ECS763U/P Natural Language Processing

Julian Hough

Week 1: Introduction

Part 1: Applications and disciplines

Julian Hough (Module Organizer)

- j.hough@qmul.ac.uk. Office hours Friday 3-4pm.
- Main research interests in NLP: dialogue systems and dialogue analysis, and human-robot dialogue.
- For content queries use the **QMPlus forum** to ask questions rather than email (a problem shared is a problem halved!).
- Sign up to our NLP seminar!:
    - https://www.lists.qmul.ac.uk/sympa/info/nlp-seminar (click Subscribe on the menu on the bottom-left side)

OUTLINE

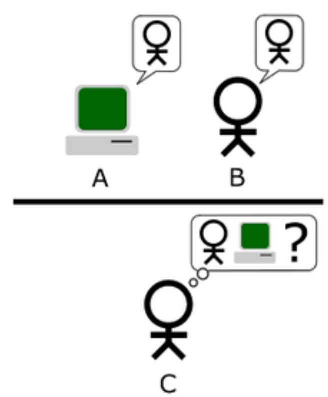
1) What is NLP and where is it used?
2) Managing big data: classification and extraction
3) Intro to statistical and probabilistic methods
4) Intro to dialogue and its challenges

What is Natural Language Processing?

- **Natural Language Processing (or Computational Linguistics) i**s the automatic processing of human language data for some purpose.

What is Natural Language Processing?

- **BIG PICTURE 1:** We really want to build machines that **understand** human language in a human way, and **produce/generate** human language in a human way.
- Alan Turing (1950) originally posed the **Turing Test** as being key to solving artificial intelligence (AI).
- Could a machine 'fool' someone into thinking they're talking to a human? That system will have solved AI**.**



What is Natural Language Processing?

- **<u>BIG PICTURE 2:</u>** We want **tools** that allow us to do tasks more effectively.
- This technology might assist you with **organizing** huge amounts of text information, accessing parts of it, and extracting data from it.
- It can help you **create** your own text data: e.g. spelling and style correction.
- It can **help** those who need it: text-to-speech from screens for the blind; speech-to-text for those with manual problems.

What is Natural Language Processing?

- **Why** is it worth studying?
  - Huge number of applications to help humans do useful tasks.
  - Consequently has huge commercial and social value.
  - Theoretical interest as it shines a light on how human beings use language to communicate.

- As a **field** it's at the intersection of:
  - Computer Science
  - Data Science
  - AI / Machine Learning (More recently Deep Learning)
  - Linguistics / Cognitive Science

Levels of analysis (small to large)

- **Phonemes/sounds (Speech recognition, prosody)**
- **Words (can be broken down into morphemes)**
- **Phrases**
- **Sentences/Turns**
- **Texts/Dialogues**

- On this module we cover approximately the level of the word upwards as an increment of analysis (not so much about vocal signal).

Why is NLP difficult and interesting? Because human language is…

- **Ambiguous (can mean several things at once)** (unlike programming languages)
- **Not always explicit and depends on context**. You leave out "code"- the listener/reader fills in the gaps!
  - **Context** includes real-world knowledge. Do words 'mean' anything without reference to real things/situations?
- **Rich** in its ability to express lots of things.
- **Creative**- you can always create a new word/phrase!

Applications: main areas

- Machine Translation (since the 1950s)
- Search (Google)
- **Managing BIG data**:
  - Analysing social media for advertising e.g. **sentiment analysis** for products.
  - Finance: buy/sell decisions based on social media texts. Health: Which hospitals are good?
- **Dialogue systems/Chatbots**:
  - Personal assistants (Amazon's Alexa, Apple's Siri).
  - Human-robot interaction with speech.
  - Automating customer service.

Applications (simple to complex)

- Keyword search
- Spell-checking/auto-complete
- Extracting information from websites: product, price, company names
- Summarization of texts
- **Classification: sentiment classification (positive or negative), difficulty of reading level of text**
- **Machine translation**
- **Question Answering**
- **Conversation Analytics**
- **Dialogue Systems (spoken and typed interfaces)**

Applications: Machine translation

- The earliest form of NLP. Started in the 50s.
- Now widely used with large scale statistical methods.
- Google translate is pretty impressive, with a huge number of language pairs.
- “The Google Translate app supports more than 100 languages and can translate 37 languages via photo, 32 via voice in "conversation mode", and 27 via real-time video in "augmented reality mode".”

Applications: Dialogue systems

Applications: Dialogue systems

- The advent of mobile phones has been a blessing to NLP for commercial systems.

- Gave rise to Siri, then Google Assistant, Cortana. Question Answering and information retrieval through voice.

- Then finally it has adopted into people's homes- Alexa, Google Home.

Applications: Dialogue systems

- Chatbots (text-based)
  - Personal assistants
  - Online helpline/FAQ answering
  - Helps reduce human labour
  - Google DialogFlow is an easy open source toolkit to build chatbots **(Unassessed lab on this)**

- Spoken dialogue systems (speech-based)
  - Artificial call centre employees
  - In robots/cars
  - Can be artificial companions and again, helps reduce human labour

Applications: Managing big (textual) data

- **CLASSIFY** text so as to identify relevant content / quickly assess this content
  - E.g., **SENTIMENT ANALYSIS**

- **EXTRACT** structured information from unstructured textual data

- **SUMMARIZE** text

OUTLINE

1) What is NLP and where is it used?
2) Managing big data: classification and extraction
3) Intro to statistical and probabilistic methods
4) Intro to dialogue and its challenges

Sentiment Analysis

1. Id: Abc123 on 5-1-2008 “I bought an iPhone a few days ago. It is such a nice phone. The touch screen is really cool. The voice quality is clear too.

2. It is much better than my old Blackberry, which was a terrible phone and so difficult to type with its tiny keys. However, my mother was mad with me as I did not tell her before I bought the phone. She also thought the phone was too expensive, …”

Sentiment Analysis

POSITIVE about IPhone ☺️

1. Id: Abc123 on 5-1-2008 “I bought an iPhone a few days ago. It is such a nice phone. The touch screen is really cool. The voice quality is clear too.

2. It is much better than my old Blackberry, which was a terrible phone and so difficult to type with its tiny keys. However, my mother was mad with me as I did not tell her before I bought the phone. She also thought the phone was too expensive, …”

Sentiment Analysis

POSITIVE about IPhone ☺️

1. Id: Abc123 on 5-1-2008 “I bought an iPhone a few days ago. It is such a nice phone. The touch screen is really cool. The voice quality is clear too.

2. It is much better than my old Blackberry, which was a terrible phone and so difficult to type with its tiny keys. However, my mother was mad with me as I did not tell her before I bought the phone. She also thought the phone was too expensive, …”

POSITIVE about IPhone ☺️

NEGATIVE about Blackberry ☹️

Sentiment Analysis

- A typical NLP task
- You have a large amount of data available to you (a **corpus**). E.g. collection of tweets or comments.
- You need to build something to make the automatic decision about the tweet:
  - **Positive** ☺️ **vs Negative** ☹️
    - *I’m really happy!*
    - *I’m having a terrible day*
    - *Oh man this is so great <3*
    - *I just can’t believe it*
- **How could we go about this?**

Sentiment Analysis 1: Dictionaries

- We could build dictionaries:
  - List of “positive” words
  - List of “negative” words

- Problem with ambiguity- is this positive or negative?:

```
i love @justinbieber #sarcasm
```

- We might need a more data-driven approach…

Sentiment Analysis 2: Data-Driven Classification

- We could **learn** the dictionaries of ‘positive’ and ‘negative’ words from:
  - List of “positive” examples
  - List of “negative” examples

