices, and all other vertices (labelled, test, and unlabelled data) example vertices.

3. We connect the labelled training web pages by edges with a high constant weight to the classification nodes that they belong to. The reason for this is that we do not want the min-cut algorithm to reverse the labels of the training data.

4. Then we connect each test and unlabelled web page to the classification nodes via a weighted edge where weights are the probability of them belonging to the classification nodes. An SVM supervised genre classifier based on word unigrams binary representation of web pages is used to set the edges' weights.

5. Then we connect web pages which are linked together via a hyper-link and set the weight to 1.

6. After the graph is constructed, we employ the maximum-flow algorithm to find the minimum $s-t$ cut of the graph and split the graph into two disconnected parts, the s part and the t part. The vertices which are on the s part are classified as the genre class which s represent and the vertices which are on the t part are classified as the genre class which t corresponds to.

An Example Graph

An example graph is shown in Figure 5.1 where vertex s (source) corresponds to the genre class blog while vertex t (sink) corresponds to the class non-blog. Moreover, the red, yellow and green vertices in the graph represent the training, test and unlabelled web pages respectively. As it is shown in Figure 5.1, the edges' weights connecting the example vertices to the classification vertices represent how likely the example vertices are preliminarily predicted as blog or non-blog by a supervised classifier. For example, web pages p_2 and p_3 are predicted as non-blog with confidence of 0.75 and 0.8 respectively.

The effect of unlabelled data as neighbouring pages, as well as the homophily assumption of the min-cut algorithm can be seen in Figure 5.1. Web page p_2 has two unlabelled neighbours p_4 and p_5 with high probability of being a blog. As a result, the minimum cut of the graph (red dashed line) classified web page p_2 as a blog too, by putting it on the s side of the graph. On the other hand, page p_6 and p_7 which are the neighbours of page p_3 did not reverse the classification result of page p_3. The reason is that the predicted classes for p_6 and p_7 are the same as for p_3. Therefore, all three nodes p_3, p_6 and p_7 are positioned on the t side of the graph.

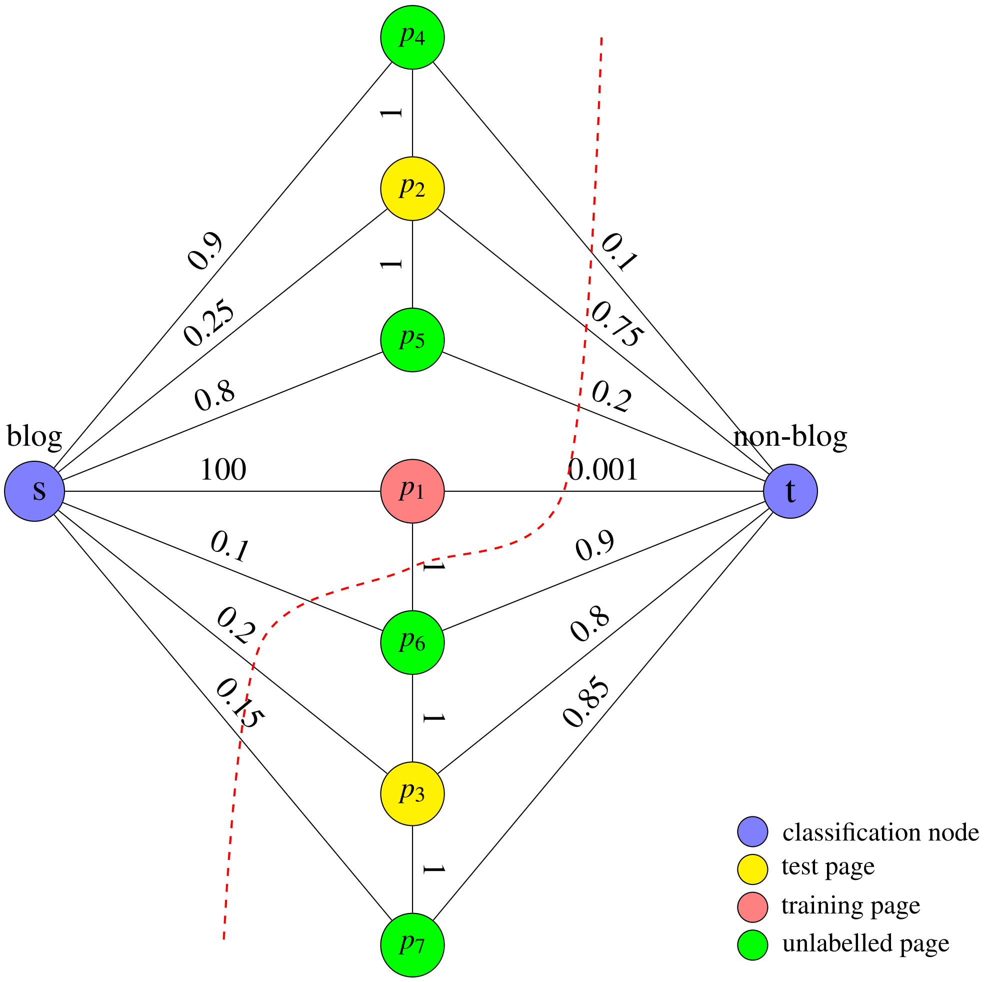


Figure 5.1: An example graph which depicts how semi-supervised binary min-cut algorithm works

5.3.3.2 Results and Discussion

We use the supervised SVM classifier with word unigram features as a baseline to compare with our proposed semi-supervised min-cut approach. Note that this supervised SVM classifier is also used to provide the edges' weights connecting example vertices to the classification vertices in the min-cut approach. Table 5.2 shows the results for the supervised SVM classifier and the semi-supervised min-cut algorithm for the best cosine similarity thresholds.

Although the results presented in Table 5.2 show very high accuracies for the individual genre classes, it must be noted that accuracy is not a good indicator of the performance in this case. Splitting the multi-class classification problem into 15 binary sub-problems results in 15 unbalanced data sets with high numbers of negative and low numbers of positive instances and binary classifiers that recognize only negative examples still achieve a high accuracy. Therefore, recall, precision and F-measure are more informative measures in binary classifications with unbalanced data.

By comparing the F-measure of these two approaches in Table 5.2, we see that the semi-supervised min-cut algorithm significantly outperformed the supervised SVM classifier for some of the genre categories such as editorial, news, interview and instruction for the learnt cosine similarity equal or greater than 0.8 (it must be noted that the result of the min-cut algorithm when we used all the neighbouring pages was much lower than the supervised SVM classifier due to the noise). Table 5.2 also shows that, although the semi-supervised min-cut algorithm resulted in lower accuracy, precision and F-measure for genre categories review and faq, we observe improvement in terms of recall for these genre classes.

However, we observe no improvement or even decline in F-measure for some genre categories such as forum, faqs and company home pages. This results illustrate that our semi-supervised min-cut algorithm is not effective uniformly across all the genre classes. Three reasons could have contributed to these results. First, the homophily assumption does not hold for all the genre classes. Second, we might need to examine other neighbours of the target web pages such as parents, siblings, grand parents or grand children in order to benefit from the homophily property. The third reason could be that our neighbour selection method is not efficient. Although increasing the cosine similarity threshold leads to a high reduction of the noise, this process could also remove web pages which are informative and have the same genre as the target pages. To see if this is the case, we manually checked the output of the neighbour selection method for a few sample web pages in our dataset, and we observed that in some cases the neighbours with the same genre class as the target web page are removed by this selection method, because their contents were not similar enough. Therefore, in some cases, our neighbour selection method removed the informative neighbours along side of the uninformative ones.

In this section, we investigated the performance of a binary semi-supervised min-cut algorithm in genre classification by dividing a multi-class classification task into several binary classification tasks. The next section describes the formu-

Genre	Binary semi-supervised min-cut				Binary supervised SVM			
	Acc	R	P	F	Acc	R	P	F
php	**0.985**	**0.872**	**0.933**	**0.901**	0.983	0.878	0.902	0.890
forum	0.991	0.875	0.992	0.930	0.991	0.886	0.988	0.934
review	0.971	0.662	0.880	0.755	0.972	0.654	0.906	0.760
news	**0.969**	**0.724**	**0.889**	**0.798**	0.968	0.715	0.877	0.788
com	0.977	0.770	0.839	0.803	0.977	0.779	0.841	0.809
shop	**0.981**	**0.825**	**0.906**	**0.864**	0.979	0.812	0.898	0.853
instruction	**0.981**	**0.736**	**0.929**	**0.821**	0.980	0.723	0.908	0.805
recipe	0.997	0.979	0.991	0.985	0.997	0.976	0.988	0.982
blog	**0.980**	**0.718**	**0.936**	**0.813**	0.978	0.697	0.929	0.796
bio	0.990	0.864	0.972	0.915	0.989	0.855	0.963	0.906
editorial	**0.972**	**0.710**	**0.917**	**0.800**	0.971	0.690	0.915	0.787
faq	0.987	0.806	0.931	0.864	0.988	0.801	0.953	0.870
edu	0.991	0.926	0.949	0.937	0.990	0.923	0.948	0.936
story	**0.988**	**0.783**	**0.960**	**0.862**	0.987	0.755	0.959	0.845
interview	**0.986**	**0.743**	**0.959**	**0.837**	0.984	0.697	0.942	0.801

Table 5.2: Accuracy, Recall, Precision and F-measure for binary semi-supervised min-cut and binary supervised SVM genre classification. The results of the categories in bold are significantly improved by binary min-cut algorithm.

lation and results of the semi-supervised multi-class min-cut algorithm described in Section 5.2.5.

5.3.4 Semi-supervised Multi-class Min-cut

5.3.4.1 Formulation of the Model

The formulation of semi-supervised multi-class min-cut for genre classification involves the following steps:

1. We built the weighted and undirected graph $G = (V, E)$ where vertices are the web pages (labelled and unlabelled) and the edges represent the hyperlinks between the web pages and set the weights to 1.

2. For training nodes set the cost of the correct label to zero and all other labels to a large constant.

3. For test labelled nodes and unlabelled nodes, we set the cost of each label

using a supervised classifier (SVM) using the following formula:

$$c(w,l) = 1 - p_l(w) \tag{5.5}$$

where $c(w,l)$ is the cost of label l for web page w and $p_l(w)$ is probability of w belonging to the label l which is computed by a supervised SVM using word unigrams binary representation of web pages.

4. Set $d(i,j)$, which denotes the distance between two labels i and j, to one if $i \neq j$ and zero otherwise.

5. Employ Boykov et al.'s [22] algorithm to find the minimum total cost using multiway min-cut algorithm.

class	R	P	F
php	**0.928**	**0.850**	**0.887**
forum	0.925	0.977	0.951
review	0.895	0.832	0.862
news	**0.897**	**0.798**	**0.845**
com	0.897	0.891	0.894
shop	**0.860**	**0.965**	**0.910**
instruction	**0.870**	**0.914**	**0.892**
recipe	**0.994**	**0.991**	**0.993**
blog	**0.889**	**0.879**	**0.884**
bio	**0.905**	**0.948**	**0.926**
editorial	**0.800**	**0.932**	**0.861**
faq	0.902	0.841	0.870
edu	**0.957**	**0.963**	**0.960**
story	**0.902**	**0.943**	**0.922**
interview	**0.870**	**0.809**	**0.839**
overall accuracy = **90.11%**			

Table 5.3: Recall, Precision and F-measure for multi-class semi-supervised min-cut genre classification.

class	R	P	F
php	0.938	0.798	0.862
forum	0.943	0.974	0.958
review	0.872	0.859	0.866
news	0.894	0.782	0.835
com	0.920	0.874	0.897
shop	0.849	0.950	0.897
instruction	0.866	0.889	0.877
recipe	0.988	0.988	0.988
blog	0.865	0.841	0.853
bio	0.884	0.926	0.905
editorial	0.765	0.926	0.837
faq	0.866	0.879	0.872
edu	0.950	0.969	0.959
story	0.864	0.941	0.901
interview	0.827	0.785	0.805
overall accuracy = 88.98% [2]			

Table 5.4: Recall, Precision and F-measure for multi-class supervised SVM genre classification using word unigrams feature set

5.3.4.2 Results and Discussion

The results which are presented in Tables 5.3 and 5.4 show that multi-class min-cut algorithm significantly outperforms the content-based classifier for the cosine

[2] Please note that in this experiment we had less training data because we used 8 folds for training, one fold for validation and one fold for testing. As a result, the accuracy of word unigrams is slightly lower than the result reported in Table 4.2.

similarity equal or greater than 0.8 [3]. The results, which show the same trend observed in the binary min-cut results, show that some genre classes such as news, editorial, blog, interview and instruction benefited more than other genre classes from the neighbouring web pages. Genre categories with improved results are shown in bold in Table 5.3. The homophily property of these genre categories was the reason behind this improvement. For example, the fact that a news article is more likely to be linked to other news articles, whereas an editorial is more likely to be linked to other editorials, helped us to differentiate these two categories further. On the other hand, we observe no improvement or even decrease in F-measure for some genre categories such as frequently asked questions, forums and company home pages.

