

# Voting Advice via Direct Access to the Relevant Data

Maarten Marx

Universiteit van Amsterdam

Politicologen etmaal, Amsterdam, 2011-06-09



## Outline

- Two types of voting advice systems
- Lipschits on the web
- Technical details
- Conclusions and how further?



## 2 types of voting advices systems

- Lipschits method (1977–1998)
- Stemwijzer method (on the web, from 1998)

**Same users:** voters

**Same motivation:** help voters in making a choice, based on manifesto data



## Primary goals

- Lipschits**
- quickly **find** standpoint of party in manifesto on topic X
  - easily **compare** standpoints of parties on topic X

universiteit onderwijze	volgheidsbijdrage (internationaal)
stic: hoger onderwijs (incl. Hoger Beroepsonderwijs)	ACV: 33
	CDU: 9, 46-48, 55-57, 59, 94
	D66: 2, 4, 90, 113, 116, 124, 126, 127
universiteit onderzocht:	GL: 8, 33, 36, 37, 40, 41
stic: hoger onderwijs (incl. Hoger Beroepsonderwijs)	GPV: 90
	PvdA: 81, 82, 85, 87
universiteit:	RPF: 67, 68, 71
stic: hoger onderwijs (incl. Hoger Beroepsonderwijs)	SCP: 54, 59
	SP: 1, 31, 32, 35
	VVD: 44, 45
Utrecht:	Verhulst/Donald:
stic: grote steden/grote-stedenbeleid	stic: Verenigde Staten

- Stemwijzer** quickly find which parties best fit the user on a (for all voters) fixed set of topics



## Secondary goals

Reuse obtained data for scientific research.

### Lipschits

- standardize manifestos
- rich list of salient topics for each election
- rich controlled vocabulary
- (Now) excellent training data for creating classifiers

### KiesKompas

- positions of parties on several topics
- positions of "the electorate" on these topics



## Differences between Lipschits and Stemwijzer

- One size fits all **vs** user decides on topics and parties
- Direct **vs** indirect access to primary sources
- Different input-output behaviour



## Input-output

	In	Out
Stemwijzer	answer to questions	ranked list of parties
Kieskompas	"	model of user as a party
Lipschits	controlled vocabulary terms	relevant paragraphs for each party
VerkiezingsKijker	" or free search terms	"



## Demo: 'Lipschits on the web'

[verkiezingskijker.nl](http://verkiezingskijker.nl)



## History VerkiezingsKijker

### TK 2006 UvA-Stemwijzer.

- Eddy Habben Jansen: take Lipschits as inspiration
- Motivation Stemwijzer: add “proof” for party positions from their manifestos

**PS 2007 UvA-Kieskompas.** Verkiezingskijker used to facilitate large amount of party-placements (12 provinces  $\times$  10 parties  $\times$  36 positions = 4320 placements).

**DNPP corpus UvA Bsc thesis:** search engine for DNPP manifesto corpus.

### TK 2010 Google



PoliticalMashup

## Technical Details: outline

1. Idea
2. How to do it
3. Main problem
4. Solutions



PoliticalMashup

## Idea verkiezingskijker

- Replicate Lipschits, “Google style”
- Add free keyword search
- Make it scalable, faster to make (and without a Lipschits . . . )



PoliticalMashup

## How to do that?

- Collect manifestos (in time . . . )
- Standardize them into one data format
- Partition each manifesto into meaningful units (paragraphs)

**Outcome** Basic Google style search engine which returns on each search term a ranked list of paragraphs

**Advanced search** restrict to parties



PoliticalMashup

## Main problem: Semantic gap

Voter and manifesto use different terms to talk about the same topic

- different parties use different terms to talk about the same topic
- small amounts of text per retrieval unit make this problem worse
- **Recall Problem:** system does not retrieve all relevant paragraphs.



PoliticalMashup

## Two solutions to this

### Hierarchical controlled vocabulary

- Basically back to Lipschits.
- Burden at the user.

### Document expansion

- **find related terms** (schiphol vliegveld luchthaven vliegtuig . . . )
- **expand paragraphs:** if it contains one term, add all others
- **Aim:** Improve recall.
- **Danger:** topic drift (thus more false positives)



PoliticalMashup

## Predicament

With both solutions we seem to be back at Lipschits and need to do all the work he did . . .



PoliticalMashup

## Our solution: learning from examples

- Stemwijzer created a list of 100 important election topics.
- For each topic, Stemwijzer found 5 highly relevant paragraphs
- From these paragraphs we harvested all **overused** terms (using corpus comparison techniques [Rayson, Garside 2000])
- For each topic we took the top k terms
- Quick manual check to remove outliers
- **Output:** classifier for each topic, and set of expansion terms.



PoliticalMashup

## Conclusion and what next?

- Both systems are **complimentary**.
  - Modern Lipschits system is useful for both **makers** and **users** of stemwijzer-like systems.
- Fine grained classification of manifestos (and alternatives . . . ) is useful for **comparative research** (e.g., Breeman-Timmermans, Louwerse)



## What next/Discussion

- Standardization of controlled vocabularies and development of high quality gold standard data is desirable
- **Soon:** Lipschits 1998 available in Excel and as a fully searchable hyperlinked web-document.
- **Wish?** same for the "Verkiezingsprogramma's met cd-rom" (Holsteyn et al) series?  
Or is the system by Google sufficient?