- Learn a classifier based on observed words … and combinations thereof

- We can use maths: **statistics** and **geometry**

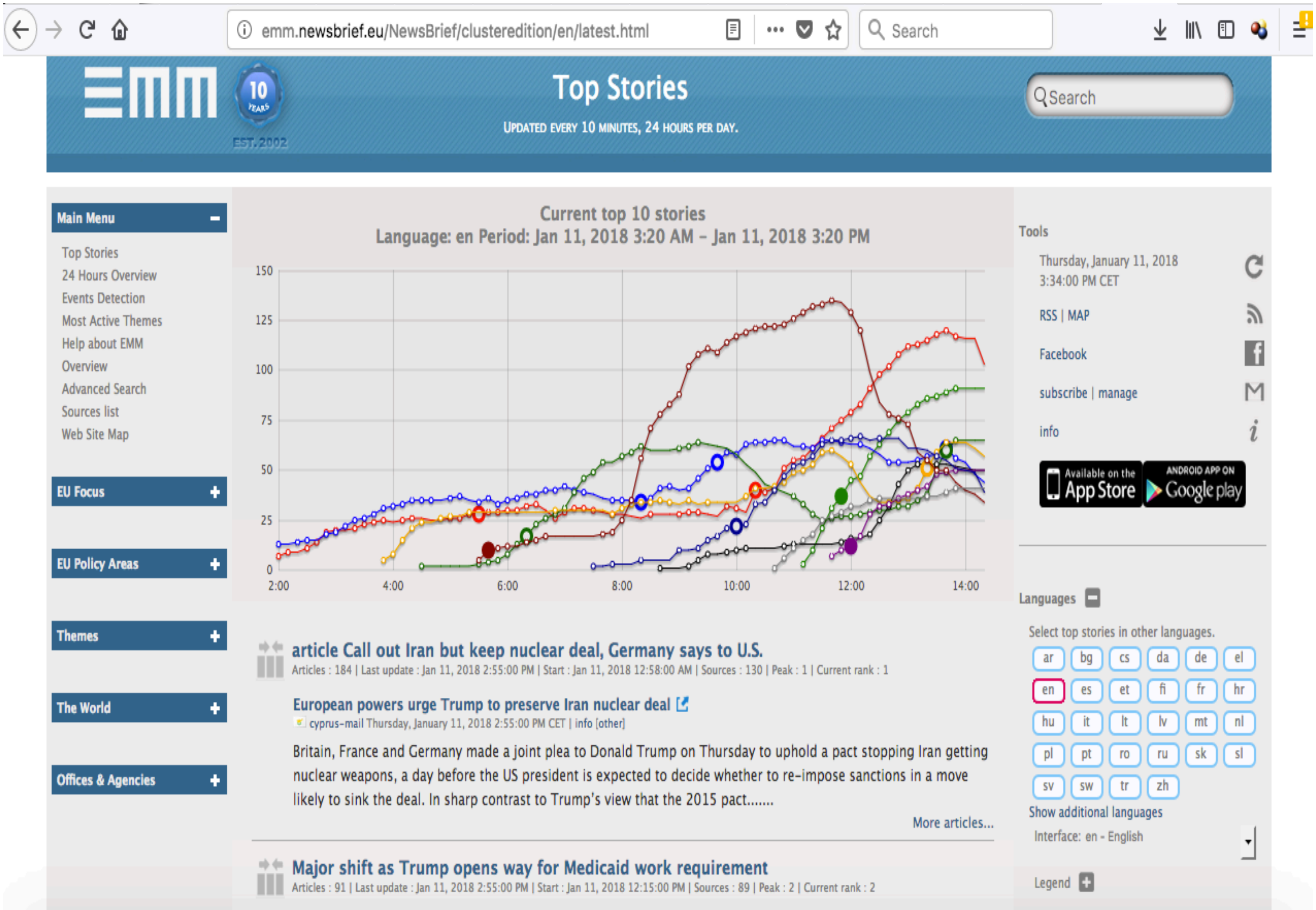
Information Extraction

Summarization

- **Summarization** is the production of a summary either from a single source (single-document summarization) or from a collection of articles (multi-document summarization)

- Main approaches are:
  - **Extractive**: Select key sentences/phrases for summary.
  - **Abstractive**: Re-generate a summary based on the meaning of the text.

Clustering and summarization



OUTLINE

1) What is NLP and where is it used?
2) Managing big data: classification and extraction
3) Intro to statistical and probabilistic methods
4) Intro to dialogue and its challenges

Mathematical foundations

- The overwhelmingly most successful methods are **statistical and probabilistic** in nature.
- They may have greater to lesser degrees of ‘linguistic’ information like phrase structure, parts of speech etc.
- Currently the trend is to have less and less linguists involved:

  “Every time I fire a linguist, the performance of the speech recognizer goes up.”

  Fred Jelinek, leading pioneer of modern day automatic speech recognition (ASR)

Mathematical foundations

- However, there's still a use for the old non-statistical insights.
- Linguists are still the only ones to point out difficult examples with classical **puzzles of meaning**:

  'Every lecturer gave a student a 1st'

  How many students got 1^{st}'s? One or several?
- And it's still difficult to get an AI system to do proper reasoning without an explicit **knowledge base**.

  User: 'Book a flight to Denver on Tuesday'

  Sys: 'Okay, where from?'
- But why are the statistical methods so powerful?

Mathematical foundations

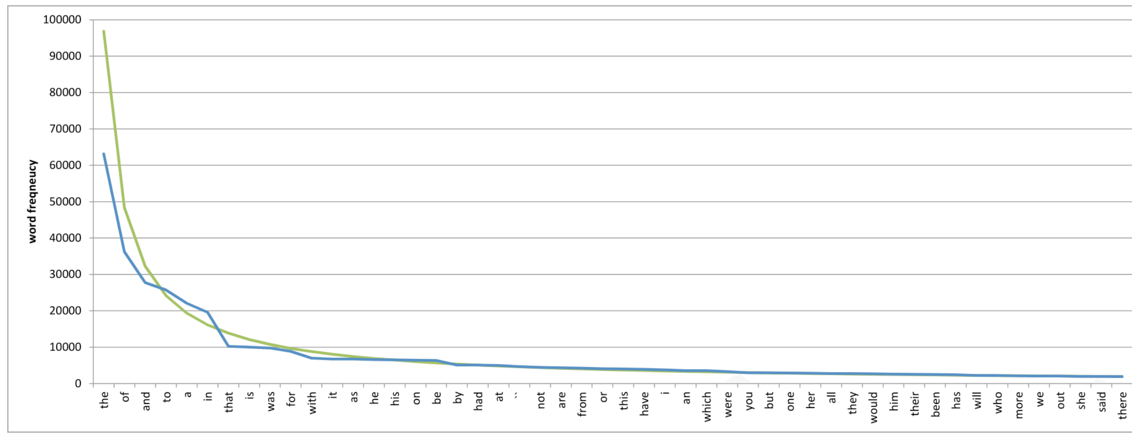
- In a corpus of text (or dialogue) you get many regular **patterns**.
- These occur fairly systematically.
- If you understand those patterns, you can figure out what is being talked about, as it’s similar to other examples.
- Simple methods can **scale** very fast.
- What are some of these **systematic properties**?

KEY POINT:

Language is Zipfian

Zipf's Law

- The frequency of any word is inversely proportional to its rank in the frequency table.
  - Brown corpus:
  - rank 1 'the': 7%
  - rank 2 'of': 3.5%
  - rank 3 'and': 2.9%



- This means:
  - We can capture most of the data easily
  - But there is a **very** long tail
  - And however big your corpus …
  - ... you will see new words as soon as you look outside it!

KEY POINT:

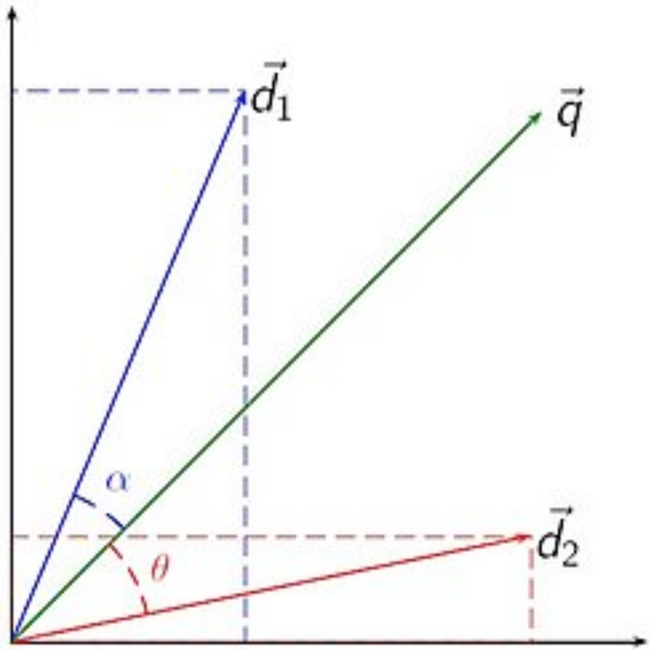
Words are not independent

Sentiment Analysis again (but with Statistical Models)

- We could **learn** these dictionaries
- Or we could train a **classifier**:
  - List of “positive” examples
  - List of “negative” examples
- Learn a decision function based on observed words … and combinations thereof