Figure 5.2 depicts the learning curve of the the semi-supervised multi-class min-cut algorithm on the validation data, i.e., the accuracy depending on the cosine similarity threshold used in the neighbour selection method. The learning curve shows that when we used all the neighbours, the genre classification accuracy yielded by semi-supervised min-cut was much lower than the supervised SVM. However, increasing the cosine similarity threshold in the neighbour selection technique resulted in improving the accuracy of the semi-supervised min-cut algorithm and at the cosine similarity threshold equal to 0.8, this algorithm outperforms the model based on the supervised SVM.

5.4 Summary

In this chapter, we investigated the feasibility of applying a semi-supervised learning approach to genre classification on the web. We give a brief introduction to semi-supervised classification methods, and propose a semi-supervised minimum cut algorithm for genre classification by exploiting the hyper-link connections between the web pages [5]. We hypothesise that the homophily property (i.e. entities with similar labels are more likely to be connected) exists for genre classes and web pages in the neighbourhood, i.e. those connected by hyper-links, should help reveal the class of the target document.

This hypothesis helped us to significantly improve the genre classification result using a semi-supervised min-cut algorithm. We employed the children of the

[3] McNemar test at the significance level of 5%

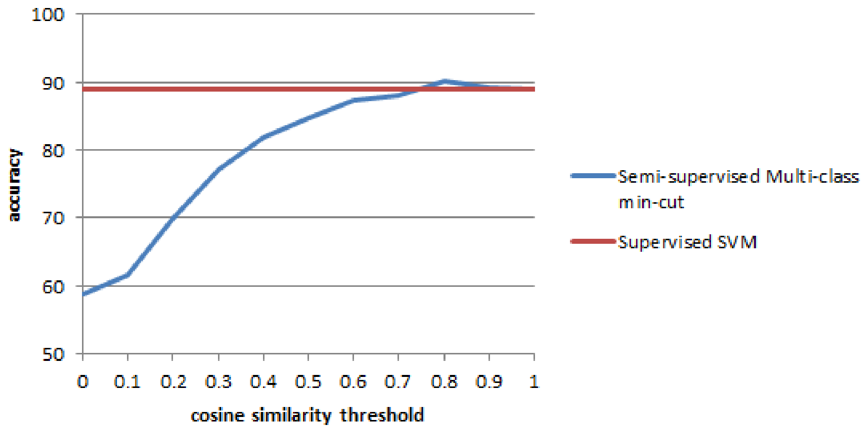


Figure 5.2: The learning curve of the the semi-supervised multi-class min-cut algorithm on the validation data which illustrates the accuracy depending on the cosine similarity threshold used in the neighbour selection method.

target web pages as unlabelled data. The results of this method which takes advantage of the graph structure of the web show that some genre classes benefit more than others from the neighbouring web pages.

So far, we have discussed supervised and semi-supervised learning models for genre classification on the web, and the experimental results show that the proposed models are all effective, with the semi-supervised min-cut framework achieving the best performance. Therefore, we might wonder that since we can automatically classify genre classes on a designed dataset at a relatively high level of accuracy, can we apply genre classification on random web pages and achieve the same performance? In order to answer this question, in the next chapter, we will investigate the reliability of human genre annotation as well as the performance of closed-set and open-set genre classification algorithms on a corpus of random web pages.

Chapter 6

Extending Genre Classification from a Designed to a Random Corpus

6.1 Introduction

In Chapter 3, we described different phases of the construction of the Leeds Web Genre Corpus Balanced-design (LWGC-B). In the corpus compilation step, we noted that we chose to build a designed corpus as opposed to a random corpus because we wanted to have a balanced collection with a large number of web pages per genre category. Therefore, we did a focused search for collecting web pages which belong to our predefined genre categories. The result of human annotation which was conducted in Amazon Mechanical Turk showed high inter-annotator agreement and, as a result, the corpus was annotated reliably. However, the questions that we are seeking to answer in this chapter are threefold: firstly, can we achieve such high inter-annotator agreement on more arbitrary web pages? Secondly, how good is the coverage of our genre inventory when applied to web page selection that is not guided by focused search for particular genres? Thirdly, can a model trained on our designed corpus successfully be used for automatic genre classification on random web pages? Section 6.2 focuses on our first and second research questions, while Section 6.3 consentrates on providing the answer to the third question.

6.2 LWGC-R: Human Annotation Study on Random Web Pages

In order to answer the first and the second questions stated above, we repeated the same annotation study that we conducted in Chapter 3, on random web pages. The following subsections describe the corpus collection, the corpus annotation and the results of the experiment in detail leading to the creation of LWGC-R(andom) corpus.

6.2.1 LWGC-R: Web Page Collection

The BootCat toolkit [8] is often used to collect a varied sample of web pages in a given language, using seed keywords that are fed as queries to search engines, and we follow this approach. Two things have to be noted when using this method that distinguish it from a truly random web page collection (which would only be possible if we had access to a snapshot of the whole web): Firstly, if specific topics such as *Rafael Nadal, tennis* are fed as seeds then we of course will get topic-specific pages back. Therefore we need to choose very general seeds in our case and we follow Sharoff [137] and use a list of the 500 most frequent words extracted from the BNC corpus, which are mostly function words, as seeds. The BootCat tool creates a list of *n*-tuples out of the seed words by randomly combining them. We use 3-tuples in this experiment (e.g., *have, we, which*). These 3-tuples are used as keywords to query a search engine.

Secondly, as search engines, such as Google, rank and retrieve web pages based not only on keyword occurrence but also on their popularity, we actually do not get a truly random result either but rather a snapshot of popular web pages. In our case, this is not a disadvantage as for practical purposes being able to label the most used parts of the web is important. However, as there is a genre bias when using the very top-most results which tend to be commercial home pages [87], we ignored the first 30 URLs retrieved for each query and collected URLs which were ranked from 31th to 50th positions. Overall, fifty queries were sent to a search engine by the BootCat toolkit and we collected 1000 URLs. After the URLs collection phase, we downloaded these web pages using the KrdWrd tool [147].

6.2.2 LWGC-R: Annotation Procedure

Similar to the annotation of our designed corpus in Chapter 3, we also used crowdsourcing for the annotation of the random web pages collected in the previous section. In fact, we carried out exactly the same annotation study that we conducted for our designed corpus using Amazon Mechanical Turk. The definitions of the genre categories as well as example web pages for each class were provided as part of the annotation guidelines. Annotators had the option to choose one of our 15 predefined genre categories or the option other for each web page. We set the number of annotations per web page to five. Moreover, the same quality control measures used in the experiment described in Section 3.2.3.1 (e.g. trap questions, qualification test and having high approval rate), were also adopted in this experiment.

6.2.3 LWGC-R: Results of Annotation Study

In order to measure the reliability of the annotation result, we calculated the intercoder agreement measures. For this experiment, the percentage agreement is 78.15% and π is 0.712 which shows substantial agreement between the annotators and therefore we can consider the annotation reliable. We also calculated π for the individual genre labels (Table 6.1). The results show that for all the categories except story and interview, the π value is above 0.6. Therefore, we can consider them reliable. Quite importantly, the agreement for the category other is high which means that the current genres cannot just be easily delimited from each other (as in LWGC-B) but also from other arbitrary web pages.

However, the π value for the two genre classes story and interview is around zero, despite the fact that they have a very high observed or percentage agreement (99.9% and 99.8% respectively).

A π of around zero usually indicates very poor agreement. However, this interpretation of a chance-corrected agreement coefficient like π and κ only makes sense if the categories occur reasonably often [45]. Skewed categories can bias chance-corrected agreement measures such as π and κ to a low value even when the proportion of annotators agreement is very high. To illustrate this point, we provided the number of times that each category was chosen by the annotators (fourth column of Table 6.1). As you can see these numbers are very skewed ranging from 2 for the genre class *story*, to 2089 for the category other. These statistics show that prevalence was indeed the reason behind the very low inter-

Genre Labels	Percentage agreement	π	N.T.C.A
Personal Homepage	0.997	0.741	39
Company/ Business homepage	0.888	0.646	961
Educational Organization Homepage	0.993	0.707	64
Personal blog /Diary	0.979	0.611	83
Online shops	0.966	0.774	414
Instruction/ How to	0.946	0.645	423
Recipe	0.999	0.928	43
News Article	0.952	0.791	626
Editorial	0.991	0.667	67
Conversational Forum	0.994	0.738	51
Biography	0.998	0.892	28
Frequently Asked Questions	0.993	0.757	58
Review	0.996	0.775	48
Story	0.999	-0.0004	2
Interview	0.998	-0.0008	4
Other	0.847	0.685	2089

Table 6.1: Inter-coder agreements for individual categories show substantial agreement among the coders on LWGC-R. N.T.C.A stands for number of times chosen by the annotators. For example the category story has been chosen only two times by the annotators.

coder agreement for the categories story and interview in this study. Due to the low number of samples of these two categories in the random corpus, we cannot draw definite conclusions with regard to the reliability of these two categories.

The comparison between the results of the annotation study on the designed corpus and the random web pages reveals that generally the π values for the more randomly selected web pages are lower. This could be due to two reasons: First, it could be the influence of the prevalence factor on π because the random dataset is highly skewed. Second, it is harder to obtain a high inter-coder agreement for random web pages as these will include more borderline cases. In order to provide more insight into this annotation study, we also compute the percentage of each type of inter-annotator agreement (Table 3.5). For 59.40% of the web pages all the five annotators agreed and for more than 80% of the data at least four annotators agreed which indicates high level of agreement between the coders. However, comparing the two Tables 6.2 and 3.6 reveals that the main reason behind lower agreement for random web pages is that annotators find it harder to agree on the

ID	Types of inter-annotator agreement	# of web pages	% of web pages
1	5,0	594	59.40%
2	4,1	219	21.90%
3	3,1,1	31	3.10%
4	3,2	122	12.20%
5	2,1,1,1	4	0.40%
6	2,2,1	29	2.90%
7	1,1,1,1,1	1	0.10%

Table 6.2: Distribution of different types of inter-annotator agreement in LWGC-R

random web pages compared to prototypical web pages in the designed corpus. Comparing Tables 6.2 and 3.6 also shows that 12.20% of the web pages in the random corpus caused the type 4 inter-coder agreement (i.e. 3,2: three annotators agreed with each other while the other two disagreed with the majority but agreed with each other), whereas, this kind of disagreement is only 2.92% in the designed corpus. As noted in Section 3.2.6, this kind of disagreement is caused by two kinds of web pages: web pages that can belong to more than one genre category and border-line web pages which are ambiguous. The fact that this kind of web pages are much more frequent in the random corpus compared to the designed corpus indicates that we are more likely to see this kind of web pages in the random snap-shot of the web than the designed corpora which usually contain prototypical examples of the genre classes. Table 6.3 shows the ten most confused genre categories which cause this kind of disagreement. It reveals that the two most confused categories are other and company home pages. There are 24 web pages in this dataset which two out of five annotators chose company homepage label for them whereas the other three annotators chose the category other or vise-versa. Figure 6.1 is an example of such a web page.

Nevertheless, the result of this study still shows substantial agreement between the annotators and as a result it was a successful annotation study.