Texts as Feature Spaces

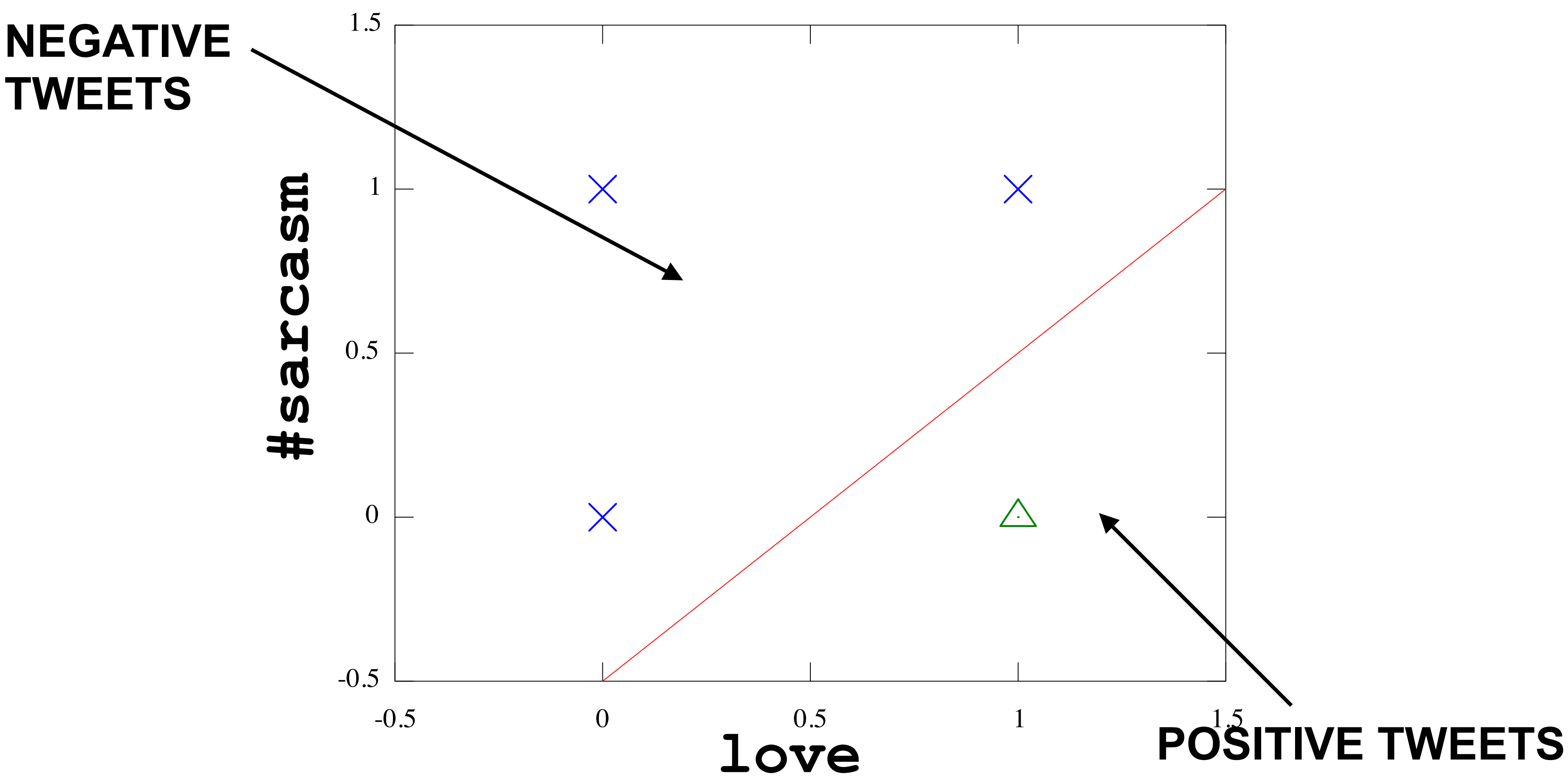
- We can characterise a text in terms of its words
- **Vector space models**
  - words = dimensions
- **“Bag of words”** model



Sentiment Analysis 2: Data-Driven Classification

- Geometric methods for classification using **Machine Learning**- fit a class boundary using data.

i love @justinbieber #sarcasm



Sentiment Analysis 2: Data-Driven Classification - Preprocessing

- We’re going to have to use the words
    - (what else is there?)
- But how do actually we get to them? i.e. what **pre-processing**?
- At least:
    - Sentence **segmentation**
        - (split? At what?)
    - Word **tokenisation**
        - (split? At what?)
- And maybe:
    - **Normalisation**, **spelling correction**
        - (how?)
    - **Stop word** removal
        - (really?)

Tokenisation

- Issues in tokenisation:
  - ***Finland's capital*** →
    Finland? Finlands? Finland's?
  - ***Hewlett-Packard*** → ***Hewlett*** and ***Packard*** as two tokens?
    - ***state-of-the-art***: break up hyphenated sequence.
    - ***co-education***
    - ***lowercase***, ***lower-case***, ***lower case*** ?
    - It's effective to get the user to put in possible hyphens
  - ***San Francisco***: one token or two? How do you decide it is one token?

Normalisation

- Need to “normalise” terms in indexed text as well as query terms into the same form
  - We want to match ***U.S.A.*** and ***USA***
- We most commonly implicitly define equivalence classes of terms
  - e.g., by deleting full-stops in a term
- Alternative is to do asymmetric expansion:
  - Enter: ***window*** Search: ***window, windows***
  - Enter: ***windows*** Search: ***Windows, windows, window***
  - Enter: ***Windows*** Search: ***Windows***
- Potentially more powerful, but less efficient

Normalisation: other languages

- Accents: ***résumé*** vs. ***resume***.
- Most important criterion:
  - How are your users likely to write their queries for these words?

- Even in languages that standardly have accents, users often may not type them

- German: ***Tuebingen*** vs. ***Tübingen***
  - Should be equivalent

- **<u>In next week's lab, you will do some preprocessing tasks in python.</u>**

What about …

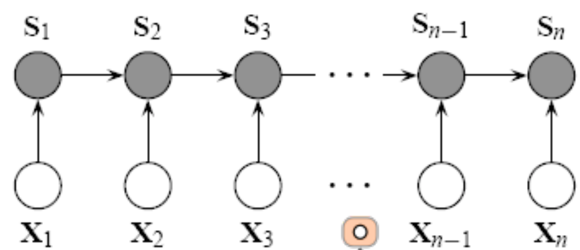
- Milk is good and not expensive
- Milk is expensive and not good

KEY POINT:

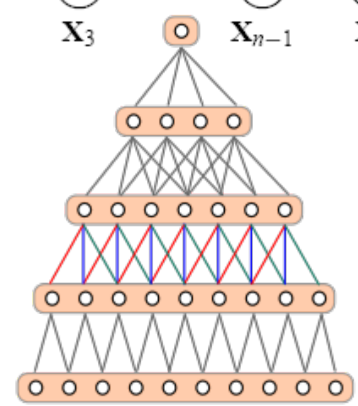
Language is not just a bag of words

Sequence modelling

- We can get a long way by using **sequence**
  - N-grams
    - [milk is], [is good], [good and], [and not], [not expensive]
    - [milk is], [is expensive], [expensive and], [and not], [not good]
  - Sequence models
    - Markov models
    - Conditional random fields



  - Convolutional / recurrent neural nets



What about …

- Milk is not very good
- Milk is not really very good
- Milk is not bad but good
- As bad as milk is, good things can come from it

- I hate happy birthdays and fluffy clouds
- I love disaster movies

- I like milk
- I like dairy products

KEY POINT:

Language has hierarchical structure

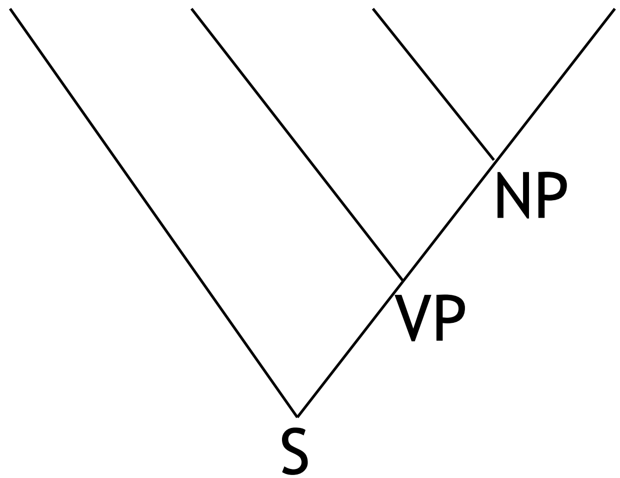
Levels of language interpretation

words:	Mary	hires	a	detective
parts of speech:	PN	VBZ	DET	CN
lemmata:	mary	hire	a	detective

TAGGING

STEMMING

syntax:



PARSING

semantics: $\exists x.\mathrm{detective}(x)\ \&\ \mathrm{hire}(\mathrm{mary},x)$

SEMANTIC PARSING

discourse:

e,x	
hire(e)	detective(x)
subj(e,mary)	obj(e,x)

DISCOURSE PARSING

What about …

- A: I like all milk, which is white and tasty

- B: I agree!
- C: No way.

- How can we tell what B and C **mean**?

KEY POINT:

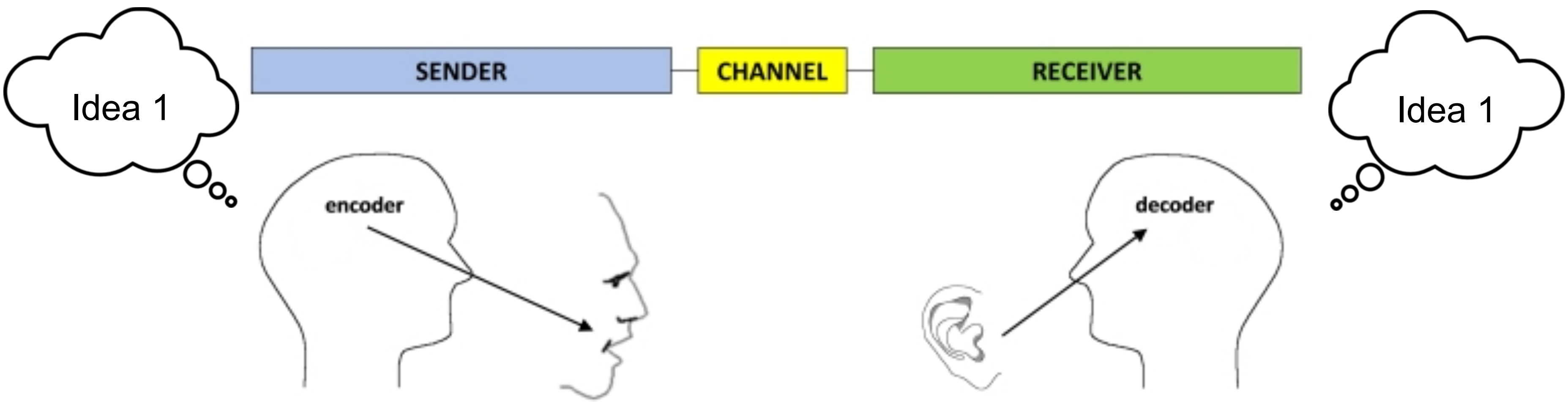
Language is ambiguous and context-dependent

OUTLINE

1) What is NLP and where is it used?
2) Managing big data: classification and extraction
3) Intro to statistical and probabilistic methods
4) Intro to dialogue and its challenges

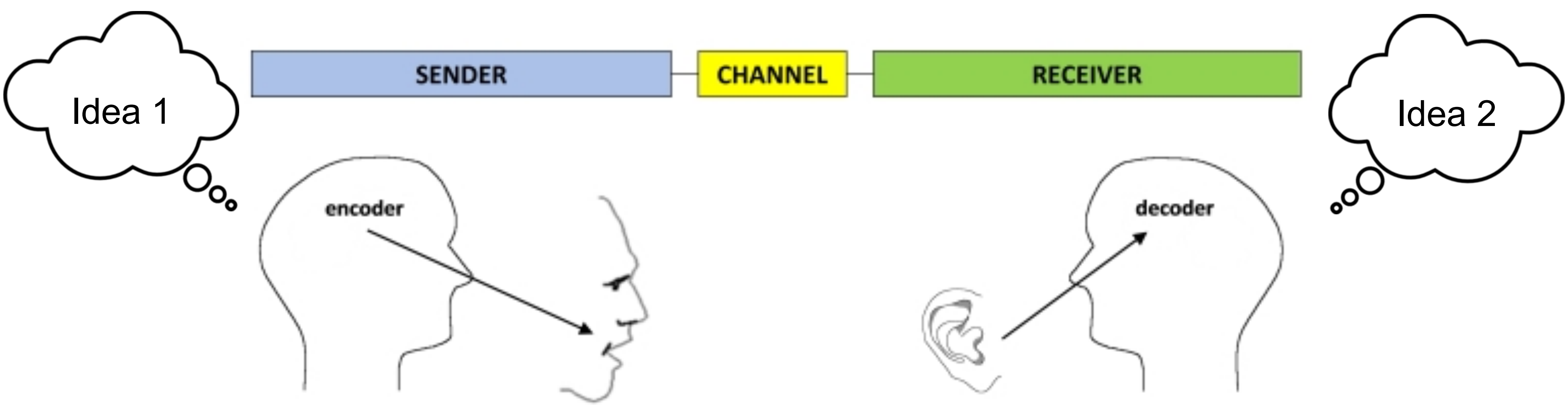
How do people communicate?

- First models similar to encoder/decoder model (Shannon, 1948).
- Communication based on a common code.



How can people *mis*communicate?

- Just noise in signal? More recent theories about aligning internal representations via ***communicative grounding*** (Clark 1996) mechanisms.
- A. ‘Put the apple over there’
  - B. ‘Where did you mean?’ (clarification)
  - A. ‘No, in the corner’ (repair)



How do people *mis*communicate?

- Self-repair/disfluency (every 25 words of natural dialogue), but not taken seriously by engineers:

  *"But one of **the, the** two things that I'm really. . ."*

  *"Our situation is **just a little bit, kind of** the opposite of that"*

  *"and you know it's like **you're, I mean,** employments are contractual by nature anyway"*

KEY POINT:

Dialogue is Messy!

And hard for systems…

How do we build systems to speak with humans?

- Dialogue system designers struggle to deal with the rich range of human dialogue behaviour and what people **mean** in their utterances/texts.
- However, many useful systems use simple assumptions to get things working.

How do we build systems to speak with humans?

- Google Dialogflow uses breaks things down to **intents** and **entities** and context variables.
- An intent is the recognized meaning of the user's intention e.g. I want a pizza -> *#orderfood*
- An entity is an individuated thing e.g. I want a pizza -> *entity:food=pizza*
- **<u>In next week's lab you will build a simple Google Dialogflow chatbot.</u>**

ECS763U/P Natural Language Processing

Julian Hough

Week 1: Introduction

Part 2: Syntax and Semantics Intro

(Many slides by Mehrnoosh Sadrzadeh)

Generative Grammars

A Generative System

S $\longrightarrow$ NP VP

VP $\longrightarrow$ itV, tV NP

tV $\longrightarrow$ drink, eat

itV $\longrightarrow$ fly, sleep

NP $\longrightarrow$ vampire, butterfly, blood

Generative Grammars

Vampires drink blood.

S $\longrightarrow$ Vampires VP

VP $\longrightarrow$ drink blood

tV $\longrightarrow$ drink

NP $\longrightarrow$ blood

Logical Grammars

A Logical System

Division and Multiplication

itV: $\dfrac{S}{NP}$

tV: $\dfrac{\dfrac{S}{NP}}{NP}$

itV: fly, sleep tV: drink, eat

NP: vampire, butterfly, blood

Logical Grammars

Butterflies sleep.

$$NP \times \frac{S}{NP} = S$$

Vampires drink blood.

$$NP \times \frac{\frac{S}{NP}}{NP} \times NP = NP \times \frac{S}{NP} = S$$

Ambiguity

Spurious Ambiguity

Generative Grammars

John saw a man with binoculars.

S → John VP

VP → saw a man with binoculars

tV → saw

NP → a man with binoculars

Meaning 1

Generative Grammars

John saw a man with binoculars.

S $\longrightarrow$ John VP PP

VP $\longrightarrow$ saw a man

tV $\longrightarrow$ saw

NP $\longrightarrow$ a man

PP $\longrightarrow$ with binoculars

Meaning 2

Ambiguity

Semantic Ambiguity

Fisher men cast their nets.

The moon cast its light.

Ambiguity

- **How can we deal with the ambiguity of the meaning of a word like 'cast'?**

- **How do we deal with word meaning in general?**
  - **Semantics**
    - **Formal logical methods**- each word maps to a formula
    - **Distributional methods**- a word's meaning is defined by its use (where it occurs in a text relative to others)

Guess the missing word

It is difficult to make a single, definitive description of the **folkloric** though there are several elements common to many European **legends**. were usually reported as bloated in appearance, and **ruddy**, **purplish**, or dark in colour; these characteristics were often attributed to the drinking of **blood**. [· · ·] Indeed, **blood** was often seen seeping from the mouth and nose of the when it was seen in its **shroud** or **coffin** and its left eye was often open. [· · ·] In Christianity, the was viewed as ”a **dead** person who retained a semblance of life and could leave its **grave**-much in the same way that Jesus had risen after his **death** and **burial** and appeared before his followers. In Asia, [· · ·] a wanders around animating **dead bodies** at night, attacking the living much like a **ghoul**.

It is difficult to make a single, definitive description of the **folkloric** **vampire**, though there are several elements common to many European **legends**. **Vampire** were usually reported as bloated in appearance, and **ruddy**, **purplish**, or dark in colour; these characteristics were often attributed to the drinking of **blood**. $[\cdots]$ Indeed, **blood** was often seen seeping from the mouth and nose of the **vampire** when it was seen in its **shroud** or **coffin** and its left eye was often open. $[\cdots]$ In Christianity, the **vampire** was viewed as ”a **dead** person who retained a semblance of life and could leave its **grave**-much in the same way that Jesus had risen after his **death** and **burial** and appeared before his followers. In Asia, $[\cdots]$ a **vampire** wanders around animating **dead bodies** at night, attacking the living much like a **ghoul**.