6.2.4 LWGC-R: The Gold Standard Corpus

In order to use the corpus of the random web pages in a supervised machine learning experiment, we need to convert the annotated dataset into a gold standard. We employed the majority vote strategy to assign the final label to the disagreed web

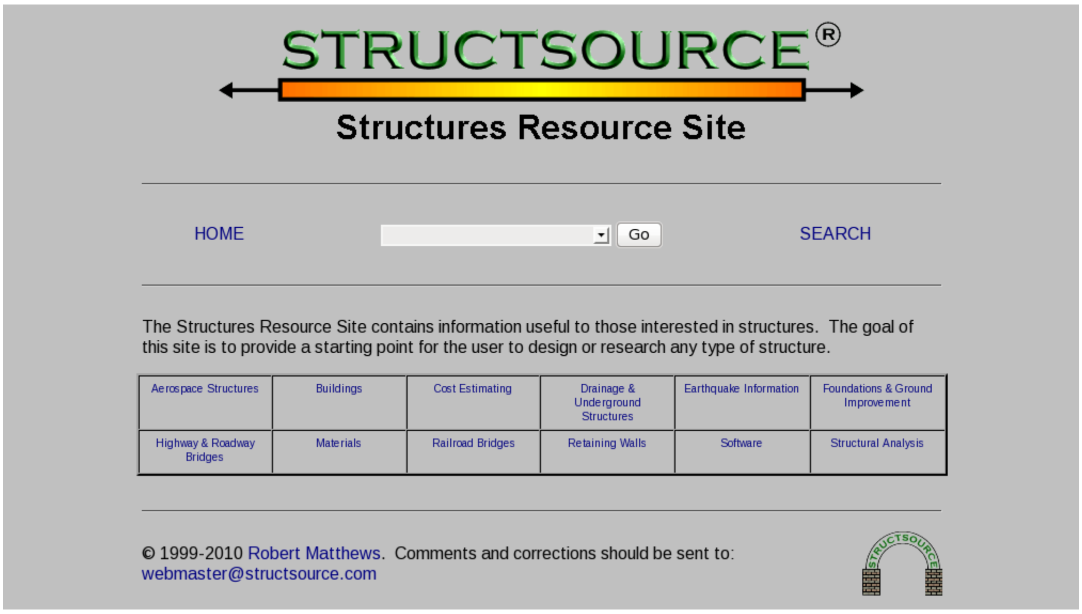
24	other and com
22	how-to and other
9	com and shop
6	other and news
6	news and editorial
4	other and blog
4	other and shop
4	how-to and com
4	com and edu
3	review and shop

Table 6.3: The ten most confused genre categories in the LWGC-R corpus which cause the type 4 inter-coder agreement where three annotators agreed with each other while the other two disagreed with the majority but agreed with each other. We show the number of web pages for which the two genre classes caused this type of disagreement.

pages in this experiment. This method was also used to create the gold standard for the LWGC-B corpus. As shown in Table 6.2, there are seven possible types of inter-annotator agreement when there are five annotators. However, there is no majority for the last three types. Therefore, as we did not have a majority vote for 34 web pages, we excluded them from the gold standard corpus.

Table 6.4 illustrates the number of web pages per genre category in the random web pages which indicates a very skewed distribution. 43.8% of pages did not belong to any of our 15 predefined genre categories, indicating a somewhat more than 50% coverage for our 15 genres. Researchers in genre classification have come up with long lists of genre classes, e.g., 292 genre labels in the Syracuse corpus [32] or 500 genre labels listed in [40]. Therefore, the web pages categorized as other in this experiment could belong to any genre class in these taxonomies. However, as most phenomena in language are Zipfian, most genre labels will after a certain point of adding genre classes be very rare.

Moreover, while genre categories such as company home pages and news articles comprise a high percentage of the total number of web pages in the LWGC-R corpus, other genre categories such as biography and personal home pages have very few web pages assigned to them. Table 6.4 also shows that no web page represents the genre categories story and interview in this corpus.



STRUCTSOURCE®

Structures Resource Site

HOME Go SEARCH

The Structures Resource Site contains information useful to those interested in structures. The goal of this site is to provide a starting point for the user to design or research any type of structure.

Aerospace Structures	Buildings	Cost Estimating	Drainage & Underground Structures	Earthquake Information	Foundations & Ground Improvement
Highway & Roadway Bridges	Materials	Railroad Bridges	Retaining Walls	Software	Structural Analysis

© 1999-2010 Robert Matthews. Comments and corrections should be sent to: webmaster@structsource.com

Figure 6.1: An example web page which caused confusion between the categories company home page and other.URL: http://structsource.com/index.html

6.3 Automatic Genre classification on Random Web pages

The aim of this section is to investigate the performance of AGI systems on the LWGC-R corpus constructed in the previous section. The main difference between this corpus and the LWGC-B corpus built in Chapter 3 is that the LWGC-R corpus includes the category other which includes all the web pages which do not belong to any of our 15 predefined categories. This property of the LWGC-R corpus allows us to examine the performance of open-set genre classification algorithms which can reject to assign any predefined label to a web page. In other words, open-set classifiers can detect outliers or noise. [1] Although, as noted in Section 2.7.3, Pritsos and Stamatatos [116] and Stubbe et al. [152] proposed algorithms for open-set genre classification, due to the lack of a dataset with noise, the real performance of these algorithms could not be examined.

Therefore, in this section, first we examine the performance of a closed-set classifier on the LWGC-R corpus without noise (i.e. the class other) in Section 6.3.1 and then we investigate the performance of an open-set classifier on the whole LWGC-R corpus including the class other in Section 6.3.2. This is the

[1] By noise we mean any web page which does not belong to any of our predefined genre classes.

Category	# Web pages	% of the corpus
other	438	45.34%
com	167	17.29%
news	117	12.11%
shop	79	8.18%
how-to	76	7.87%
blog	16	1.66%
edu	12	1.24%
editorial	12	1.24%
faq	10	1.04%
review	9	0.93%
recipe	9	0.93%
php	8	0.83%
forum	8	0.83%
bio	5	0.52%
story	0	0%
interview	0	0%
in total 966 web pages		

Table 6.4: Distribution of genre categories in the gold standard LWGC-R corpus

first time in genre classification that the performance of an open-set classifier in detecting noise is being investigated.

6.3.1 Closed-set Genre Classification

In Chapter 4, we investigated the performance of various lexical and POS-based as well as text statistics features in genre classification. The result showed that word unigrams outperformed other features on the LWGC-B corpus with the accuracy of 89.32%. The question is: can we achieve the same high result on the random web pages? In order to answer this question, in this section we test the discriminative power of the genre classification model built using the LWGC-B corpus, on the LWGC-R corpus.

6.3.1.1 Experimental Setup

In this experiment, we use the LWGC-B gold standard corpus as the training data and the gold standard LWGC-R corpus of the random web pages built in Section 6.2.4 without the class other as unseen test data.

Moreover, in this experiment we used word unigrams binary representation as features, because they outperformed other features in the set of experiments presented in Chapter 4. Although feature combination produced a better result, this improvement was insignificant. We converted web pages to plain text using KrdWrd tool. Then, we tokenized each document using the Stanford tokenizer (included as part of the Stanford part of speech tagger [157]) and converted all the tokens to lower case. For machine learning we used one-versus-one multi-class SVM implemented in Weka [57] with the default setting. The next section presents the results.

6.3.1.2 Results and Discussion

In order to evaluate the experiment conducted in the previous section, we computed overall accuracy as well as precision, recall and F-measure for the individual genre classes. The results which are presented in Table **??** show 80.68% overall accuracy. Although this is quite high, it is approximately 9% lower than the result on the LWGC-B corpus. Moreover, the F-measure values for the individual categories depict a very mixed picture. While we observe high value F-measure for genre categories such as biography, news and company home pages, the results show low F-measure for categories such as frequently asked questions and editorial. In order to have a full picture of the classification result, we also provided the confusion matrix of this classification experiment in Table 6.6. The left most column shows the true genre labels of the web pages while the top row of the table illustrates the genre labels assigned by the classifier. The fact that the LWGC-R data set contains more border-line web pages which are harder to classify automatically, is the reason for the lower F-measure on the random dataset. These border-line web pages even caused disagreement between humans which resulted in lower inter-annotator agreement on the LWGC-R corpus compared to the LWGC-B corpus.

6.3.2 Open-set Genre Classification

Until now, we focused on closed-set multi-class genre classification algorithms which assign one predefined genre label to each web page. In order to apply such algorithms on the real web, we need to create a dataset which contains all the possible genre classes. However, creating such a dataset is very expensive or even completely impractical, because the web is an evolving phenomenon and new

Genre	Precision	Recall	F-measure
php	0.545	0.750	0.632
forum	0.700	0.875	0.778
review	0.444	0.444	0.444
news	0.898	0.829	0.862
com	0.849	0.910	0.879
shop	0.829	0.734	0.779
how-to	0.923	0.789	0.851
recipe	1.000	0.889	0.941
blog	0.800	0.500	0.615
bio	1.000	1.000	1.000
editorial	0.267	0.333	0.296
faq	0.333	0.800	0.471
edu	0.750	0.750	0.750
story	n/a	n/a	n/a
interview	0.000	n/a	n/a
overall accuracy = 80.68%			

Table 6.5: Results for closed-set genre classification of random web pages with word unigram binary representation as feature set.

genres are emerging all the time. Therefore, as discussed in detail in Section 2.7.3, open-set classification algorithms which do not require a complete set of genre labels are more suitable for genre classification on the web. In this section, we reimplement the open-set classification technique proposed by Stubbe et al. [152] and test it on the LWGC-R corpus. As noted in Section 6.3, the novelty of this experiment lies in the use of a dataset which contains noise. This characteristic of the dataset enables us to investigate the outlier detection property of the open-set genre classifiers. The following subsections describe the experimental set up and the results.

6.3.2.1 Experimental Setup

As noted in Section 2.7.3, Stubbe et al. [152] proposed a cascading classifier which can be used as an open-set classification algorithm. A cascading classifier is a sequential ensemble of binary classifiers ordered based on a particular selection scheme. Therefore, in the first phase, we build binary classifiers for individual genre classes using the LWGC-B corpus as the training data and applied them on the whole LWGC-R corpus as the test data.

	php	forum	review	news	com	shop	how-to	recipe	blog	bio	editorial	faq	edu	story	interview
php	6	1	1	0	0	0	0	0	0	0	0	0	0	0	0
forum	0	7	0	0	0	0	0	0	0	0	0	1	0	0	0
review	0	0	4	0	1	2	1	0	0	0	0	0	0	0	1
news	0	1	0	97	6	1	0	0	1	0	11	0	0	0	0
com	1	0	0	0	152	9	0	0	0	0	0	4	1	0	0
shop	2	0	2	0	13	58	1	0	0	0	0	1	2	0	0
how-to	0	0	1	2	3	0	60	0	1	0	0	9	0	0	0
recipe	0	0	0	0	1	0	0	8	0	0	0	0	0	0	0
blog	2	1	0	1	1	0	2	0	8	0	0	0	0	0	1
bio	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0
editorial	0	0	1	6	1	0	0	0	0	0	4	0	0	0	0
faq	0	0	0	1	0	0	1	0	0	0	0	8	0	0	0
edu	0	0	0	1	1	0	0	0	0	0	0	1	9	0	0
story	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
interview	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 6.6: Confusion matrix for closed-set genre classification of random web pages

In the second phase, we need to define a selection scheme which orders the binary classifiers. In Stubbe et al. [152] the binary classifiers are ordered in two different ways based on their performance on the training set. In the first method, they used F-measure value and the genre class with highest F-measure in the training set is the first classifier to label the test web pages. In the second method, they used ordering by dependencies and recall which is a much more sophisticated approach. In this method, a preliminary sequence of classifiers is ordered by the recall value of these classifiers on the training data set. In the second step, a dependency graph is generated using the confusion matrices of these classifiers. In the dependency graph, a directed edge is created from class g_i to g_j if web pages in class g_j are misclassified as g_i. The weight of this edge is the number of misclassified web pages. Then this dependency graph is used to rearrange the ordering of the binary classifiers. If a classifier g_i is followed by g_j and has a dependency edge with g_j , g_j is placed before g_i . If there are cycles in the dependency graph, g_i is only rearranged if the label of the outgoing edge to g_j is higher than that of the incoming edge from g_j.We reimplemented both selection schemes. We divided the LWGC-B corpus into 10 folds and used 9 folds for training and one fold as validation data for learning the ordering of the binary classifiers.

In this open-set classification technique, the first binary classifier has the advantage to classify all the test instances and assign the final label to the web pages which it classifies as positive. Then all the web pages classified as negative are passed to the next binary classifier. This process continues to the last binary classifier. A test instance can remain unclassified if it is not positively identified by any of the binary classifiers.

6.3.2.2 Results and Discussion

Table 6.7 presents the order of the binary classifiers in the two selection schemes described in the previous section. It must be noted that the both selection schemes performed exactly the same. This is not surprising, as there is a very small difference between the ordering of the binary classifiers in the two schemes. The result of the classification which is presented in Table 6.8 shows 59.32% accuracy. This result is considerably lower than the accuracy of the closed-set classification presented in Section 6.3.1.2 (59.32% vs. 80.68%). Also, precision, recall and F-measure for all the categories except forum experience a moderate or a sharp fall. It must be noted that noise pages affect mainly precision, whereas recall is mostly affected by the open-set genre classification model.