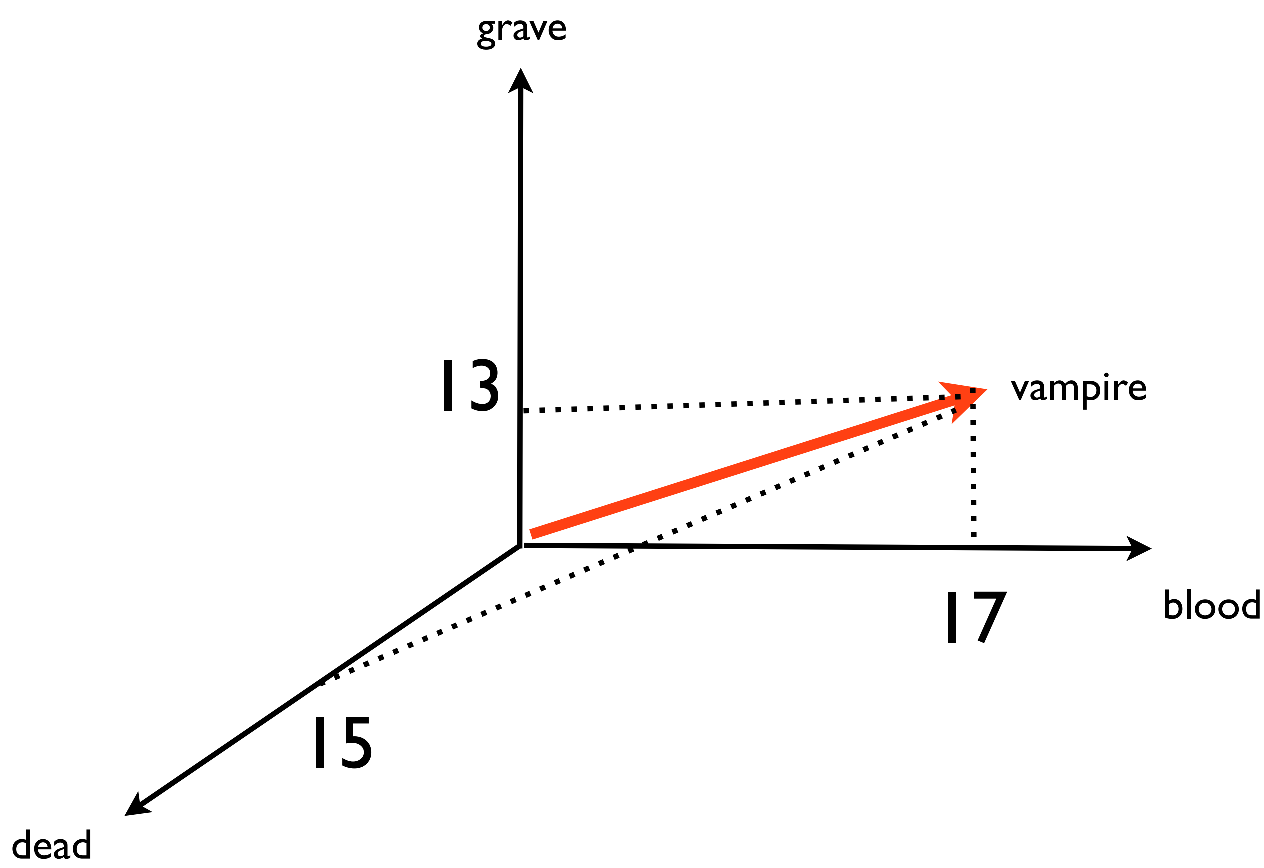
Guess the missing word

Butterflie are beautiful, flying insects with large scaly wings. Like all insects, they have six jointed legs, 3 body parts, a pair of antennae, compound eyes, and an exoskeleton. The three body parts are the head, thorax (the chest), and abdomen (the tail end). The butterfly's body is covered by tiny sensory hairs. The four wings and the six legs of the [illegible] are attached to the thorax. The thorax contains the muscles that make the legs and wings move. Butterflies are very good fliers. They have two pairs of large wings covered with colorful, iridescent scales in overlapping rows. Lepidoptera (butterflies and moths) are the only insects that have scaly wings. The wings are attached to the butterfly's thorax (mid-section). Veins support the delicate wings and nourish them with blood.

Butterflie are beautiful, flying insects with large scaly wings. Like all insects, they have six jointed legs, 3 body parts, a pair of antennae, compound eyes, and an exoskeleton. The three body parts are the head, thorax (the chest), and abdomen (the tail end). The butterfly's body is covered by tiny sensory hairs. The four wings and the six legs of the butterfly are attached to the thorax. The thorax contains the muscles that make the legs and wings move. Butterflies are very good fliers. They have two pairs of large wings covered with colorful, iridescent scales in overlapping rows. Lepidoptera (butterflies and moths) are the only insects that have scaly wings. The wings are attached to the butterfly's thorax (mid-section). Veins support the delicate wings and nourish them with blood.

The Maths Behind: Words as vectors

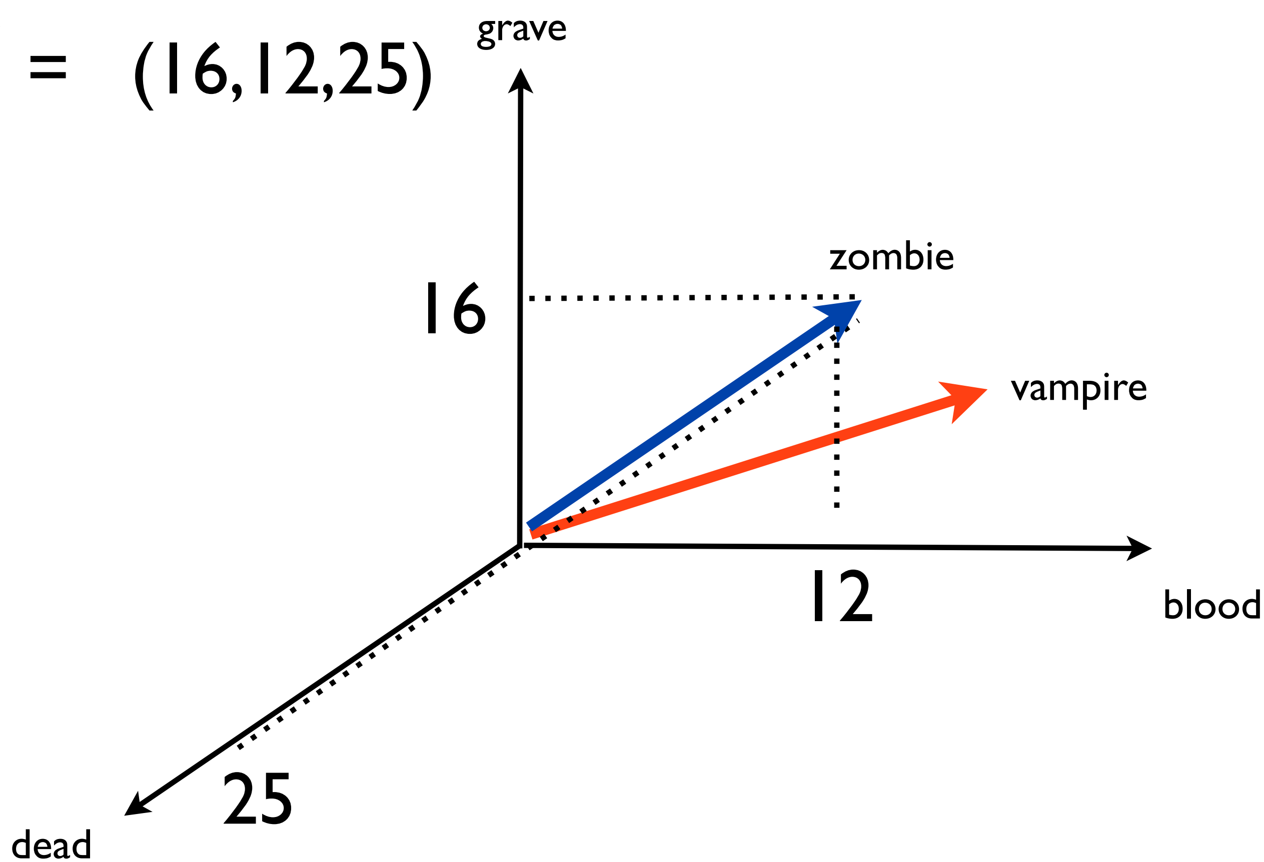
$$\overrightarrow{\textit{vampire}} = (17, 13, 15)$$



The Maths Behind: Words as vectors

$\overrightarrow{vampire} = (17,13,15)$

$\overrightarrow{zombie} = (16,12,25)$

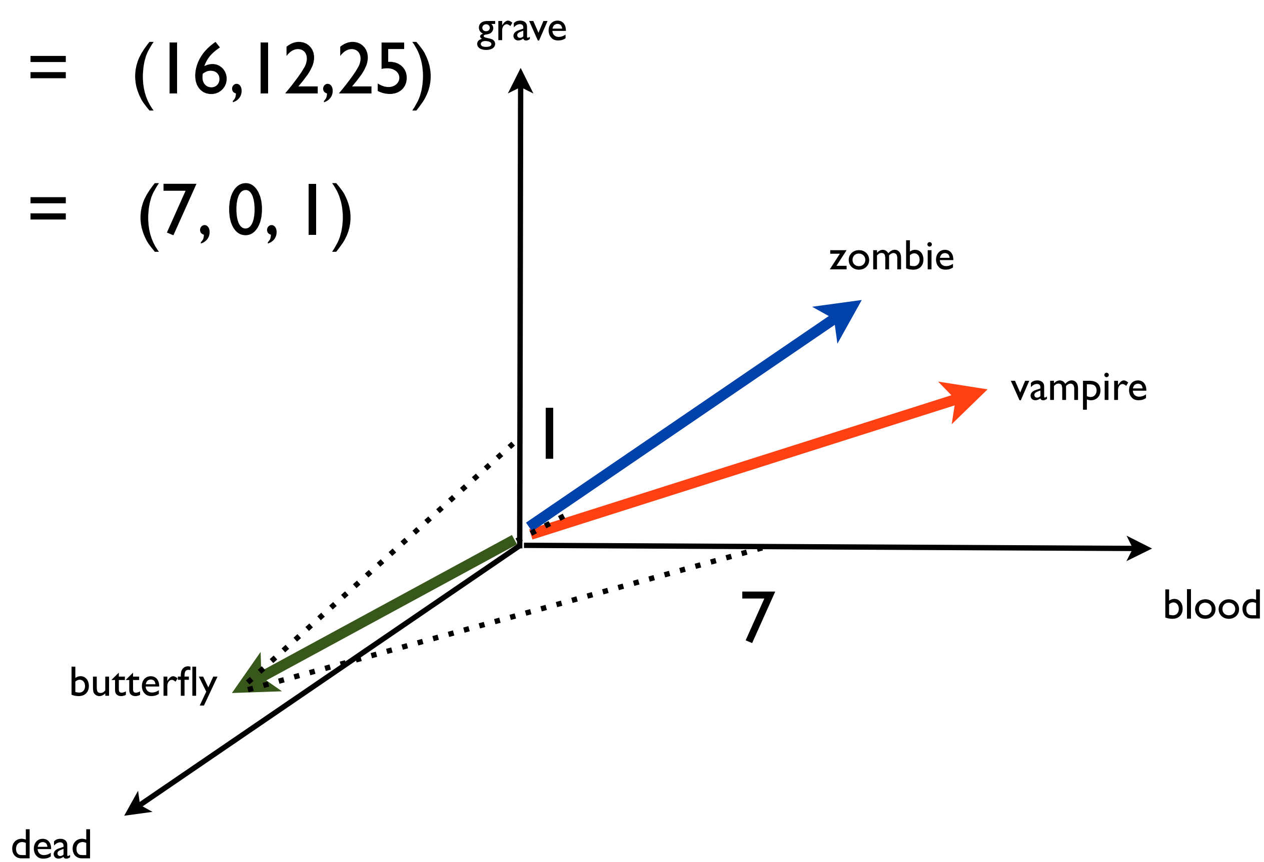


The Maths Behind: Words as vectors

$$\overrightarrow{vampire} = (17, 13, 15)$$

$$\overrightarrow{zombie} = (16, 12, 25)$$

$$\overrightarrow{butterfly} = (7, 0, 1)$$

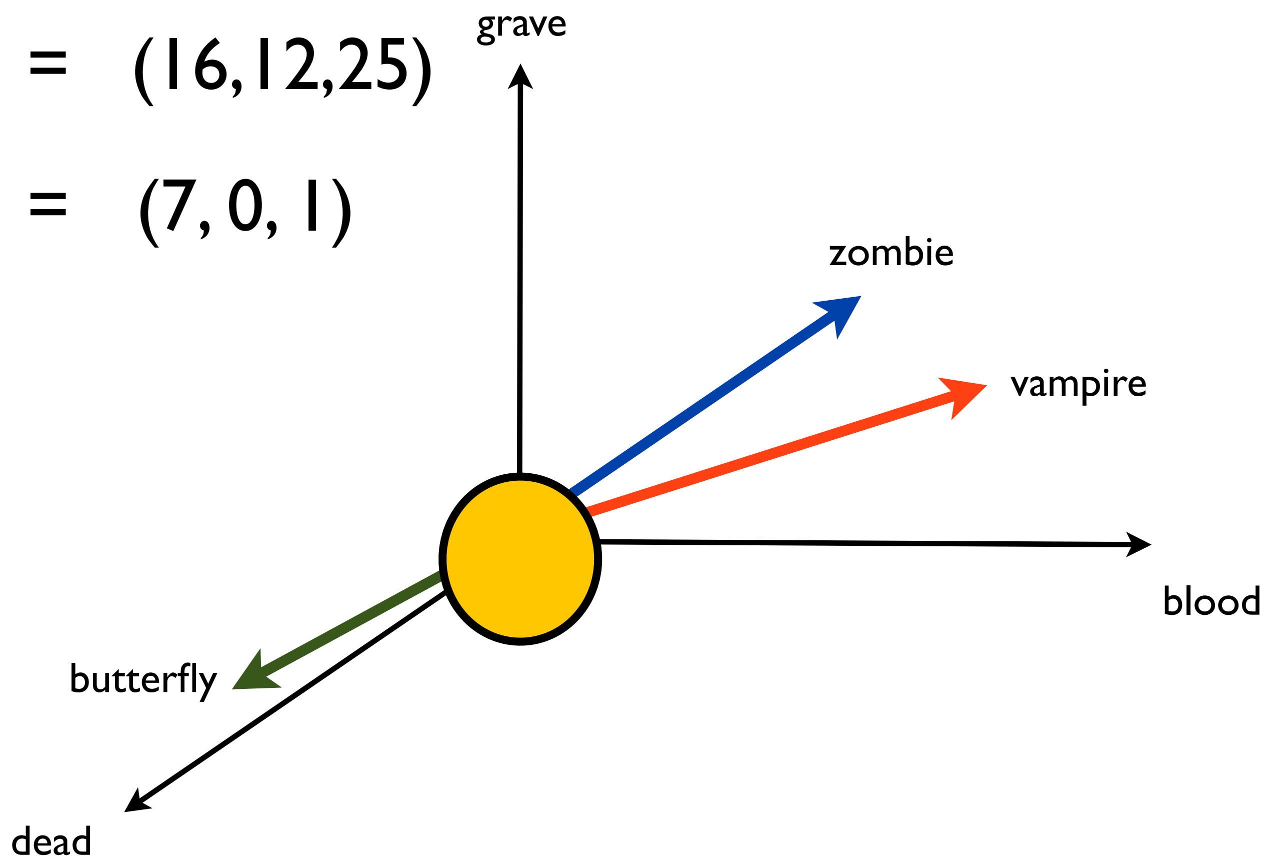


The Maths Behind: Words as vectors

$$\overrightarrow{\textit{vampire}} = (17, 13, 15)$$

$$\overrightarrow{\textit{zombie}} = (16, 12, 25)$$

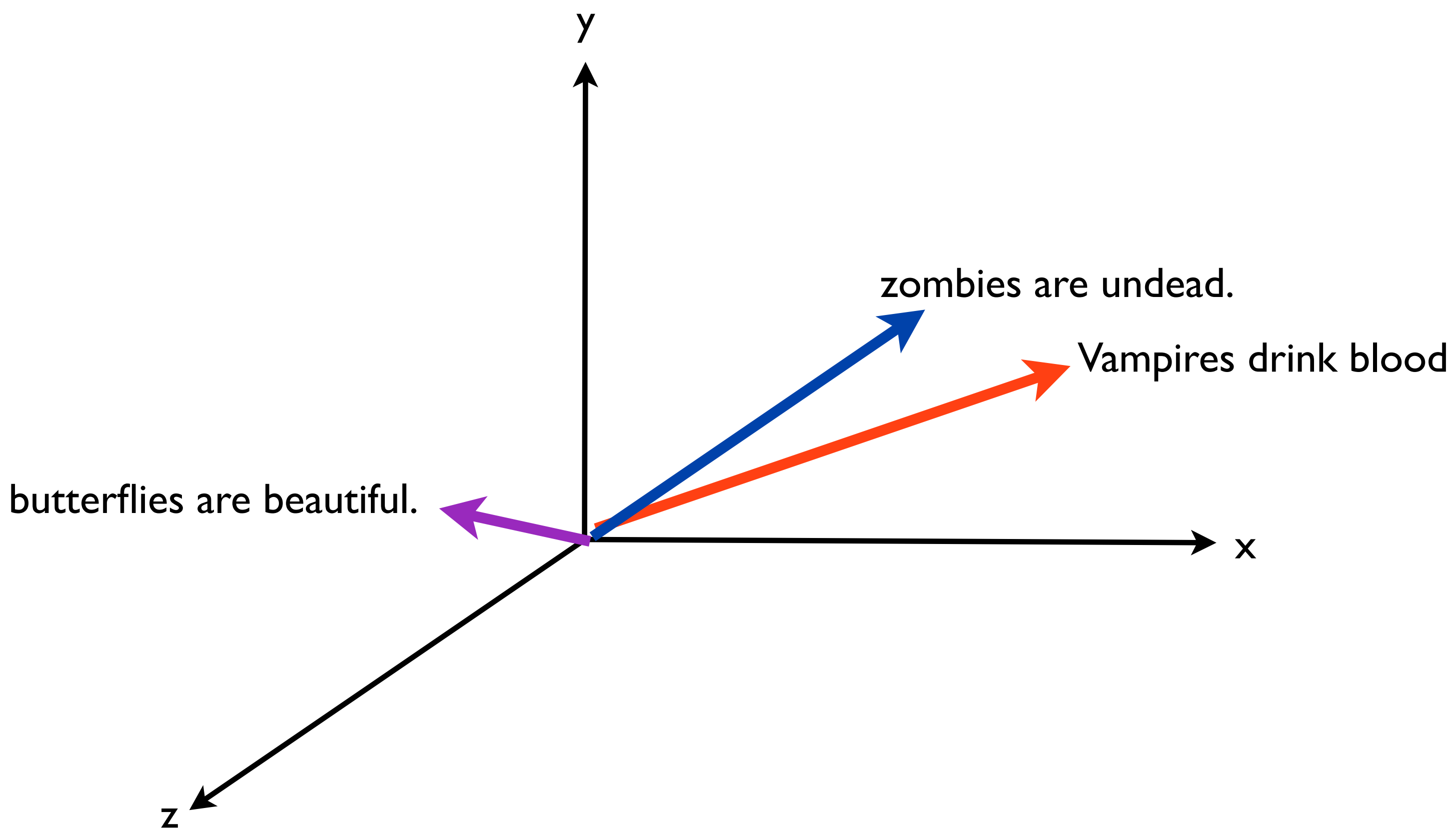
$$\overrightarrow{\textit{butterfly}} = (7, 0, 1)$$



Guess the missing sentence

[redacted] **Vampire** were usually reported as bloated in appearance, and **ruddy**, **purplish**, or dark in colour; these characteristics were often attributed to the drinking of **blood**. [· · ·] Indeed, **blood** was often seen seeping from the mouth and nose of the **vampire** when it was seen in its **shroud** or **coffin** and its left eye was often open. [· · ·] In Christianity, the **vampire** was viewed as ”a **dead** person who retained a semblance of life and could leave its **grave**-much in the same way that Jesus had risen after his **death** and **burial** and appeared before his followers. In Asia, [· · ·] a **vampire** wanders around animating **dead bodies** at night, attacking the living much like a **ghoul**.