Selection scheme	Order of binary classifiers
F-measure	recipe→ edu → forum → bio → php → faq → shop → story → com → how-to → interview → blog → news → editorial → review
Recall and dependency graph	recipe → edu → forum → php → bio → shop → faq → com → story → how-to → news → blog → interview → editorial → review

Table 6.7: The order of binary classifiers using two different selection schemes.

In order to provide a more detailed picture of the classification result, we also present the confusion matrix in Table 6.9 which shows that almost a quarter of the web pages that belong to class other are misclassified as one of our 15 predefined genre categories, especially as company home pages, how-to and personal home pages. Therefore, the noise (i.e. category other) is the main reason for the poor performance of the open-set classification algorithm. This shows that although we can yield a high accuracy in the closed-set genre classification, this good result is not easily achievable on the real web which contains noise.

Genre	Precision	Recall	F-measure
php	0.114	0.500	0.186
forum	0.875	0.875	0.875
review	0.667	0.222	0.333
news	0.845	0.419	0.560
com	0.725	0.647	0.684
shop	0.936	0.557	0.698
how-to	0.509	0.355	0.419
recipe	1.000	0.778	0.875
blog	0.333	0.125	0.182
bio	0.294	1.000	0.455
editorial	0.167	0.083	0.111
faq	0.133	0.400	0.200
edu	0.429	0.500	0.462
story	n/a	n/a	n/a
interview	n/a	n/a	n/a
other	0.576	0.701	0.632
overall accuracy = 59.32%			

Table 6.8: Precision, recall and F-measure for open-set genre classification of random web pages with word unigrams binary representation as feature set.

6.4 Conclusions

In this chapter, we present the LWGC-R corpus, which is the first reliably annotated random web genre collection. We expanded our human annotation method successfully to this corpus where the pages to be annotated are collected in a more arbitrary way among web pages returned by search engines. We showed that humans can reliably distinguish our 15 predefined genre categories from each other as well as from all other web pages in the LWGC-R corpus. Therefore, our annotation scheme is applicable to random web pages as well.

In addition, the LWGC-R corpus allowed us to investigate coverage of our genre inventory. We show that our 15 genre categories are sufficient to cover the majority but not the vast majority of random web pages. Moreover, the skewed distribution of genre classes in this corpus illustrates that some genre categories are more frequent on the web than others. Web pages which did not belong to any of our 15 predefined genre classes were labelled as other in this corpus.

We also investigated the performance of automatic genre classification models on LWGC-R corpus. First, we examined the performance of a closed-set genre classification model on the LWGC-R corpus without noise. The model was trained

	php	forum	review	news	com	shop	how-to	recipe	blog	bio	editorial	faq	edu	story	interview	other
php	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4
forum	0	7	0	0	0	0	0	0	0	0	0	0	0	0	0	1
review	0	0	2	0	0	1	0	0	0	0	0	0	0	0	0	6
news	0	0	0	49	3	0	0	0	2	0	4	0	0	0	0	59
com	1	0	0	0	108	1	0	0	0	0	0	2	1	0	0	54
shop	2	0	0	0	5	44	0	0	0	0	0	1	0	0	0	27
how-to	0	0	0	0	0	0	27	0	1	0	0	4	0	0	0	44
recipe	0	0	0	0	1	0	0	7	0	0	0	0	0	0	0	1
blog	2	1	0	0	0	0	0	0	2	0	0	0	0	0	0	11
bio	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0
editorial	0	0	0	1	1	0	0	0	0	0	1	0	0	0	0	9
faq	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	6
edu	0	0	0	0	1	0	0	0	0	0	0	1	6	0	0	4
story	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
interview	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
other	26	0	1	8	30	1	26	0	1	12	1	18	7	0	0	307

Table 6.9: Confusion matrix for open-set genre classification of random web pages

on the LWGC-B corpus and tested on the noise-free LWGC-R corpus which was created by removing the class other in this collection. The result showed 80.68% overall accuracy which is approximately 9% lower than the result of the same model using the designed corpus via 10 fold cross-validation. This drop in accuracy indicates that automatic genre classification on the random web pages is a harder task compared to AGI on the designed corpus. The reason could be that the random web pages include more ambiguous and borderline cases as we noted in Section 6.2.

Second, we experimented with an open-set classification algorithm on the LWGC-R corpus. The unique property of this corpus (i.e. having category other) allowed us to, for the first time, evaluate the performance of an open-set genre classification algorithm on a dataset with noise. The outcome of this experiment indicates that automatic genre classification on the noisy web is a much more challenging task compared to genre classification on the noise-free datasets.

Chapter 7

Conclusion

In today's information era, genre classification of documents plays a very important role in Information Retrieval. As the web grows rapidly, the need for a well-performing automatic genre classification technique is becoming more apparent. Search engines would be more useful if they could automatically distinguish the genre of web pages. With the aim of improving search engines, the past studies of automatic genre classification paid great attention to exploring the performance of various features in supervised genre classification.

However, limitations of the genre-annotated web corpora such as low inter-coder agreement, false correlation between topic and genre labels and their small size, resulted in researchers' doubt on the outcomes of classification models based on these corpora. Therefore, with the aim of overcoming the limitations of these collections, our research journey began with constructing a reliable genre-annotated web corpus which we then used for developing and evaluating automatic genre classification techniques. In this chapter, we conclude the main findings of this thesis. Section 7.1 summarizes our main findings in respect to the construction of the first reliable genre-annotated web corpus via crowd-sourcing as well as developing novel machine learning algorithms for genre classifications. Section 7.2 discusses the potential future work.

7.1 Main Findings

Our major findings include:

- Crowd-sourcing can be a fast and cheap method which yields reliable genre annotation. In Chapter 3, we used Amazon Mechanical Turk which is a crowd-sourcing website to annotate our designed web genre corpus (LWGC-B). This corpus was compiled by a focused search for 15 genre categories. The high inter-coder agreement results of this experiment show that the corpus is annotated reliably. Moreover, in Chapter 6, we employed crowd-sourcing to obtain annotation for a corpus of random web pages (LWGC-R). The result of this experiment also shows high inter-coder agreement. However, the agreement for the LWGC-B corpus was higher than the LWGC-R corpus. The reason is that the LWGC-B corpus consists prototypical instances of our 15 predefined genre categories, whereas, the LWGC-R corpus includes some irregular and borderline instances of the genre classes which can cause disagreement between the annotators. Moreover, the LWGC-B corpus is a balanced collection. In contrast, the LWGC-R corpus is very skewed with no or very few samples for some genre categories. As noted in Section 6.2.3, Skewed categories can bias chance-corrected agreement measures such as π and κ to a low value even when the proportion of annotators agreement is very high [45] and that is the reason for low π for categories story and interview in the LWGC-R corpus.

  Also, it must be noted that the quality of manual annotations depends on the use of precise and consistent guidelines which include the definitions of the categories. Therefore, in our annotation guidelines, we only included categories which are well-defined and well-recognizable and avoided broad and vague categories such as article in KI-04 [104], informative and entertainment in MGC corpus [159] and information and reporting in I-EN-Sample [140].

- The LWGC-B corpus is topically diverse. One of the main short-comings of some of the existing web genre-annotated collections was the spurious correlation between topic and genre categories. The effect of topic on genre classification was discussed extensively in Section 2.8. In order to minimize the impact of topic on genre classification, In construction of the LWGC-B corpus, we collected data from various topics and sources. In Section 3.4, we investigated the topic diversity of the LWGC-B corpus by extracting keywords for each genre categories using log-likelihood statistic [42]. The qualitative analysis of the keywords confirms that the LWGC-B corpus is topically diverse.

- Word unigrams binary representation of web pages is the best feature set for automatic genre classification of web pages. In Chapter 4, we compared the performance of a wide range of lexical, part of speech based and text statistics features in AGI. In general lexical features such as function words, word unigrams and character *n*-grams performed better than POS *n*-gram and text statistics features (e.g. punctuation marks, readability features and HTML features). The advantage of the lexical features is that they can be extracted from the text very easily, whereas, extraction of POS-based features and some of the readability features relies on POS taggers and parsers whose performance varies on different genre categories. Also, these tools are not available for many languages.

  Moreover, for the first time in automatic web genre classification, we compared the performance of classification models on the main text of the web pages as well as the whole text. The results show that genre classification models generally perform better when we use the text that appeared on the web pages in entirety. Boilerplate parts of the web pages which usually contain the genre names (e.g. editorial, news, interview and review) play an important role in genre classification on the web.

- Our novel proposed semi-supervised graph-based genre classification algorithm can improve genre classification on the web. The underlying assumption of this algorithm is that a web page is more likely to be connected to web pages with the same genre category (i.e. homophily property). We hypothesised that homophily exists for genre classes and it can help us to improve genre classification on the web. We used semi-supervised min-cut classification algorithm which is based on homophily assumption. We implement this algorithm in a semi-supervised manner, because we did not have the genre labels for the neighbouring web pages. The results of both binary and multi-class min-cut algorithms show improvement for some genre classes such as news, editorial and blog when the neighbouring web pages are automatically preselected based on cosine similarity measure.

- The genre classification model trained on our designed corpus (LWGC-B) can be used for automatic genre classification on the random web pages corpus (LWGC-R) without noise with reasonable performance (80.68% overall accuracy which is 9% lower than the result of the same model using the LWGC-B corpus via 10 fold cross-validation). However, the result of the open-set genre classification on the LWGC-R corpus with noise is much less encouraging (overall accuracy of 59.32%).

7.2 Direction for Further Research

The following are some of the possible extensions to our work.

7.2.1 Corpus Extensions

The LWGC corpus constructed in this thesis can be extended in several ways as explained in the following paragraphs.

Size Expansion

The corpus which we built in this thesis is significantly bigger than the other existing web genre-annotated corpora in respect to the number of web pages per category (see the corpus statistics in Sections 3.3 and 6.2.4). However, increasing the amount of data can be beneficial for machine learning algorithms and it is recommended in the literature (e.g. [7]). Therefore, one future direction lies in extending this corpus in terms of size which could be done by the focussed search (i.e. the method that we used in Chapter 3) or by annotating random web pages (i.e. the method that we used in Chapter 6). Both of these approaches have advantages and disadvantages. While extending the corpus using random web pages results in creating an unbalanced corpus (as it is shown in Chapter 6), it ensures representativeness of less prototypical examples of genre categories. On the other hand, by employing focussed search approach, we can create a balanced corpus and overcome the problems that a skewed corpus can create for machine learning algorithms. Therefore, we think extending the corpus should be done by employing both of these approaches in order to create a balanced and representative genre corpus.

Another way of extending the corpus in terms of size is to increase the number of genre categories. As noted in Section 2.4, there is no universally agreed set of genre labels. However, as long as the web users can identify a genre category reliably in an annotation task, it can be added to the corpus. In other words, identification of a genre category should be done by the users warrant as noted by Rosso and Haas [127].

Multi-label Annotation

In this thesis, we considered genres as mutually exclusive categories and adopted a single-labelling approach in our annotation tasks (we asked annotators to choose the main genre of the web pages). However, some researchers in the field of AGI have suggested that multi-labelling is a better approach for genre classification (e.g. [34, 70, 130]), because there are web pages which belong to more than one genre category. Therefore, another important future direction lies in allowing multi-labelling genre annotation. One way of designing this approach is to give annotators different levels of reply options for each category because some web pages have several parts which might have different characteristics and each part can comprise a different proportion of the web page. Whether this approach results in a reliable annotation or causes more disagreement is yet to be seen.

7.2.2 Machine Learning Algorithm Enhancements

This section focuses on possible avenues for future work that could lead to improvement in the performance of automatic genre classification.

Graph-based Genre Classification

In Chapter 5, we proposed the min-cut algorithm as a semi-supervised graph-based genre classification technique which employed the neighbouring web pages as unlabelled data. We significantly improved the genre classification result using semi-supervised min-cut algorithm by employing the children of the target web pages as unlabelled data. However, as explained in Section 5.3.2, a web page has different kind of neighbours on the web which are mainly differentiated based on the distance to the target web page as well as the direction of the links. Therefore, a natural extension to this work is to examine the effect of other types of neighbours on genre classification of web pages.

Another possible avenue for future work is to explore other graph-based algorithm such as iterative classification algorithm [90] and Markov Logic Networks [29]. These two algorithms are much more flexible that the min-cut as they can learn from the link pattern and are not restricted to the homophily property. Potential tools for conducting experiments based on this techniques include NetKit-SRL [92] and Alchemy [41]. However, our personal experience with these tools shows that they are not efficient in terms of speed and scalability.