Sentences as vectors?

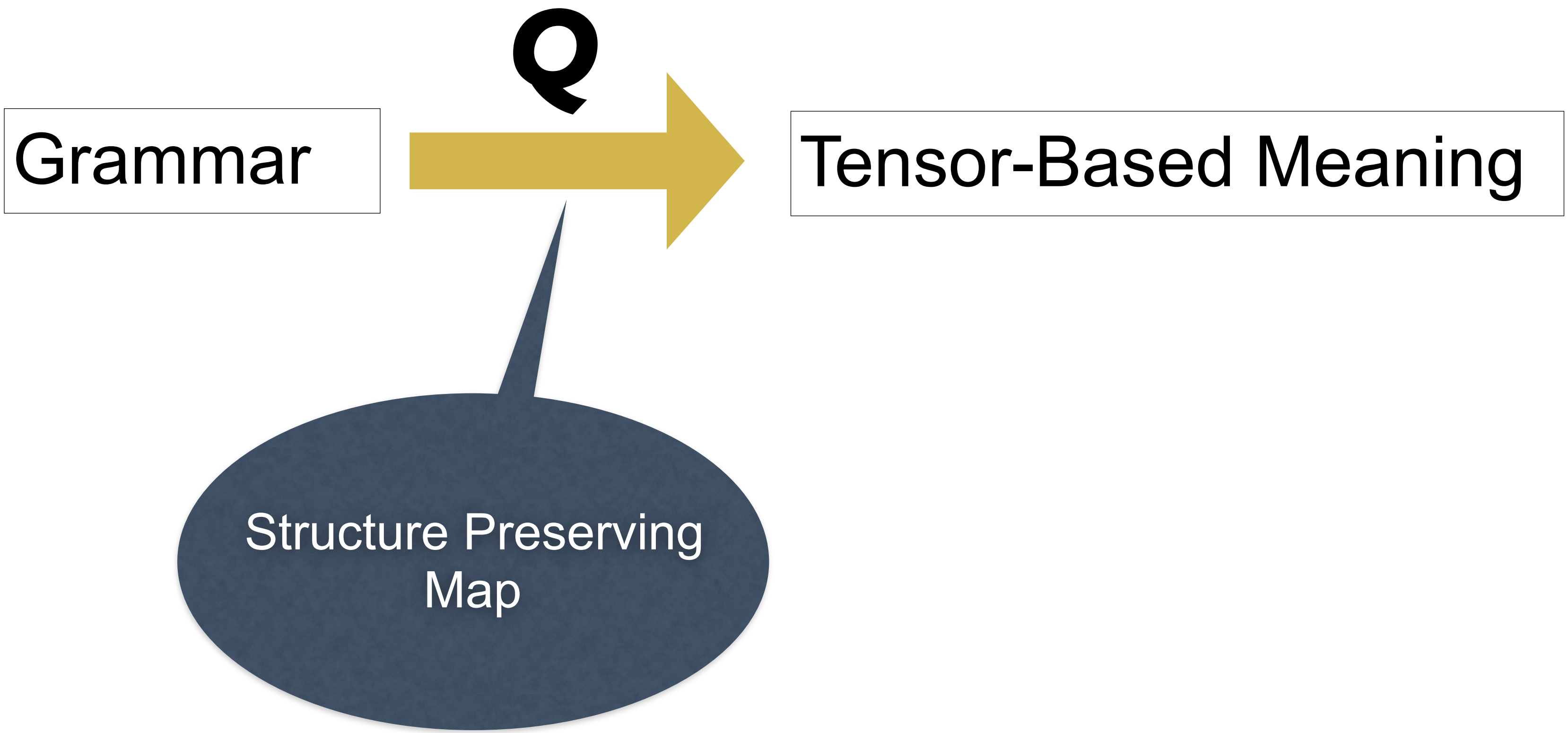


one way: simple vector operations

$$\overrightarrow{\text{vampires kill men}} = \overrightarrow{\text{vampires}} + \overrightarrow{\text{kill}} + \overrightarrow{\text{men}}$$
$$= \overrightarrow{\text{vampires}} \odot \overrightarrow{\text{kill}} \odot \overrightarrow{\text{men}}$$

$$\overrightarrow{\text{vampires kill men}} = \overrightarrow{\text{men kill vampires}}$$

another way: grammar based tensor operations



Example Tensors

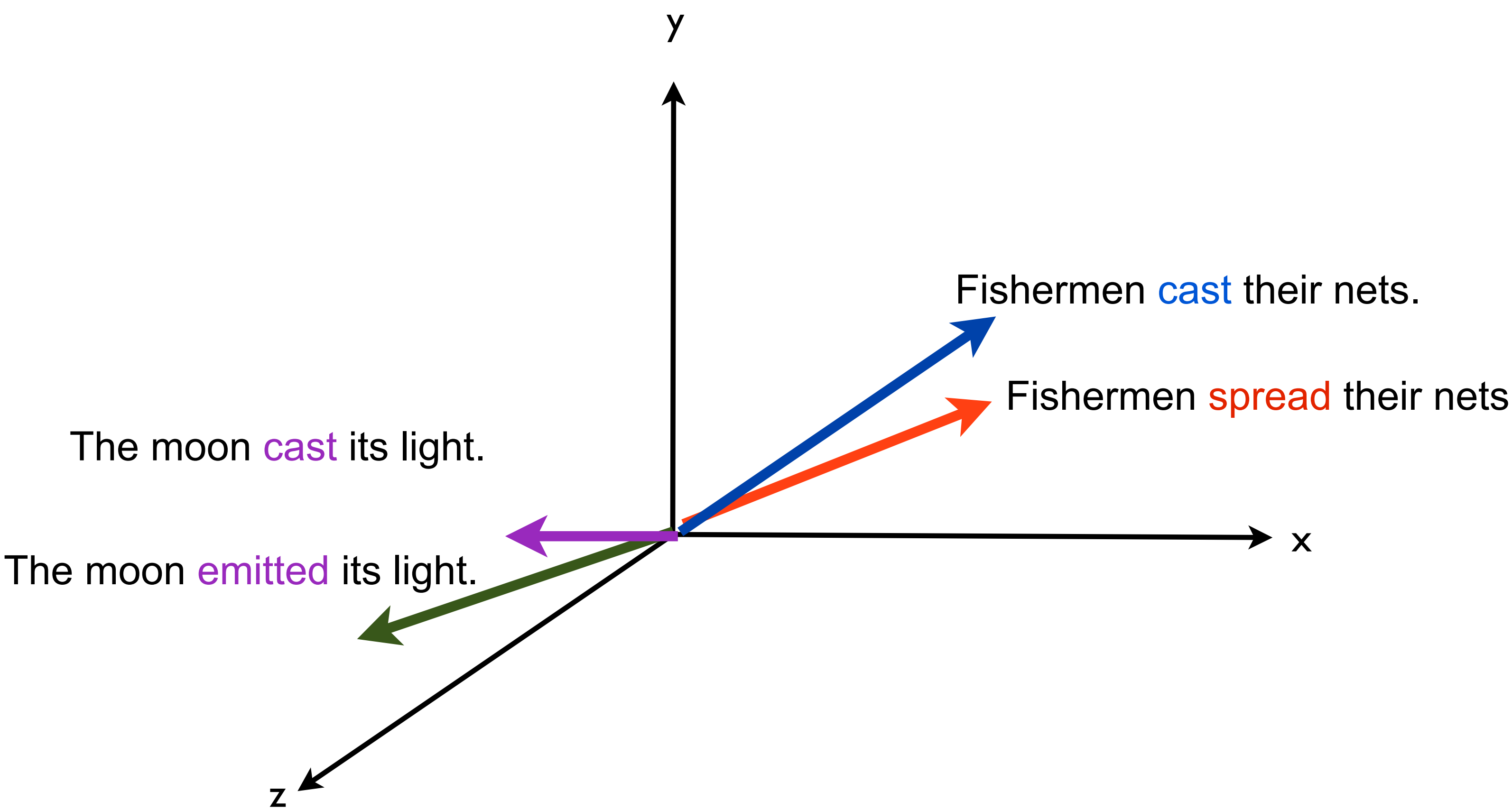
$$\overrightarrow{\text{red car}} = \mathcal{F}(nn^l n \to n)(\overline{\text{red}} \otimes \overrightarrow{\text{car}})$$

$$= \sum_{ij} \sum_{k} C_{ij} C_k \overrightarrow{n_i} \langle \overrightarrow{n_j} \mid \overrightarrow{n_k} \rangle$$

$$\overrightarrow{\text{men like red cars}} = \mathcal{F}(nn^r s n^l n n^l n \to s)(\overrightarrow{\text{men}} \otimes \overrightarrow{\text{like}} \otimes \overrightarrow{\text{red}} \otimes \overrightarrow{\text{cars}})$$

$$= \sum_{i} \sum_{jkl} \langle \overrightarrow{n_i} \mid \overrightarrow{n_j} \rangle \overrightarrow{s_k} \langle \overrightarrow{n_l} \mid \sum_{mn} \sum_{o} C_{mn} C_o \langle \overrightarrow{n_m} \mid \overrightarrow{n_n} \rangle \overrightarrow{n_o} \rangle$$