Open-set Genre Classification

An important future direction lies in exploring other open-set genre classification algorithms. In construction of the LWGC-R corpus in Chapter 6, we have a collection of web pages which is annotated as other. In the closed-set classification conducted in Section 6.3, we removed these web pages from the test data, because they do not share common characteristics as they are not part of a single genre category. Then we used the whole LWGC-R corpus including class other to evaluate the performance of an open-set genre classification algorithm. In this experiment, we simulated the web environment by having our 15 predefined genre classes as well as noise web pages. In future work, the LWGC-R corpus can be employed in order to investigate the performance of other open-set genre classification algorithms such as one-class SVM and the random feature sub-spacing ensemble (RFSE) proposed by Pritsos and Stamatatos [116].

Bibliography

[1] Jack Andersen. The concept of genre in information studies. *Annual review of information science and technology*, 42(1):339–367, 2008.

[2] Shlomo Argamon, Moshe Koppel, and Galit Avneri. Routing documents according to style. In *First international workshop on innovative information systems*, pages 85–92. Citeseer, 1998.

[3] Shlomo Argamon, Casey Whitelaw, Paul Chase, Sobhan Raj Hota, Navendu Garg, and Shlomo Levitan. Stylistic text classification using functional lexical features. *Journal of the American Society for Information Science and Technology*, 58(6):802–822, 2007.

[4] R. Artstein and M. Poesio. Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596, 2008.

[5] Noushin Rezapour Asheghi, Katja Markert, and Serge Sharoff. Semi-supervised graph-based genre classification for web pages. In *TextGraphs-9*, page 39, 2014.

[6] Noushin Rezapour Asheghi, Serge Sharoff, and Katja Markert. Designing and evaluating a reliable corpus of web genres via crowd-sourcing. In *(LREC'14)*. European Language Resources Association (ELRA), 2014.

[7] Michele Banko and Eric Brill. Scaling to very very large corpora for natural language disambiguation. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, pages 26–33. Association for Computational Linguistics, 2001.

[8] M. Baroni and S. Bernardini. Bootcat: Bootstrapping corpora and terms from the web. In *Proceedings of LREC*, volume 4. Citeseer, 2004.

[9] M. Baroni, S. Bernardini, A. Ferraresi, and E. Zanchetta. The wacky wide web: a collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*, 43(3):209–226, 2009.

[10] Regina Barzilay and Mirella Lapata. Modeling local coherence: An entity-based approach. *Computational Linguistics*, 34(1):1–34, 2008.

[11] B. Beigman Klebanov and E. Beigman. From annotator agreement to noise models. *Computational Linguistics*, 35(4):495–503, 2009.

[12] E.M. Bennett, R. Alpert, and AC Goldstein. Communications through limited-response questioning. *Public Opinion Quarterly*, 18(3):303, 1954.

[13] L. Bentivogli, M. Federico, G. Moretti, and M. Paul. Getting expert quality from the crowd for machine translation evaluation. *Proceedings of the MT Summmit*, 13:521–528, 2011.

[14] T. Berners-Lee, R. Cailliau, A. Luotonen, H.F. Nielsen, and A. Secret. The world-wide web. *Communications of the ACM*, 37(8):76–82, 1994.

[15] Tim Berners-Lee, Larry Masinter, Mark McCahill, et al. Uniform resource locators (url). *RFC-1738, CERN, Xerox Corporation, University of Minnesota*, 1994.

[16] V. Berninger, Y. Kim, and S. Ross. Building a document genre corpus: a profile of the krys i corpus. 2008.

[17] Douglas Biber. *Variation across speech and writing*. Cambridge University Press, 1991.

[18] Douglas Biber and Susan Conrad. *Register, genre, and style*. Cambridge University Press, 2009.

[19] Douglas Biber, Stig Johansson, Geoffrey Leech, Susan Conrad, Edward Finegan, and Randolph Quirk. *Longman grammar of spoken and written English*, volume 2. MIT Press, 1999.

[20] Avrim Blum and Shuchi Chawla. Learning from labeled and unlabeled data using graph mincuts. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 19–26. Morgan Kaufmann Publishers Inc., 2001.

[21] Elizabeth Sugar Boese and Adele E Howe. Effects of web document evolution on genre classification. In *Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 632–639. ACM, 2005.

[22] Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(11):1222–1239, 2001.

[23] C. Callison-Burch. Fast, cheap, and creative: Evaluating translation quality using amazon's mechanical turk. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*, pages 286–295. Association for Computational Linguistics, 2009.

[24] Karlyn Kohrs Campbell and Kathleen Hall Jamieson. Form and genre in rhetorical criticism: An introduction. *Form and genre: Shaping rhetorical action*, pages 9–32, 1978.

[25] Soumen Chakrabarti, Byron Dom, and Piotr Indyk. Enhanced hypertext categorization using hyperlinks. In *ACM SIGMOD Record*, volume 27, pages 307–318. ACM, 1998.

[26] Guangyu Chen and Ben Choi. Web page genre classification. In *Proceedings of the 2008 ACM symposium on Applied computing*, pages 2353–2357. ACM, 2008.

[27] J. Cohen et al. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46, 1960.

[28] Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, Clifford Stein, et al. *Introduction to algorithms*, volume 2. MIT press Cambridge, 2001.

[29] Robert Crane and Luke McDowell. Investigating markov logic networks for collective classification. In *ICAART (1)*, pages 5–15, 2012.

[30] Mark Craven. Learning to extract symbolic knowledge from the world wide web. In *Proc. of the 15th National Conference on Artificial Intelligence (AAAI-98)*, 1998.

[31] Mark Craven, Seán Slattery, and Kamal Nigam. First-order learning for web mining. In *Machine Learning: ECML-98*, pages 250–255. Springer, 1998.

[32] K. Crowston, B. Kwaśnik, and J. Rubleske. Problems in the use-centered development of a taxonomy of web genres. *Genres on the Web*, pages 69–84, 2011.

[33] K. Crowston and M. Williams. Reproduced and emergent genres of communication on the world wide web. *The Information Society*, 16(3):201–215, 2000.

[34] Kevin Crowston and Barbara H Kwasnik. A framework for creating a facetted classification for genres: Addressing issues of multidimensionality. In *System Sciences, 2004. Proceedings of the 37th Annual Hawaii International Conference on*, pages 9–pp. IEEE, 2004.

[35] M. Davies and J.L. Fleiss. Measuring agreement for multinomial data. *Biometrics*, pages 1047–1051, 1982.

[36] Kristin E Denham and Anne C Lobeck. *Linguistics for Everyone: an Introduction*. CengageBrain. com, 2011.

[37] Johan Dewe, Jussi Karlgren, and Ivan Bretan. Assembling a balanced corpus from the internet. In *11th Nordic Conference of Computational Linguistics, Copenhagen, Denmark*, pages 28–29, 1998.

[38] Thomas G Dietterich. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural computation*, 10(7):1895–1923, 1998.

[39] Andrew Dillon and Barbara A Gushrowski. Genres and the web: Is the personal home page the first uniquely digital genre? *Journal of the American Society for Information Science*, 51(2):202–205, 2000.

[40] Matthias Dimter. *Textklassenkonzepte heutiger Alltagssprache: Kommunikationssituation, Textfunktion und Textinhalt als Kategorien alltagssprachlicher Textklassifikation*, volume 32. Walter de Gruyter, 1981.

[41] Pedro Domingos, Stanley Kok, Hoifung Poon, Matthew Richardson, and Parag Singla. Unifying logical and statistical ai. In *AAAI*, volume 6, pages 2–7, 2006.

[42] T. Dunning. Accurate methods for the statistics of surprise and coincidence. *Computational linguistics*, 19(1):61–74, 1993.

[43] Jesse Egbert and Douglas Biber. Developing a user-based method of register classification. In *Proc. 8th Web as Corpus Workshop*, Lancaster, July 2013.

[44] R.E. Fan, K.W. Chang, C.J. Hsieh, X.R. Wang, and C.J. Lin. Liblinear: A library for large linear classification. *The Journal of Machine Learning Research*, 9:1871–1874, 2008.

[45] Alvan R Feinstein and Domenic V Cicchetti. High agreement but low kappa: I. the problems of two paradoxes. *Journal of clinical epidemiology*, 43(6):543–549, 1990.

[46] S. Feldman, MA Marin, M. Ostendorf, and MR Gupta. Part-of-speech histograms for genre classification of text. In *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, pages 4781–4784. IEEE, 2009.

[47] Jenny Rose Finkel, Trond Grenager, and Christopher Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 363–370. Association for Computational Linguistics, 2005.

[48] Aidan Finn and Nicholas Kushmerick. Learning to classify documents according to genre. *Journal of the American Society for Information Science and Technology*, 57(11):1506–1518, 2006.

[49] J.L. Fleiss. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378, 1971.

[50] L. Freund, C.L.A. Clarke, and E.G. Toms. Towards genre classification for IR in the workplace. In *Proceedings of the 1st international conference on Information interaction in context*, pages 30–36. ACM, 2006.

[51] Kuzman Ganchev and Fernando Pereira. Transductive structured classification through constrained min-cuts. *TextGraphs-2: Graph-Based Algorithms for Natural Language Processing*, page 37, 2007.

[52] E. Giesbrecht and S. Evert. Is part-of-speech tagging a solved task? an evaluation of pos taggers for the German web as corpus. In *Web as Corpus Workshop (WAC5)*, page 27, 2009.

[53] C Lee Giles, Kurt D Bollacker, and Steve Lawrence. Citeseer: An automatic citation indexing system. In *Proceedings of the third ACM conference on Digital libraries*, pages 89–98. ACM, 1998.

[54] R. Girju, A. Badulescu, and D. Moldovan. Automatic discovery of part-whole relations. *Computational Linguistics*, 32(1):83–135, 2006.

[55] Jade Goldstein, Gary M Ciany, and Jaime G Carbonell. Genre identification and goal-focused summarization. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 889–892. ACM, 2007.

[56] Manfred Görlach. *Text types and the history of English*, volume 139. Walter de Gruyter, 2004.

[57] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I.H. Witten. The weka data mining software: an update. *ACM SIGKDD Explorations Newsletter*, 11(1):10–18, 2009.

[58] Douglas M Hawkins. The problem of overfitting. *Journal of chemical information and computer sciences*, 44(1):1–12, 2004.

[59] K. Hofland and S. Johansson. *Word frequencies in British and American English*. Norwegian Computing Centre for the Humanities, 1982.

[60] David Jensen, Jennifer Neville, and Brian Gallagher. Why collective inference improves relational classification. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 593–598. ACM, 2004.

[61] T. Joachims. Text categorization with support vector machines: Learning with many relevant features. *Machine Learning: ECML-98*, pages 137–142, 1998.

[62] M. Kaisser, M. Hearst, and J.B. Lowe. Evidence for varying search results summary lengths. In *Proc. of ACL*, 2008.

[63] I. Kanaris and E. Stamatatos. Learning to recognize webpage genres. *Information Processing & Management*, 45(5):499–512, 2009.

[64] Ioannis Kanaris and Efstathios Stamatatos. Webpage genre identification using variable-length character n-grams. In *Tools with Artificial Intelligence, 2007. ICTAI 2007. 19th IEEE International Conference on*, volume 2, pages 3–10. IEEE, 2007.

[65] Jörg H Kappes, Bjoern Andres, Fred A Hamprecht, Christoph Schnörr, Sebastian Nowozin, Dhruv Batra, Sungwoong Kim, Bernhard X Kausler, Jan Lellmann, Nikos Komodakis, et al. A comparative study of modern inference techniques for discrete energy minimization problems. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[66] J. Karlgren, I. Bretan, J. Dewe, A. Hallberg, and N. Wolkert. Iterative information retrieval using fast clustering and usage-specific genres. In *ERCIM*, page 85. Citeseer.

[67] J. Karlgren and D. Cutting. Recognizing text genres with simple metrics using discriminant analysis. In *Proceedings of the 15th conference on Computational linguistics-Volume 2*, pages 1071–1075. Association for Computational Linguistics, 1994.

[68] Jussi Karlgren. Conventions and mutual expectations. In *Genres on the Web*, pages 33–46. Springer, 2011.

[69] Rohit J Kate, Xiaoqiang Luo, Siddharth Patwardhan, Martin Franz, Radu Florian, Raymond J Mooney, Salim Roukos, and Chris Welty. Learning to predict readability using diverse linguistic features. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 546–554. Association for Computational Linguistics, 2010.

[70] B. Kessler, G. Numberg, and H. Schutze. Automatic detection of text genre. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, pages 32–38. Association for Computational Linguistics, 1997.

[71] A. Kilgarriff. Getting to know your corpus. In *Text, Speech and Dialogue*, pages 3–15. Springer, 2012.

[72] Y. Kim and S. Ross. Examining variations of prominent features in genre classification. In *Hawaii International Conference on System Sciences, Proceedings of the 41st Annual*, pages 132–132. IEEE, 2008.

[73] Yunhyong Kim and Seamus Ross. Searching for ground truth: a stepping stone in automating genre classification. In *Digital Libraries: Research and Development*, pages 248–261. Springer, 2007.

[74] Yunhyong Kim and Seamus Ross. Formulating representative features with respect to genre classification. In *Genres on the Web*, pages 129–147. Springer, 2011.

[75] GR Klare. *The Measurement of Readability*. Ames, IA, Iowa State University Press, 1963.

[76] Dan Klein and Christopher D Manning. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 423–430. Association for Computational Linguistics, 2003.

[77] J.M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5):604–632, 1999.

[78] Jon Kleinberg and Eva Tardos. Approximation algorithms for classification problems with pairwise relationships: Metric labeling and markov random fields. *Journal of the ACM (JACM)*, 49(5):616–639, 2002.

[79] Jon Kleinberg and Éva Tardos. *Algorithm design*. Pearson Education India, 2006.

[80] Moshe Koppel and Jonathan Schler. Exploiting stylistic idiosyncrasies for authorship attribution. In *Proceedings of IJCAI'03 Workshop on Computational Approaches to Style Analysis and Synthesis*, volume 69, page 72. Citeseer, 2003.

[81] K. Krippendorff. Estimating the reliability, systematic error and random error of interval data. *Educational and Psychological Measurement*, 30(1):61, 1970.

[82] Barbara H Kwasnik, You-Lee Chun, Kevin Crowston, J DIgnazio, and Joe Rubleske. Challenges in creating a taxonomy for genres of digital documents. In *Proceedings of the 2006 ISKO Conference*, 2006.

[83] J.R. Landis and G.G. Koch. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159, 1977.

[84] David Lee. Genres, registers, text types, domains, and styles: clarifying the concepts and navigating a path through the BNC jungle. *Language Learning and Technology*, 5(3):37–72, 2001.

[85] Ryan Levering, Michal Cutler, and Lei Yu. Using visual features for fine-grained genre classification of web pages. In *HICSS*, page 131. Citeseer, 2008.

[86] Chul Su Lim, Kong Joo Lee, and Gil Chang Kim. Automatic genre detection of web documents. In *Natural Language Processing–IJCNLP 2004*, pages 310–319. Springer, 2005.

[87] Chul Su Lim, Kong Joo Lee, and Gil Chang Kim. Multiple sets of features for automatic genre classification of web documents. *Information processing & management*, 41(5):1263–1276, 2005.

[88] Christoph Lindemann and Lars Littig. Classification of web sites at super-genre level. In *Genres on the Web*, pages 211–235. Springer, 2011.

[89] D. Litman, J. Hirschberg, and M. Swerts. Characterizing and predicting corrections in spoken dialogue systems. *Computational linguistics*, 32(3):417–438, 2006.

[90] Q. Lu and L. Getoor. Link-based classification using labeled and unlabeled data. *The Continuum from Labeled to Unlabeled Data in Machine Learning & Data Mining*, page 88, 2003.

[91] Sofus A Macskassy and Foster Provost. A simple relational classifier. Technical report, DTIC Document, 2003.

[92] Sofus A Macskassy and Foster Provost. Classification in networked data: A toolkit and a univariate case study. *The Journal of Machine Learning Research*, 8:935–983, 2007.

[93] Mitchell P Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. Building a large annotated corpus of english: The penn treebank. *Computational linguistics*, 19(2):313–330, 1993.

[94] David Martinez and Eneko Agirre. One sense per collocation and genre/topic variations. In *Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics-Volume 13*, pages 207–215. Association for Computational Linguistics, 2000.

[95] Jane E Mason, Michael Shepherd, and Jack Duffy. An n-gram based approach to automatically identifying web page genre. In *System Sciences, 2009. HICSS'09. 42nd Hawaii International Conference on*, pages 1–10. IEEE, 2009.

[96] W. Mason and S. Suri. Conducting behavioral research on amazons mechanical turk. *Behavior Research Methods*, 44(1):1–23, 2012.

[97] G.V. Maverick. Computational analysis of present-day american english. *International Journal of American Linguistics*, 35(1):71–75, 1969.

[98] Andrew Kachites McCallum, Kamal Nigam, Jason Rennie, and Kristie Seymore. Automating the construction of internet portals with machine learning. *Information Retrieval*, 3(2):127–163, 2000.

[99] R. McCreadie, C. Macdonald, and I. Ounis. Crowdsourcing blog track top news judgments at TREC. In *Proceedings of the Workshop on Crowdsourcing for Search and Data Mining (CSDM) at the Fourth ACM International Conference on Web Search and Data Mining (WSDM)*, pages 23–26, 2011.

[100] AM McEnery and MP Oakes. Authorship studies/textual statistics. 2000.

[101] Quinn McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157, 1947.

[102] Alexander Mehler, Matthias Dehmer, and Rüdiger Gleim. Towards logical hypertext structure. In *Innovative Internet Community Systems*, pages 136–150. Springer, 2006.

[103] Alexander Mehler, Serge Sharoff, and Marina Santini, editors. *Genres on the Web: Computational Models and Empirical Studies*. Springer, Berlin/New York, 2010.

[104] S. Meyer zu Eissen and B. Stein. Genre classification of web pages. *KI 2004: Advances in Artificial Intelligence*, pages 256–269, 2004.

[105] Carolyn R Miller. Genre as social action. *Quarterly journal of speech*, 70(2):151–167, 1984.

[106] Lilo Moessner. Genre, text type, style, register: A terminological maze? *European Journal of English Studies*, 5(2):131–138, 2001.

[107] S.M. Mohammad and P.D. Turney. Crowdsourcing a word–emotion association lexicon. *Computational Intelligence*, 2012.

[108] Hyo-Jung Oh, Sung Hyon Myaeng, and Mann-Ho Lee. A practical hypertext catergorization method using links and incrementally available class information. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 264–271. ACM, 2000.

[109] W.J. Orlikowski and J.A. Yates. Genre repertoire: The structuring of communicative practices in organizations. *Administrative science quarterly*, pages 541–574, 1994.

[110] Sarah E Petersen and Mari Ostendorf. A machine learning approach to reading level assessment. *Computer Speech & Language*, 23(1):89–106, 2009.

[111] Philipp Petrenz. Assessing approaches to genre classification. *M. Sc. thesis, School of Informatics, University of Edinburgh*, 2009.

[112] Philipp Petrenz and Bonnie Webber. Stable classification of text genres. *Computational Linguistics*, 37(2):385–393, 2011.

[113] Philipp Petrenz and Bonnie Webber. Robust cross-lingual genre classification through comparable corpora. In *The 5th Workshop on Building and Using Comparable Corpora*, page 1, 2012.

[114] Emily Pitler and Ani Nenkova. Revisiting readability: A unified framework for predicting text quality. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 186–195. Association for Computational Linguistics, 2008.

[115] Jan Pomikálek. *Removing Boilerplate and Duplicate Content from Web Corpora*. PhD thesis, Ph. D. thesis, Masaryk University, 2011.

[116] Dimitrios A Pritsos and Efstathios Stamatatos. Open-set classification for automated genre identification. In *Advances in Information Retrieval*, pages 207–217. Springer, 2013.

[117] Xiaoguang Qi and Brian D Davison. Web page classification: Features and algorithms. *ACM Computing Surveys (CSUR)*, 41(2):12, 2009.

[118] J. Ross Quinlan. Learning logical definitions from relations. *Machine learning*, 5(3):239–266, 1990.

[119] J.R. Quinlan. *C4. 5: programs for machine learning*. Morgan Kaufmann, 1993.

[120] Randolph Quirk, Sidney Greenbaum, Geoffrey Neil Leech, Jan Svartvik, et al. A grammar of contemporary english. 1972.

[121] P. Rayson and R. Garside. Comparing corpora using frequency profiling. In *Proceedings of the workshop on Comparing Corpora*, pages 1–6. Association for Computational Linguistics, 2000.

[122] G. Rehm, M. Santini, A. Mehler, P. Braslavski, R. Gleim, A. Stubbe, S. Symonenko, M. Tavosanis, and V. Vidulin. Towards a reference corpus of web genres for the evaluation of genre identification systems. In *Proc. of the 6th Language Resources and Evaluation Conf.(LREC 2008), Marrakech, Morocco, May*, 2008.

[123] Philip Resnik. Using information content to evaluate semantic similarity in a taxonomy. *arXiv preprint cmp-lg/9511007*, 1995.

[124] Stefan Riezler. On the problem of theoretical terms in empirical computational linguistics. *Computational Linguistics*, 40(1):235–245, 2014.

[125] Azriel Rosenfeld, Robert A Hummel, and Steven W Zucker. Scene labeling by relaxation operations. *Systems, Man and Cybernetics, IEEE Transactions on*, (6):420–433, 1976.

[126] Mark A Rosso. *Using genre to improve web search*. PhD thesis, Citeseer, 2005.

[127] Mark A Rosso and Stephanie W Haas. Identification of web genres by user warrant. In *Genres on the Web*, pages 47–67. Springer, 2011.

[128] Dmitri Roussinov, Kevin Crowston, Mike Nilan, Barbara Kwasnik, Jin Cai, and Xiaoyong Liu. Genre based navigation on the web. In *System Sciences, 2001. Proceedings of the 34th Annual Hawaii International Conference on*, pages 10–pp. IEEE, 2001.

[129] E. Sandhaus. The new york times annotated corpus. *Linguistic Data Consortium, Philadelphia*, 2008.

[130] M. Santini. Zero, single, or multi? genre of web pages through the users' perspective. *Information Processing & Management*, 44(2):702–737, 2008.

[131] Marina Santini. *Automatic identification of genre in web pages*. PhD thesis, University of Brighton, 2007.

[132] Marina Santini. Characterizing genres of web pages: Genre hybridism and individualization. In *System Sciences, 2007. HICSS 2007. 40th Annual Hawaii International Conference on*, pages 71–71. IEEE, 2007.

[133] Marina Santini, Alexander Mehler, and Serge Sharoff. Riding the rough waves of genre on the web. In Alexander Mehler, Serge Sharoff, and Marina Santini, editors, *Genres on the Web: Computational Models and Empirical Studies*. Springer, Berlin/New York, 2010.

[134] Subhajit Dey Sarkar and Saptarsi Goswami. Empirical study on filter based feature selection methods for text classification. *International Journal of Computer Applications*, 81, 2013.

[135] W.A. Scott. Reliability of content analysis: the case of nominal scale coding. *Public Opinion Quarterly*, 1955.

[136] Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. Collective classification in network data. *AI magazine*, 29(3):93, 2008.

[137] S. Sharoff. Creating general-purpose corpora using automated search engine queries. *WaCky*, pages 63–98, 2006.

[138] S. Sharoff. In the garden and in the jungle: Comparing genres in the bnc and internet. *Towards a Reference Corpus of Web Genres*, page 10, 2007.

[139] S. Sharoff, Z. Wu, and K. Markert. The web library of babel: evaluating genre collections. In *Proceedings of the Seventh Conference on International Language Resources and Evaluation*, pages 3063–3070, 2010.

[140] Serge Sharoff. In the garden and in the jungle: Comparing genres in the BNC and Internet. In Alexander Mehler, Serge Sharoff, and Marina Santini, editors, *Genres on the Web: Computational Models and Empirical Studies*, pages 149–166. Springer, Berlin/New York, 2010.

[141] M. Shepherd and C. Watters. The evolution of cybergenres. In *System Sciences, 1998., Proceedings of the Thirty-First Hawaii International Conference on*, volume 2, pages 97–109. IEEE, 1998.

[142] M. Shepherd, C. Watters, and A. Kennedy. Cybergenre: automatic identification of home pages on the web. *Journal of Web Engineering*, 3(3-4):236–251, 2004.

[143] Seán Slattery and Tom M Mitchell. Discovering test set regularities in relational domains. In *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 895–902. Morgan Kaufmann Publishers Inc., 2000.

[144] R. Snow, B. O'Connor, D. Jurafsky, and A.Y. Ng. Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 254–263. Association for Computational Linguistics, 2008.

[145] B. Snyder and M. Palmer. The english all-words task. In *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, pages 41–43, 2004.

[146] Efstathios Stamatatos, Nikos Fakotakis, and George Kokkinakis. Text genre detection using common word frequencies. In *Proceedings of the 18th conference on Computational linguistics-Volume 2*, pages 808–814. Association for Computational Linguistics, 2000.