Word Sense Disambiguation



Entity Disambiguation

DBPedia Spotlight:

https://www.dbpedia-spotlight.org/demo/

BBC R&D
Projects

Module Housekeeping

Reading, Labs, Coursework, Exams

Text Books

MODULE READING LIST

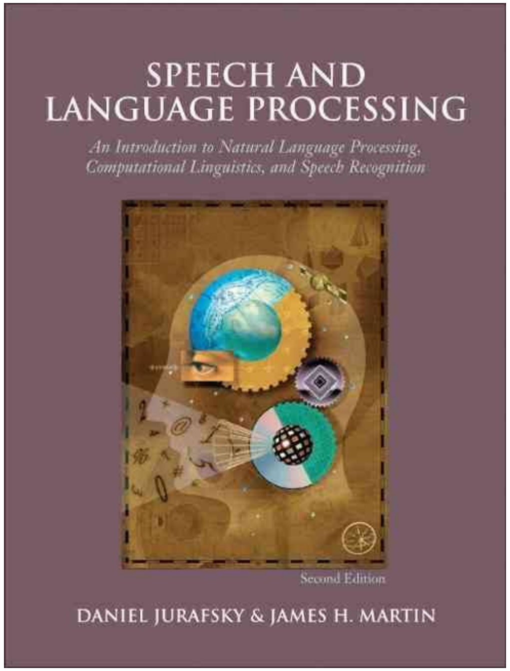
- Speech and language processing Daniel Jurafsky, James H. Martin 2014 (electronic resource)

  Book
- Speech and language processing James H. Martin 2013

  Book
- Foundations of statistical natural language processing Christopher D. Manning, Hinrich Schütze 2003, c1999

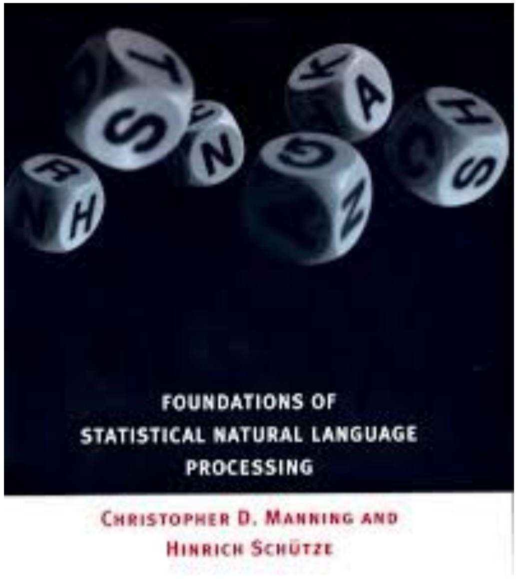
  Book
- NLTK Book Steven Bird, Ewan Klein, Edward Loper

  Webpage



Online Book (newest edition in progress)

https://web.stanford.edu/~jurafsky/slp3/



Online Book

http://www.corpus.unam.mx/cursoenah/ManningSchutze_1999_FoundationsofStatisticalNaturalLanguageProcessing.pdf

Two example papers on advanced topics:

On the Means for Clarification in Dialogue

Matthew Purver, Jonathan Ginzburg, Patrick Healey

http://www.aclweb.org/anthology/W01-1616

Vector Space Models of Lexical Meaning

Stephen Clark

https://www.cl.cam.ac.uk/~sc609/pubs/sem_handbook.pdf

Lecture Outline

Week 1: Motivation and introduction
Week 2: Statistical methods 1: language modelling
Week 3: Statistical methods 2: classification/regression
Week 4: Statistical methods 3: sequence modelling (HMMs, CRFs)
Week 5: Syntax 1: generative and logical grammars
Week 6: Syntax 2: dependency and probabilistic grammars
Week 7: Syntax 3: limitations of syntax, tools and TreeBanks
Week 8: Semantics 1: formal and distributional semantics
Week 9: Semantics 2: compositional distributional semantics
Week 10: Discourse & Dialogue 1: coreference resolution
Week 11: Discourse & Dialogue 2: dialogue models and systems
Week 12: Review week

Learning Outcomes

Statistical Methods

- Explain how language models are used in NLP applications.
- Build and evaluate an n-gram language model.
- Explain how classification methods are often used in NLP tasks, and what features are often used.
- Build a simple classifier for an NLP task and evaluate it.
- Explain how Hidden Markov Models (HMMs) and Conditional Random Fields (CRFs) are used in NLP.
- Build HMM and CRF based taggers and evaluate them.

Learning Outcomes

Syntax

- Analyse grammatical structures of phrases and sentences of natural language using the different taught systems
- Be able to distinguish between various different ambiguities
- Analyse different meanings of ambiguous sentences
- Disambiguate sentences that have different grammatical structures using probabilistic grammars
- Know what is a Treebank and how to use it
- Compute probabilities of parses
- Be familiar with limitations of parsers and how to overcome them
- Define key elements of different grammatical systems and how they are different from one another.

Learning Outcomes

Semantics

- Compute symbolic meanings for phrases and sentences of language
- Be familiar with vector space models of corpora of text
- Compute vector meanings for words and sentences
- Compute degrees of semantic similarity
- Be familiar with how to evaluate these semantic similarities
- Define key elements of each of the semantics systems, e.g. symbolic, distributional and compositional distributional systems.

Learning Outcomes

Discourse & Dialogue

- Explain how sentences relate to each other in a long text.
- Explain the importance of coreference resolution in discourse and dialogue.
- Describe popular methods in coreference resolution and their evaluation.
- Describe the unique challenges of spoken and text-based dialogue.
- Describe how Questions-Under-Discussion (QUD) based formal dialogue models work.
- Describe the components of a dialogue system/chatbot and how they work together.

Assessment

40%: Coursework
4 Lab Sheets (20%, 5% each)
Project (20%)

60%: Final Exam

Labs and Project

- Labs are on Mondays 2-4. ITL Second Floor.
- They start at week 2 with an unassessed lab on how to use Python/do pre-processing and build a chatbot with Google's Dialogflow.

Weeks 3-5: assessed labs 1+2

Weeks 6-8: assessed labs 3+4

Weeks 9-11: project labs

Weeks 12: revision/exam labs

Labs and Project

- Each lab has a lab sheet that will be put on QMPlus just before the lab. You will hand them in two at a time.

- For lab sheets 1 and 2 you have until **Friday (12pm noon) of week 5** for online submission.

- For lab sheets 3 and 4 you have until **Friday (12pm noon) of week 8** for online submission.

- We will try to get feedback to you within 2 weeks, at most 3.

Labs and Project

- The project will be released on week 8.
- It will involve implementing a natural language tool of some sort using the techniques taught during the lectures.
- The project needs to be submitted on QMPlus by the **end of week 11 (Friday 12pm noon).**

Lab Outline

Week 1: (No lab)
Week 2: Unassessed lab: introduction to python, NLTK and chatbots
Week 3: Assessed lab 1: language modelling
Week 4: Assessed lab 2: classification/regression
Week 5: Catch up lab finishing labs 1 and 2 **(hand-in end of week)**
Week 6: Assessed lab 3: generative and logical grammars
Week 7: Assessed lab 4: dependancy and probabilistic grammars
Week 8: Catch up lab finishing labs 3 and 4 **(hand-in end of week)**
Week 9: Project lab
Week 10: Project lab
Week 11: Project lab **(hand-in end of week)**
Week 12: Any exam questions

Exam

4 questions in total:

1 on statistical methods:

- Explain how probabilities of sequences of words are calculated in ngram models.
- Describe examples of uses of HMMs and CRFs in sequence-modelling tasks.
- Describe how several classification methods work in NLP tasks.
- Explain how evaluation is done in several NLP tasks.

Exam

4 questions in total:

1 on syntax:

- Define elements of
  - formal grammatical systems introduced
  - their limitations
- Specify grammatical structures of sentences
- Disambiguate sentence meaning and structure
- Compute probabilities of different parses of sentences

Exam

4 questions in total:

1 on semantics:

- Define of elements of
    - symbolic semantic systems
    - distributional and compositional distributional semantics systems
- Compute the semantic structure of sentences
- Compute vector semantics for words and phrases
- Compute semantic similarities of words and phrases

Exam

4 questions in total:

1 on discourse and dialogue:

- Describe why coreference and anaphora detection is important for different tasks.
- Identify and describe several dialogue phenomena in a dialogue transcript.
- Describe a QUD-based formal dialogue model.
- Describe the components of a dialogue system or chatbot and what they do.

Reading

- Christopher D. Manning and Hinrich Schuetze (2003/1999). **Foundations of Statistical Natural Language Processing.** Chapter 1

- (optional) If you aren’t familiar with Python / don’t know much about language or corpora:
  - **NLTK book** (online), Chapters 1 and 2