[147] Johannes M. Steger and Egon W. Stemle. KrdWrd – architecture for unified processing of web content. 2009.

[148] B. Stein and S.M. Eissen. Retrieval models for genre classification. *Scandinavian Journal of Information Systems*, 20(1):3, 2008.

[149] B. Stein, S.M. Eissen, and N. Lipka. Web genre analysis: Use cases, retrieval models, and implementation issues. *Genres on the Web*, pages 167–189, 2011.

[150] Jade Goldstein Stewart. *Genre oriented summarization.* PhD thesis, Google, 2009.

[151] A. Stubbe and C. Ringlstetter. Recognizing genres. *Proc. Towards a Reference Corpus of Web Genres*, 2007.

[152] Andrea Stubbe, Christoph Ringlstetter, and Klaus U Schulz. Genre as noise: Noise in genre. *International Journal of Document Analysis and Recognition (IJDAR)*, 10(3-4):199–209, 2007.

[153] J. Svartvik. *The London-Lund corpus of spoken English: Description and research.* Number 82. Lund University Press, 1990.

[154] John Swales. *Genre analysis: English in academic and research settings.* Cambridge University Press, 1990.

[155] Ben Taskar, Pieter Abbeel, and Daphne Koller. Discriminative probabilistic models for relational data. In *Proceedings of the Eighteenth conference on Uncertainty in artificial intelligence*, pages 485–492. Morgan Kaufmann Publishers Inc., 2002.

[156] Yla R Tausczik and James W Pennebaker. The psychological meaning of words: Liwc and computerized text analysis methods. *Journal of Language and Social Psychology*, 29(1):24–54, 2010.

[157] Kristina Toutanova, Dan Klein, Christopher D Manning, and Yoram Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 173–180. Association for Computational Linguistics, 2003.

[158] Grigorios Tsoumakas and Ioannis Katakis. Multi-label classification: An overview. *International Journal of Data Warehousing and Mining (IJDWM)*, 3(3):1–13, 2007.

[159] V. Vidulin, M. Luštrek, and M. Gams. Using genres to improve search engines. *Towards Genre-Enabled Search Engines: The Impact of Natural Language Processing*, 4:45, 2007.

[160] Vedrana Vidulin, Mitja Lustrek, and Matjaz Gams. Multi-label approaches to web genre identification. *JLCL*, 24(1):97–114, 2009.

[161] R. Vieira and M. Poesio. An empirically based system for processing definite descriptions. *Computational Linguistics*, 26(4):539–593, 2000.

[162] B. Webber. Genre distinctions for discourse in the penn treebank. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 674–682. Association for Computational Linguistics, 2009.

[163] Z. Wu, K. Markert, and S. Sharoff. Fine-grained genre classification using structural learning algorithms. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 749–759. Association for Computational Linguistics, 2010.

[164] J.A. Yates, W.J. Orlikowski, and J. Rennecker. Collaborative genres for collaboration: Genre systems in digital media. In *System Sciences, 1997, Proceedings of the Thirtieth Hawaii International Conference on*, volume 6, pages 50–59. IEEE, 1997.

[165] VA Yatsko, MS Starikov, and AV Butakov. Automatic genre recognition and adaptive text summarization. *Automatic Documentation and Mathematical Linguistics*, 44(3):111–120, 2010.

[166] Ying Zhao and Justin Zobel. Effective and scalable authorship attribution using function words. In *Information Retrieval Technology*, pages 174–189. Springer, 2005.

[167] Xiaojin Zhu, John Lafferty, and Ronald Rosenfeld. *Semi-supervised learning with graphs*. PhD thesis, Carnegie Mellon University, Language Technologies Institute, School of Computer Science, 2005.

[168] George Kingsley Zipf. Human behavior and the principle of least effort, 1949.

Appendix A

Genre Classification Schemes for Existing Web Genre Corpora

This appendix contains the genre classification schemes for existing web genre collections which are described in Section 2.5. By Comparing the Tables in this appendix, we see that these collections are different in terms of the set of genre labels used in their genre classification schemes.

1. Personal Blogs	5. Listings
2. Eshop	6. Personal Home Page
3. FAQs	7. Search Pages
4. Online Newspaper Front Pages	

Table A.1: The SANTINIS [131] corpus genre classification scheme

1. Article	5. Link Collection
2. Discussion	6. Non-personal Home Page
3. Download	7. Personal Home Page
4. Help	8. Shop

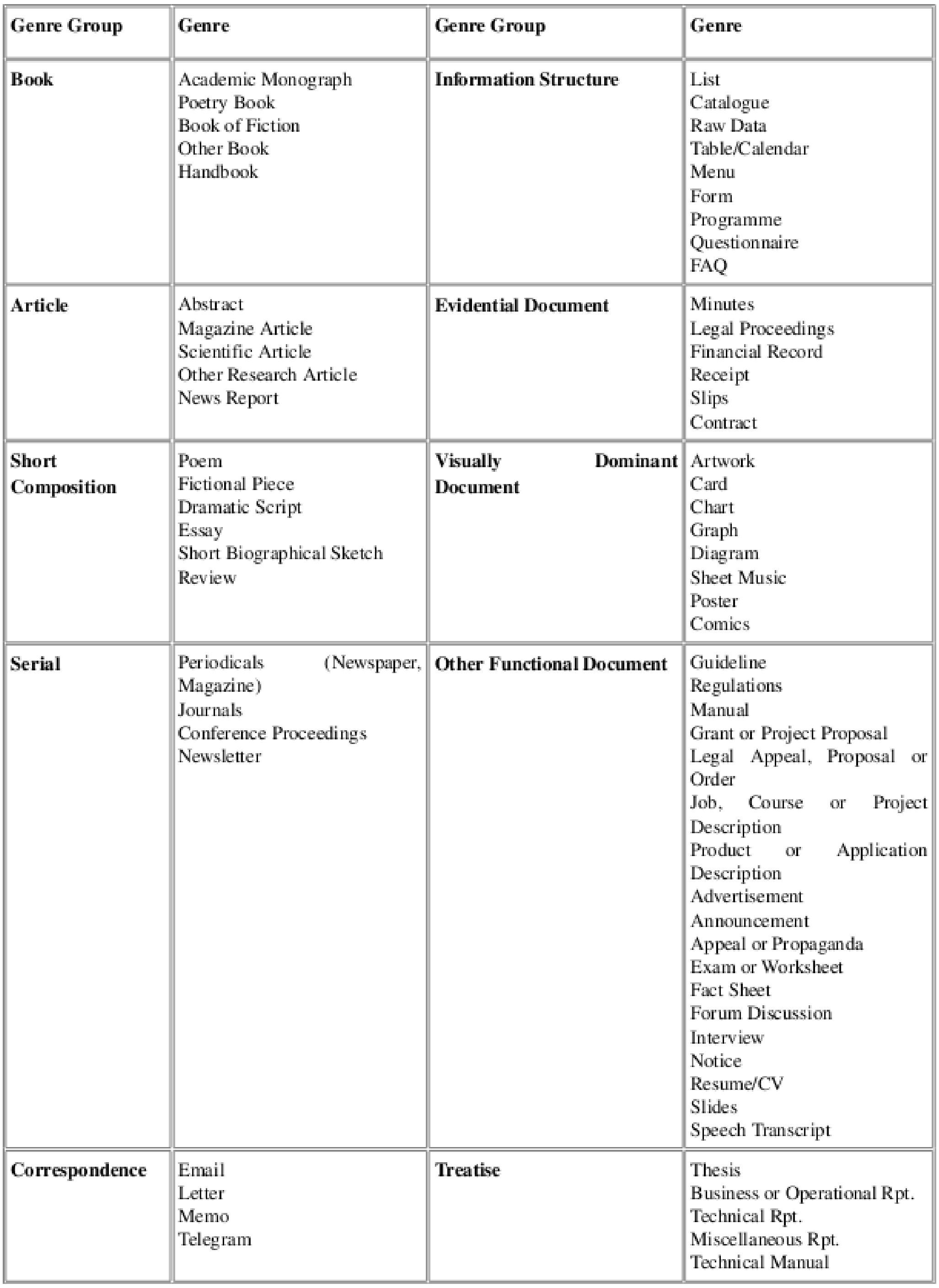
Table A.2: The KI-04 [104] corpus genre classification scheme

i. Journalism	iv. Documentation
1. Commentary	21. Law
2. Review	22. Official Report
3. Portrait	23. Protocol
4. Marginal Note	
5. Interview	v. Dictionary
6. News	24. Person
7. Feature Story	25. Catalog
8. Reportage	26. Resources
	27. Timeline
ii. Literature	
9. Poem	vi. Communcation
10. Prose	28. Mail, Talk
11. Drama	29. Forum, Guestbook
	30. Blog
iii.Information	31. Form
12. Science Report	
13. Explanation	vii. Nothing
14. Receipt	32. Nothing
15. FAQ	
16. Lexicon, Word List	
17. Bilingual Dictionary	
18. Presentation	
19. Statistics	
20. Code	

Table A.3: The HGC [151] corpus genre classification scheme

1. Personal	11. Index
2. Informative	12. Gateway
3. Journalistic	13. Community
4. Commercial/promotional	14. Content Delivery
5. Shopping	15. User input
6. Official	16. Entertainment
7. Scientific	17. Adult
8. Prose fiction	18. Children's
9. Poetry	19. Blog
10. FAQs	20. Error message

Table A.4: The MGC [159] corpus genre classification scheme



Genre Group	Genre	Genre Group	Genre
Book	Academic Monograph Poetry Book Book of Fiction Other Book Handbook	**Information Structure**	List Catalogue Raw Data Table/Calendar Menu Form Programme Questionnaire FAQ
Article	Abstract Magazine Article Scientific Article Other Research Article News Report	**Evidential Document**	Minutes Legal Proceedings Financial Record Receipt Slips Contract
Short Composition	Poem Fictional Piece Dramatic Script Essay Short Biographical Sketch Review	**Visually Dominant Document**	Artwork Card Chart Graph Diagram Sheet Music Poster Comics
Serial	Periodicals (Newspaper, Magazine) Journals Conference Proceedings Newsletter	**Other Functional Document**	Guideline Regulations Manual Grant or Project Proposal Legal Appeal, Proposal or Order Job, Course or Project Description Product or Application Description Advertisement Announcement Appeal or Propaganda Exam or Worksheet Fact Sheet Forum Discussion Interview Notice Resume/CV Slides Speech Transcript
Correspondence	Email Letter Memo Telegram	**Treatise**	Thesis Business or Operational Rpt. Technical Rpt. Miscellaneous Rpt. Technical Manual

Table A.5: The KRYS I [16] corpus genre classification scheme

- Narrative
  - News report/blog
  - Sports report
  - Personal/diary blog
  - Historical article
  - Short story
  - Novel
  - Biographical story/history
  - Joke
  - Magazine article
  - Memoir
  - Obituary
  - Other factual narrative
  - Other fictional narrative
  - Other personal narrative
  - Travel blog
- Opinion
  - Opinion blog
  - Review
  - Advice
  - Religious blog/sermon
  - Self-help
  - Advertisement
  - Letter to the editor
- Description
  - Description of a thing
  - Description of a person
  - Research article
  - Abstract
  - Legal terms and conditions
  - FAQ about information
  - Encyclopedia article
  - Informational blog
  - Course materials
  - Technical report
  - Other
- Discussion
  - Question/answer forum
  - Other forum
  - Other discussion
  - Reader/viewer responses
- Lyrical
  - Song lyrics
  - Other
  - Poem
  - Prayer
- How-to/Instructional
  - How-to
  - Technical support
  - Recipe
  - Instructions
  - FAQ about how to do something
  - Other
- Informational Persuasion
  - Description with intent to sell
  - Persuasive article or essay
  - Editorial
  - Other
- Spoken
  - Interview
  - Formal speech
  - Transcript of video/audio
  - Other
  - TV/movie script

Table A.6: Genre classification scheme for the corpus presented in [43]

- “About Us” Page
  - About a Companys Services
  - About a Country/State/City
  - About a Program/Institution
  - Mission Statement
  - Other ”About Us” Page
- Abstract
- Account Creation Page
- Advertisement
- Announcement
  - Event Announcement
  - Press Release
  - Other Announcement
- Article
  - Journal Article
  - Magazine Article
  - Article Excerpt
  - Other Article
- Syllabus
- Bibliography
  - Annotated Bibliography
  - Bibliographic Citation
  - Other Bibliography
- Biography
  - Extended Biography
  - Brief Biography
  - Featured Profile
  - Biographical Timeline
  - Other Biography
- Brief
  - Report Brief
  - Registry Brief
  - Policy Brief
- Brochure
- Resume/Curriculum Vitae
  - Cv Excerpt
- Shopping
  - Catalog
  - Testimonial
  - Product/Service Description Page
  - Product/Service Features Page
  - Product/Service Specification Page
  - Product/Service Promotion
  - Product/Service Purchase Page
- Contact Page
- “Contact Us” Page
- Contract
- Proclamation
- Data
  - Specifications
  - Material Data Sheets
- Glossary
- Description
  - Course Description
- Interview
  - Interview Transcript
- List
  - List of People
  - List of Companies
  - List of Places
  - List of Services
  - List of Products
  - List of Projects
- Login Page
- Manual
  - Coding Manual
  - User Manual Page
- Minutes
- Legal Decision
- Outline
- Portal
  - News Portal
  - Research Portal
- Blog
  - Blog Diary
  - Blog News
  - Blog Political
  - Other Blog
- Position Statement
- Report
  - Research Report
  - Government Report
  - Report Chapter
  - Other Report
- Rules and Regulations
  - Policy Statements
  - Accreditation Guidelines
  - License Requirements
  - Rules For Games
  - Draft Bills
  - State and Federal Laws
  - Construction Requirements
  - Other Rules and Regulations
- Search Page
  - Archives Search
  - Search For Statistics
  - Search For Products/Services
  - Search For Recipes
- Search Results
- Site Map
- Source Code
  - Source Code Library
- Table of Contents
- White Paper
- Gallery
- Quotation Page
- Book
  - Book Excerpt
  - Index to Videos
  - Index to Tutorials
  - Index to News Stories
  - Index to Blogs
  - Index to Reviews
  - Index to Past Issues
  - Index to Recipes
  - Index to Projects
  - Index to Articles
  - Index to Resources
  - Index to Biographies
  - Index to Lesson Plans
  - Index to Statistics
  - Index to Reports
  - Index to Laws and Policies
  - Index to Agencies and Offices
  - Index to Press Releases
  - Index to Images
  - Index to Newsletters
  - Other Index
- Standards
- Instructions
- Legal Agreement
- Lesson Resource
- Letter
- License
- Statistics
- Tutorial
- Locator
  - Office Locator
- Map
- Memo
- Menu
- Discussion Board
- Simulation
- Newsletter
- Roster
- Notification
- Overview
- Plan
  - Lesson Plan
  - Test Plan
  - Strategic Plan
  - Long-Term Plan
- Presentation
  - Slideshow Page
- Redirect Page
- Reference
  - Fact Sheet
  - Other Reference
- Schedule
  - Program Schedule
- Definition
- Sponsored Links

- Directory
  - Directory of Places/Institutions
  - Directory of Lesson Plans
  - Directory of Blogs
  - Directory of White Papers
  - Directory of Books
  - Directory of Phone Numbers
  - Directory of Articles
  - Directory of Resources/Links
  - Directory of News Stories
  - Directory of Online Stores
  - Directory of Companies
  - Directory of Laws and Policies
  - Directory of Software
  - Directory of Reports
  - Directory of Standards
- Dissertation
- Case Study
- Encyclopedia Entry
  - Wikipedia Page
  - Other Encyclopedia Entry
- Flyer
- Discography
- Form
  - Poll
  - Quiz
  - Questionnaire
  - Survey
  - Comment Entry
  - Registration Form
  - Other Form
- Frequently Asked Questions
- Guide
  - City Guide
  - Shopping Guide
  - Travel Guide
  - User Guide
  - Programming Guide
  - Guide to Resources
  - Event Guide
  - Other Guide
- Help Page
- Playbill

  - Book Chapter
  - Preface
  - Book Summary
  - Other Book
- Calendar
  - Calendar of Events
- Column
- Digest
- Editorial
- Email Page
- Legal Proposal
- Video
- 404 Not Found Page
  - Stop Page
- Essay
  - Personal Essay
- News
  - Feature
  - News Story
  - News Stories
  - News Blurb/Excerpt
- Homepage
  - Professional Homepage
  - Magazine Homepage
  - Organizational Homepage
  - Shopping Homepage
  - Company Homepage
  - Project Homepage
  - Personal Homepage
  - Program Homepage
  - Community Homepage
  - Product Homepage
  - Group Homepage
  - Government Homepage
  - Other Homepage
- How-To
  - Recipe
  - Other How-To
- Index
  - Index to Books/Essays
  - Index to Standards
  - Index to Products
  - Index to Tests

- Legal Opinion
- Speech
- Review
  - Book/Textbook Review
  - Game Review
  - Article Review
  - Report Review
- Notes
  - Class Notes
  - Meeting Notes
  - Working Notes
- Agenda
- Obituary
- Proposal
  - Request For Proposal (RFP)
  - Other Proposal
- Splash Page
- Podcast
- Best Practices
- Bulletin
- Calculator
- Profile
- 403 Company Profile
- Verification Page
  - Age Verification Page
- Story
  - Promotional Story
- Summary
  - Change Summary
  - Executive Summary
  - Legal Summary
- Synopsis
- History
  - Timeline
  - Other History
- Terms and Conditons
  - Disclaimer
- Image Page
- Suspended Page
- Under Construction Page
- User Preferences Page
- Game
- Petition
- Question-and-Answer Page

Table A.7: The Syracuse [32] corpus genre classification scheme

Appendix B

Lexical Features in Detail

B.1 Genre Names

Table B.1 and Table B.2 show the list of genre names and genre keywords, which have been used in this thesis, respectively.

news	biographies	reviews
editorial	forum	shop
editorials	forums	shops
opinion	blog	faqs
recipe	blogs	faq
recipes	diary	frequently asked questions
instruction	diaries	company homepage
instructions	story	school homepage
how to	stories	university homepage
bio	interview	college homepage
bios	interviews	personal homepage
biography	review	

Table B.1: Genre names

B.2 Common Words

Table B.3 shows 50 most common words in the BNC which were used by Stamatatos et al. [146] for the first time in automatic genre classification.

news	question	recipes	forum	interview
editorial	company	instruction	forums	review
editorials	school	instructions	blog	shop
opinion	university	how	blogs	interviews
faq	faqs	college	bio	diary
frequently	personal	bios	diaries	shops
asked	homepage	biography	story	reviews
questions	recipe	biographies	stories	

Table B.2: Genre keywords

the	to	with	that	are	but	or	been	her	as
of	is	he	by	not	had	an	has	n’t	if
and	was	be	at	his	which	were	have	there	who
a	it	on	you	this	she	we	will	can	what
in	for	i	’s	from	they	their	would	all	said

Table B.3: 50 most common words

B.3 Function Words

Table B.4 shows the list of British National Corpus (BNC) part of speech tags [1] which are used for extracting function words. The list of function words extracted from BNC using these tags, contains 927 words. I Also added hundred most frequent general adverbs (AV0) to this list to construct the final list of function words.

POS tag	Description
AT0	Article (e.g. the, a, an, no)
AVP	Adverb particle (e.g. up, off, out)
AVQ	Wh-adverb (e.g. when, where, how, why, wherever)
CJC	Coordinating conjunction (e.g. and, or, but)
CJS	Subordinating conjunction (e.g. although, when)
CJT	The subordinating conjunction that
DPS	Possessive determiner-pronoun (e.g. your, their, his)
DT0	General determiner-pronoun: i.e. a determiner-pronoun which is not a DTQ or an AT0.
DTQ	Wh-determiner-pronoun (e.g. which, what, whose, whichever)
EX0	Existential there, i.e. there occurring in the there is ... or there are ... construction
ITJ	Interjection or other isolate (e.g. oh, yes, mhm, wow)

[1] http://ucrel.lancs.ac.uk/bnc2/bnc2guide.htm

PNI	Indefinite pronoun (e.g. none, everything, one [as pronoun], nobody)
PNP	Personal pronoun (e.g. I, you, them, ours)
PNQ	Wh-pronoun (e.g. who, whoever, whom)
PNX	Reflexive pronoun (e.g. myself, yourself, itself, ourselves)
POS	The possessive or genitive marker ’s or ’
PRF	The preposition of
PRP	Preposition (except for of) (e.g. about, at, in, on, on behalf of, with)
TO0	Infinitive marker to
VBB	The present tense forms of the verb BE, except for is, ’s: i.e. am, are, ’m, ’re and be [subjunctive or imperative]
VBD	The past tense forms of the verb BE: was and were
VBG	The -ing form of the verb BE: being
VBI	The infinitive form of the verb BE: be
VBN	The past participle form of the verb BE: been
VBZ	The -s form of the verb BE: is, ’s
VDB	The finite base form of the verb BE: do
VDD	The past tense form of the verb DO: did
VDG	The -ing form of the verb DO: doing
VDI	The infinitive form of the verb DO: do
VDN	The past participle form of the verb DO: done
VDZ	The -s form of the verb DO: does, ’s
VHB	The finite base form of the verb HAVE: have, ’ve
VHD	The past tense form of the verb HAVE: had, ’d
VHG	The -ing form of the verb HAVE: having
VHI	The infinitive form of the verb HAVE: have
VHN	The past participle form of the verb HAVE: had
VHZ	The -s form of the verb HAVE: has, ’s
VM0	Modal auxiliary verb (e.g. will, would, can, could, ’ll, ’d)
XX0	The negative particle not or n’t

Table B.4: The list of British National Corpus part of speech tags which we used for extracting function words.

Appendix C

HTML tags

Table C.1 lists the HTML tags that I used in my experiments. I grouped these tags into four categories: sections and style, formatting, visual features and programming. The description for individual tags are from w3schools [1] website.

Sections / Style	Description
<title>	Defines a title for the document
<body>	Defines the document's body
<h1> to <h6>	Defines HTML headings
<p>	Defines a paragraph
<br>	Inserts a single line break
<hr>	Defines a thematic change in the content
<style>	Defines style information for a document
<div>	Defines a section in a document
<span>	Defines a section in a document
<header>	Defines a header for a document or section
<footer>	Defines a footer for a document or section
<section>	Defines a section in a document
<article>	Defines an article
<aside>	Defines content aside from the page content
<details>	Defines additional details that the user can view or hide
<dialog>	Defines a dialog box or window
<summary>	Defines a visible heading for a <details> element
<!–...–>	Defines a comment
Formatting	

[1] http://www.w3schools.com/tags/ref_byfunc.asp

<acronym>	Defines an acronym
<abbr>	Defines an abbreviation
<address>	Defines contact information for the author/owner of a document/article
<b>	Defines bold text
<big>	Defines big text
<blockquote>	Defines a section that is quoted from another source
<center>	Defines centered text
<cite>	Defines the title of a work
<code>	Defines a piece of computer code
<dfn>	Defines a definition term
<em>	Defines emphasized text
<font>	Defines font, color, and size for text
<mark>	Defines marked/highlighted text
<meter>	Defines a scalar measurement within a known range
<pre>	Defines preformatted text
<q>	Defines a short quotation
<small>	Defines smaller text
<strike>	Defines strikethrough text
<strong>	Defines important text
<sub>	Defines subscripted text
<sup>	Defines superscripted text
<time>	Defines a date/time
<u>	Defines text that should be stylistically different from normal text
<var>	Defines a variable
<wbr>	Defines a possible line-break

Visual features

<form>	Defines an HTML form for user input
<input>	Defines an input control
<button>	Defines a clickable button
<img>	Defines an image
<canvas>	Used to draw graphics, on the fly, via scripting
<ul>	Defines an unordered list
<ol>	Defines an ordered list
<li>	Defines a list item
<dir>	Defines a directory list

<dl>	Defines a description list
<menu>	Defines a list/menu of commands
<table>	Defines a table
<caption>	Defines a table caption
<th>	Defines a header cell in a table
<tr>	Defines a row in a table
<td>	Defines a cell in a table
<thead>	Groups the header content in a table
<tbody>	Groups the body content in a table
<tfoot>	Groups the footer content in a table
<col>	Specifies column properties for each column within a <colgroup> element
<colgroup>	Specifies a group of one or more columns in a table for formatting
<audio>	Defines sound content
<source>	Defines multiple media resources for media elements (<video> and <audio>)
<track>	Defines text tracks for media elements (<video> and <audio>)
<video>	Defines a video or movie
<a>	Defines a hyperlink
<link>	Defines the relationship between a document and an external resource (most used to link to style sheets)
Programming	
<script>	Defines a client-side script
<noscript>	Defines an alternate content for users that do not support client-side scripts
<applet>	Defines an embedded applet
<embed>	Defines a container for an external (non-HTML) application
<object>	Defines an embedded object
<param>	Defines a parameter for an object

Table C.1: HTML tags and their descriptions